



Reconocimiento-CompartirIgual 3.0
España (CC BY-SA 3.0 ES)

Introducción a la Ciberseguridad

Práctica 3.2: Redes e Internet

Marta Beltrán Pardo
Miguel Calvo Matalobos

Agradecimientos (versiones anteriores): Isaac Martín de Diego y Alberto Fernández Isabel

Contenidos

Contenidos.....	2
1. Introducción	3
2. Material de la práctica.....	4
3. Normativa y evaluación	5
4. Enunciado de la práctica	6
4.1 Página Web con HTML.....	6
Ejercicio 8	6
4.2 Página Web con PHP	15
Ejercicio 9	15
4.3 Inyección SQL	19
Ejercicio 10	19

1. Introducción

En la tercera práctica de la asignatura vamos a realizar varias tareas relacionadas con las Redes e Internet. Esta práctica, se divide en dos partes ([Practica3.1_Guion.pdf](#) y [Practica3.2_Guion.pdf](#)):

- En la primera parte de la práctica ([Practica3.1_Guion.pdf](#)), programaremos un escáner de puertos en C. Posteriormente montaremos una máquina virtual con LAMP (Linux + Apache + MySQL + PHP) para probar nuestro escáner de puertos.
- En la segunda parte ([Practica3.2_Guion.pdf](#)), montaremos nuestra primera página web con HTML y PHP para añadir una zona privada empleando una base de datos MySQL. El objetivo de esta segunda parte será entender la diferencia entre código estático y dinámico. Además, probaremos nuestra primera inyección SQL para poder acceder a la zona privada sin tener cuenta de usuario.

2. Material de la práctica

A continuación, se exponen los archivos necesarios para la realización de esta práctica, que pueden descargarse desde el Aula Virtual:

- **Practica3.2_Guion.pdf:** este documento. Se corresponde con el guion de la primera parte de la práctica y en él están contenidas todas las explicaciones necesarias para su realización.

Además, será necesario descargar el programa **VirtualBox** en su última versión. Este programa, servirá para crear la máquina virtual que se utilizará a lo largo del desarrollo de esta práctica. Puede descargarse desde el [siguiente enlace](#). También será necesario el pack de extensiones “Oracle VM VirtualBox”, que permitirá la configuración y el uso de ciertos parámetros y características en las máquinas virtuales (puede descargarse [aquí](#)).

Para el desarrollo de esta práctica, hará falta también la descarga de una distribución de un sistema operativo Linux. En este sentido, puede utilizarse la [máquina virtual “Kali Linux”](#) utilizada en prácticas anteriores o una máquina virtual creada a partir de una ISO de [“Kali Linux”](#), [“Ubuntu”](#) o cualquier otra distribución Linux que conozcas. Este sistema operativo será el utilizado a lo largo del desarrollo de esta práctica.

Por último, se deberá tener instalada y configurada, en la máquina virtual mencionada en el anterior párrafo, la pila LAMP (véase la primera parte de la práctica, “Practica3.1_Guion.pdf”).

3. Normativa y evaluación

En este apartado se detalla el formato de entrega de la práctica y la forma en la que se evaluará la misma:

- El porcentaje de la nota final de la asignatura al que corresponde esta práctica puede consultarse en la Guía docente de la propia asignatura.
- La práctica deberá realizarse, de forma obligatoria, en grupos de dos personas. Para la asignación de los grupos se deberán seguir las indicaciones del profesor.
- Cada grupo deberá:
 - Realizar una única memoria (puede utilizarse esta misma guía como plantilla) en la que responda las preguntas planteadas y/o en la que se exponga, de forma argumentada, las decisiones tomadas para la realización de la práctica.
 - Desarrollar y/o modificar tantos archivos de código como se soliciten en los diferentes ejercicios que forman la práctica.
- El resultado de la realización de la práctica consistirá en un fichero .zip llamado **Practica3.zip** que deberá entregarse a través del Aula Virtual en el espacio habilitado para ello y en la fecha límite allí expuesta. Este fichero debe contener:
 - La memoria completa en formato PDF. En ella debe indicarse, en la primera página y de forma clara, el nombre y apellidos de los integrantes del grupo.
 - Los archivos de código resultantes de la realización de los ejercicios solicitados:
 - Practica3.2_holamundo.html
 - Practica3.2_styles.css
 - Practica3.2_control_de_accesos.php

4. Enunciado de la práctica

En este apartado se describirán las distintas actividades, programas y ejercicios que deberá realizar cada grupo de prácticas, así como la información a completar y/o rellenar en la memoria. Recuerda entregar también los ficheros HTML y PHP resultantes de la realización de esta práctica.

