

Técnicas para proteger los correos electrónicos en nuestra web

Sumario

- 1 Proteger los correos electrónicos contra SPAMBOTS
 - ♦ 1.1 Protección de e-mails utilizando JavaScript
 - ♦ 1.2 Sustitución de caracteres
 - ♦ 1.3 Ofuscar el correo electrónico
- 2 Generar imágenes de forma dinámica con PHP
 - ♦ 2.1 Generación de una imagen CAPTCHA
 - ♦ 2.2 Generación imágenes partiendo de una imagen base

Proteger los correos electrónicos contra SPAMBOTS

Todos conocemos lo que es el spam (envío masivo de correo no deseado). Pues bien, los spambots (robots de spam) que se dedican a la búsqueda de correos, lo que hacen es descargarse las páginas web y buscar e-mails dentro del código fuente de la página.

Una forma de proteger los e-mails dentro del código fuente, para evitar que capturen las direcciones de correo, consiste en codificar esas direcciones de forma que no aparezcan en el código fuente de la página web. Veamos algunas técnicas:

Protección de e-mails utilizando JavaScript

```
<script type="text/javascript">
    direccion=('tunombre@' + 'dominio.com');
    document.write('<a href="mailto:' + direccion + '">' + direccion + '</a>');
</script>

<noscript>
    <em>Direccion de Email protegido por Javascript.<br/>
    Por favor habilita JavaScript para ver la dirección.</em>
</noscript>
```

Sustitución de caracteres

Otra forma más sencilla pero menos efectiva puede consistir en **sustituir** la @ por "en" y el . por "punto".

Ejemplo:

```
E-mail: marta en terra punto net
```

Ofuscar el correo electrónico

Consiste en codificar los caracteres del correo por su equivalente en ASCII. Por ejemplo si ofuscamos la dirección **usuario@dominio.com** quedaría como:

```
<a href="%x6d;%x97;i1%#116;%x6f;:%x75;%x73;u%#97;%#114;i%#x6f;@%#100;om%#x69;n%#105;o.%#x63;%#x6f;%#109;">
usuario@dominio.com</a>
```

En la siguiente dirección tenéis el código fuente en JavaScript para ofuscar el código:

<http://www.elcodigo.net/cgi-bin/DBread.cgi?tabla=scripts&campo=0&clave=154&info=4>

Generar imágenes de forma dinámica con PHP

Mediante esta técnica más avanzada se pueden proteger también los e-mails ya que éste no va aparecer nunca en el código fuente de la página web. En su lugar aparecerá una imagen con la información deseada.

Para poder generar imágenes de forma dinámica con PHP, necesitaremos tener instalada la extensión **PHP-GD2**.

Generación de una imagen CAPTCHA

El siguiente código genera una imagen Captcha en formato JPG lista para ser utilizada en nuestra web.

Ejemplo de uso:

```
<img src='captcha.php' />
```

Contenido del fichero **captcha.php**:

```
<?php
// Ancho y alto de la imagen en pixels.
$ancho = 100;
$alto = 30;
$imagen = imagecreate($ancho, $alto);

// Definimos el color amarillo.
$amarillo = imagecolorallocate($imagen, 255, 255, 0);

// Rellenamos la imagen de color amarillo.
ImageFill($imagen, 0, 0, $amarillo);
$colorTexto = imagecolorallocate($imagen, 28, 63, 108);
$colorLineas = imagecolorallocate($imagen, 255, 0, 0);
$valoraleatorio = rand(100000, 999999);

// Escribimos el valor aleatorio en la imagen.
ImageString($imagen, 5, 25, 5, $valoraleatorio, $colorTexto);
for ($c = 0; $c <= 5; $c++)
{
    $x1 = rand(0, $ancho);
    $y1 = rand(0, $alto);
    $x2 = rand(0, $ancho);
    $y2 = rand(0, $alto);
    ImageLine($imagen, $x1, $y1, $x2, $y2, $colorLineas);
}

// Enviamos la cabecera de imagen JPEG.
// Importantísimo no tener ningún espacio en blanco al principio del fichero PHP
// ni hacer ningún echo o similar, por que entonces el envío de la cabecera fallaría.
header("Content-type: image/jpeg");

// Enviamos el contenido de la imagen.
imagejpeg($imagen);

// Destruimos la memoria ocupada por la imagen en el servidor.
imagedestroy($imagen);
?>
```

Generación imágenes partiendo de una imagen base

En este ejemplo vamos a crear una imagen nueva partiendo de una imagen base, a la que le añadimos el texto deseado.

Ejemplo de utilización:

```
<img src='email.php' />
```

Contenido de fichero **email.php**:

```
<?php
# Texto a superponer en la imagen.
$texto = "E-mail: info@dominio.com";
```

```

# Cargamos la imagen de fondo.
$imagen = imagecreatefrompng("img/email.png");

#Color del texto
$color = imagecolorallocate($imagen, 19, 107, 216);

#Calculamos la distancia para centrar el texto.
$px = (imagesx($imagen) - 7.5 * strlen($texto)) / 2;

// Situamos el texto en la imagen.
imagestring($imagen, 4, $px, 190, $texto, $color);

// En caso de tener transparencias, almacenar el canal Alpha completo.
imagesavealpha($imagen,true);

// Enviamos la cabecera de imagen PNG.
// Importantísimo no tener ningún espacio en blanco al principio del fichero PHP
// ni hacer ningún echo o similar, por que entonces el envío de la cabecera fallaría.
Header ("Content-type: image/png");

// Si queremos que almacene la imagen generada, haríamos:
// imagepng($imagen, "fotografia.png");

// Enviamos el contenido de la imagen.
imagepng($imagen);

// Destruimos la imagen en el servidor.
imagedestroy($imagen);
?>

```

--Veiga (discusión) 22:05 21 may 2013 (CEST)