4.1 Página Web con HTML

Ejercicio 8

En la segunda parte de la práctica, en primer lugar, vamos a montar nuestra primera página web con HTML. Podemos crear una página web sencilla, por ejemplo: Practica3.2_holamundo.html (véase la Tabla 1). Esta página, deberá insertarse en nuestro servidor web (concretamente en el directorio “/var/www/html” de Apache2).

```
<!DOCTYPE html>
<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <title>Hola Mundo</title>
  </head>
  <body>
    <p>¡Hola Mundo! </p>
  </body>
</html>
```

Tabla 1. Características solicitadas para la base de datos.

A continuación, se pide añadir una serie de elementos a la página web. No se pretende diseñar una web funcional (para ser colgada en un servidor y que todo el mundo tenga acceso a ella), el objetivo de esta práctica es utilizar algunas de las herramientas y opciones HTML más comunes.

RECUERDA ARRANCAR LOS SERVICIOS (Apache2 y MySQL) CADA VEZ QUE REINICIES LA MÁQUINA.

Pregunta 8.1. Crea la página web sencilla indicada en la Tabla 1 y verifica que funciona. Muestra un pantallazo al acceder a ella desde un navegador web.

Pregunta 8.2. Añade una tabla a la página web dentro de la etiqueta <body> (véase el ejemplo de la Tabla 2). En ella, deben incluirse tres filas y tres columnas. Las columnas corresponden a los campos “Profesor”, “Despacho” y “Edificio”. Las filas corresponden a la siguiente información:

- **Marta Beltrán | 122 | Departamental II**
- **Isaac Martín | 167 | Departamental II**
- **Miguel Calvo | S/N | Departamental II**

```
<table style="width:100%">
<tr>
  <th>Firstname</th>
  <th>Lastname</th>
  <th>Age</th>
</tr>
<tr>
  <td>Jill</td>
  <td>Smith</td>
  <td>50</td>
</tr>
```

```
<tr>
  <td>Eve</td>
  <td>Jackson</td>
  <td>94</td>
</tr>
</table>
```

Tabla 2. Ejemplo de tabla en HTML.

Pregunta 8.3. Investiga sobre las etiquetas “h” de HTML y añade una cabecera con el título “Listado de profesores” a la página web. ¿Qué línea HTML has tenido que añadir?

Pregunta 8.4. Cambia el estilo de la tabla utilizando el código CSS de la Tabla 3. ¿Qué has tenido que modificar o añadir en la tabla para que se muestre en color amarillo?

```
<style>
  table, th, td {
    border: 1px solid black;
    border-collapse: collapse;
  }
```



```

th, td {
    padding: 15px;
    text-align: left;
}
table#t01 {
    width: 100%;
    background-color: #f1f1c1;
}
</style>

```

Tabla 3. Ejemplo de código CSS para tabla en HTML.

Pregunta 8.5. Modifica el CSS que acabas de insertar para que el borde de la tabla (véase la Figura 1) se muestre del mismo color que tiene la corona del logotipo de la Universidad Rey Juan Carlos. Para descubrir el código de color puedes utilizar GIMP, la extensión para Google Chrome “ColorZilla” o alguna herramienta similar. ¿Qué línea has añadido y dónde para realizar este cambio? Haz un pantallazo del resultado.



Listado de profesores

Profesor	Despacho	Edificio
Marta	122	Depar II
Isaac	167	Depar II
Miguel	S/N	Depar II

Figura 1. Tabla HTML con CSS aplicado.

Pregunta 8.6. Añade un formulario a la web. Dicho formulario ha de solicitar un nombre y una contraseña (oculta cuando se escribe, sustituida por “*”). Si das al botón “Acceder” del formulario, la petición de acceso será enviada a una página llamada “Practica3.2_control_de_accesos.php” (esta página de momento no tendrá ninguna funcionalidad, se escribirá en próximos apartados de la práctica). Además, añade una cabecera al formulario que diga “Acceso alumnos”, (de tamaño menor que la anterior “Listado de Profesores”). El resultado debe ser similar al de la Figura 2. ¿Qué código has utilizado?



Acceso alumnos

Usuario:

Contraseña:

Figura 2. Formulario de Acceso HTML.

Pregunta 8.7. En este ejercicio, se debe añadir una lista a la web. Esta lista debe incluir una cabecera (de tamaño aún menor que la cabecera del formulario de acceso) de color azul que diga “Listado de prácticas”. Para ello, investiga sobre las etiquetas “ul” y “li” de HTML y lee la ayuda que te proporcionamos. El resultado debe ser similar al de la Figura 3. Pega el código utilizado para cambiar el color de la cabecera (tanto el CSS como la etiqueta HTML de la propia cabecera).

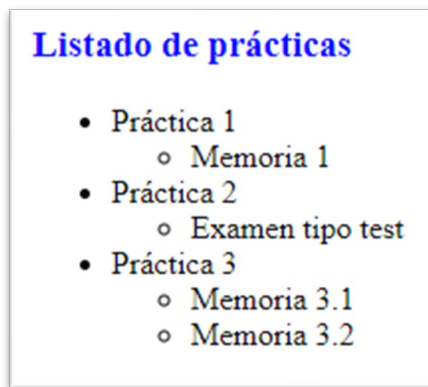


Figura 3. Formulario de Acceso HTML.

AYUDA:

- Una práctica habitual en CSS cuando se quiere cambiar el color de diferentes elementos en una web es añadir una nueva clase para ese. Por ejemplo, en el fichero CSS y/o en el apartado <style> del propio HTML se incluiría lo siguiente:

```
.nombre_del_color {  
    color: color_deseado;  
}
```

- Además, en el elemento sobre el que se quiere modificar el color, se debería indicar la clase, por ejemplo: `<p class="nombre_del_color"> Mi texto </p>`

Pregunta 8.8. Después de añadir el listado de elementos y buscando un resultado similar al de la Figura 4, añade un nuevo apartado en tu web con una cabecera que diga “Enlaces externos” y una sub-cabecera que diga “Enlaces de interés”. En este apartado, debes incluir un enlace a la web de la URJC, otro al de la ETSI y otro al Aula Virtual. El color de los enlaces debe ser (investiga sobre “CSS Links”):

- Verde cuando no haya sido visitado.
- Rosa cuando haya sido visitado.
- Subrayado y rojo cuando esté activo.
- Azul cuando se tenga el ratón encima.

¿Qué código CSS has utilizado? Adjunta un pantallazo del resultado.

Enlaces externos

Enlaces de interés

- [Página web de la Universidad Rey Juan Carlos](#)
- [Página web de la Escuela Técnica Superior de Informática](#)
- [Aula Virtual](#)

Figura 4. Enlaces en HTML.

Pregunta 8.9. Graba un audio con la herramienta de grabación de tu ordenador (por ejemplo, “Grabadora de voz” en Windows, “Audio Recorder” en Ubuntu, etc.). En este audio, todos los integrantes del grupo debéis presentaros (diciendo vuestro nombre, la carrera que estáis haciendo y el nombre de la asignatura). Después, incluye el audio dentro de la página web (con cabecera “Presentación”), apoyándote en la etiqueta “audio” de HTML5. El resultado debe ser similar al de la Figura 5. ¿Qué código has utilizado para incluir tu audio en el HTML?

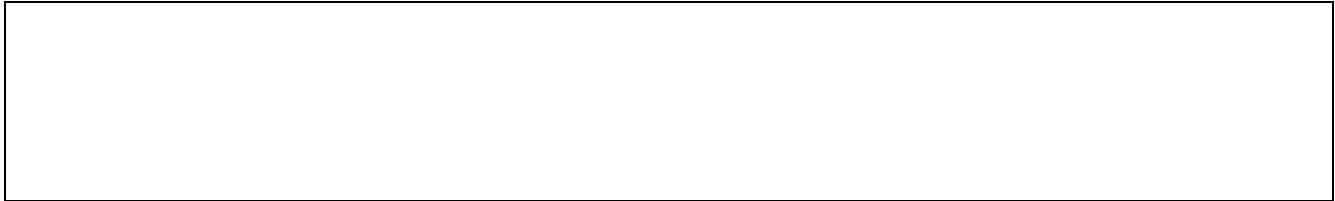


Figura 5. Audio en HTML.

Pregunta 8.10. Cuando los estilos se van complicando, como ha ocurrido en esta web tan sencilla ¿cómo suelen gestionarse? ¿Se dejan en el HTML como hemos hecho en esta primera página o se separan de los contenidos de alguna manera? Hazlo en tu web y llama al nuevo fichero “Practica3.2_styles.css”.



El resultado final de este apartado (“Página Web con HTML”), debería ser algo similar a lo que se muestra en la Figura 6.

Listado de profesores

Profesor	Despacho	Edificio
Marta	122	Depar II
Isaac	167	Depar II
Miguel	S/N	Depar II

Acceso alumnos

Usuario:

Contraseña:

Listado de prácticas

- Práctica 1
 - Memoria 1
- Práctica 2
 - Examen tipo test
- Práctica 3
 - Memoria 3.1
 - Memoria 3.2

Enlaces externos

Enlaces de interés

- [Página web de la Universidad Rey Juan Carlos](#)
- [Página web de la Escuela Técnica Superior de Informática](#)
- [Aula Virtual](#)

Presentación

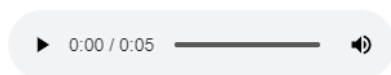


Figura 6. Resultado final del ejercicio “5.1 Página Web con HTML”.

4.2 Página Web con PHP

Ejercicio 9

En la primera parte de esta práctica (“Practica3.1_Guion.pdf”), se instaló toda la pila LAMP y, por ende, debería estar instalado PHP en la máquina virtual. Además, en el directorio “/var/www/html” debería existir un archivo llamado “info.php” (que también se creó en la primera parte de la práctica) y que, accediendo desde el navegador (<http://localhost/info.php>), debería mostrar algo similar a lo que puede observarse en la Figura 7.


<div> <div>PHP Version 7.4.21</div> <div>  </div> </div>	
System	Linux kali 5.10.0-kali9-686-pae #1 SMP Debian 5.10.46-1kali1 (2021-06-25) i686
Build Date	Jul 2 2021 03:59:48
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.4/apache2
Loaded Configuration File	/etc/php/7.4/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/7.4/apache2/conf.d
Additional .ini files parsed	/etc/php/7.4/apache2/conf.d/10-mysqld.ini, /etc/php/7.4/apache2/conf.d/10-opcache.ini, /etc/php/7.4/apache2/conf.d/10-pdo.ini, /etc/php/7.4/apache2/conf.d/15-xml.ini, /etc/php/7.4/apache2/conf.d/20-calendar.ini, /etc/php/7.4/apache2/conf.d/20-ctype.ini, /etc/php/7.4/apache2/conf.d/20-curl.ini, /etc/php/7.4/apache2/conf.d/20-dom.ini, /etc/php/7.4/apache2/conf.d/20-exif.ini, /etc/php/7.4/apache2/conf.d/20-ffi.ini, /etc/php/7.4/apache2/conf.d/20-fileinfo.ini, /etc/php/7.4/apache2/conf.d/20-ftp.ini, /etc/php/7.4/apache2/conf.d/20-gettext.ini, /etc/php/7.4/apache2/conf.d/20-iconv.ini, /etc/php/7.4/apache2/conf.d/20-json.ini, /etc/php/7.4/apache2/conf.d/20-mysqli.ini, /etc/php/7.4/apache2/conf.d/20-pdo_mysql.ini, /etc/php/7.4/apache2/conf.d/20-phar.ini, /etc/php/7.4/apache2/conf.d/20-posix.ini, /etc/php/7.4/apache2/conf.d/20-readline.ini, /etc/php/7.4/apache2/conf.d/20-shmop.ini, /etc/php/7.4/apache2/conf.d/20-simplexml.ini, /etc/php/7.4/apache2/conf.d/20-sockets.ini, /etc/php/7.4/apache2/conf.d/20-sysvmsg.ini, /etc/php/7.4/apache2/conf.d/20-sysvsem.ini, /etc/php/7.4/apache2/conf.d/20-sysvshm.ini, /etc/php/7.4/apache2/conf.d/20-tokenizer.ini, /etc/php/7.4/apache2/conf.d/20-xmlreader.ini, /etc/php/7.4/apache2/conf.d/20-xmlwriter.ini, /etc/php/7.4/apache2/conf.d/20-xsl.ini
PHP API	20190902
PHP Extension	20190902
Zend Extension	320190902
Zend Extension Build	API320190902.NTS
PHP Extension Build	API20190902.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
DTrace Support	available, disabled

Figura 7. Comprobación de la instalación de LAMP.

A continuación, se pide dotar de funcionalidad al formulario que has incluido antes en la página web HTML. Si recuerdas:

- En la primera parte de la práctica, has creado una base de datos llamada “accesos” con una tabla “user_pass” con campos “user” y “password”. Además, has añadido dos filas como mínimo a esta tabla.
- En esta segunda parte de la práctica has creado en tu página web HTML (Practica3.2_holamundo.html) un formulario para dar acceso a una zona privada que pide un campo “usuario” y un campo “contraseña” y que muestra un botón “Acceder”.

PHP nos permite conectar este formulario con la base de datos de manera que un usuario que se encuentre en la tabla y que escribe correctamente su nombre y su contraseña, tendrá éxito al dar al botón “Acceder”. De la misma forma, si un usuario no se encuentra en la tabla o no escribe correctamente su nombre y/o contraseña, se le devolverá un error.

Pregunta 9.1. ¿Qué hay que modificar en el código HTML de la página “Practica3.2_holamundo.html” para que al dar al botón “Acceder” del formulario se procese el control de accesos (“Practica3.2_control_de_accesos.php”) contra la base de datos de usuarios y contraseñas que tenemos en MySQL?

Pregunta 9.2. ¿Cómo debe programarse la página “Practica3.2_control_de_accesos.php”? Escribe completando el esqueleto de la Tabla 4 y comprueba que funciona correctamente en todos los casos posibles.

NOTA: No te limites a copiar el código y rellenar los huecos, intenta entenderlo, comentarlo, mejorar la gestión de los errores, etc. No hace falta programar la página web para la zona privada, con los mensajes que se muestra en los “echo” es suficiente.

```
<?php

if (isset($_POST['XXX1']) && isset($_POST['XXX2']) && !empty($_POST['XXX1']) && !empty($_POST['XXX2'])) {

    $user = $_POST['XXX1'];
    $pass = $_POST['XXX2'];

    $conn = new mysqli('localhost', 'app', 'XXX3', 'XXX4');

    if ($conn->connect_error) {
        die("Falló la conexión a BBDD: " . $conn->connect_error);
    }

    $resultado = $conn->query("SELECT * FROM XXX5 WHERE XXX6 AND XXX7;");

    if ($resultado->XXX8){
        echo 'Usuario encontrado: Puedes acceder a la zona privada';
    } else {
        echo 'Usuario no encontrado: Vuelve a intentarlo';
    }

    XXX9->close();
} else {
    echo 'Introduce un usuario y una contraseña.';
}

?>
```

Tabla 4. Estructura del código del fichero “Practica3.2_control_de_accesos.php”.

AYUDA:

- Sin necesitas mostrar los errores que tienes en PHP mientras desarrollas, puedes utilizar, colocándolo al inicio del código, las siguientes líneas:

```
ini_set('display_errors', 1);
```

```
ini_set('display_startup_errors', 1);
```

```
error_reporting(E_ALL);
```

- Ten en cuenta que en una página web en producción, esos errores JAMÁS deberían mostrarse, ya que pueden ser utilizados por los atacantes para descubrir fallos en nuestras aplicaciones.

4.3 Inyección SQL

Ejercicio 10

Tal y como se ha desarrollado y escrito la página “Practica3.2_control_de_accesos.php”, se está confiando en la entrada que viene desde el lado del usuario. Directamente, los datos que el usuario introduce en el formulario son los que se utilizan para construir la consulta que se lanza a la base de datos (recuperados mediante “\$_POST['XXXXX']”), lo cual hace nuestra aplicación completamente insegura.

Pregunta 10.1. Sabiendo que el adversario no tiene cuenta en nuestra aplicación y, por lo tanto, no está en la base de datos, ¿Qué podría escribir en el campo del formulario de la contraseña para ganar acceso a la zona privada? ¿Qué daría siempre un resultado TRUE al lanzar la consulta SQL contra la base de datos? Este patrón de ataque se denomina inyección SQL.

AYUDA:

- Piensa en operadores lógicos como el AND o el OR y en cómo afectarían al resultado de la consulta si se inyectaran en alguna posición concreta.
- Como eres el desarrollador de “Practica3.2_control_de_accesos.php” y estás empezando, puedes ayudarte con la inyección SQL haciendo un “echo” en PHP de la consulta que se lanzaría a la base de datos. De esta forma, podrás ver el resultado de introducir diferentes parámetros a través del formulario y comprobar si la consulta está bien formada o no.

Pregunta 10.2. ¿Cómo tendría que mejorarse la página “Practica3.2_control_de_accesos.php” para evitar este tipo de inyecciones? Propón al menos dos alternativas y explícalas (no hace falta que las implementes).