

PHP Form image verification

When you build a form in your web page, you are susceptible to being spamed by automated systems. In order to make sure that the one who completes the form is human, you can use the system with image verification. You can see here a sample of form with a random text image verification.

Requirements :

You will need PHP and GD Graphics Library installed. To see if you have gd installed on your server, create a file named info.php, please the following content in it: <?php phpinfo(); ?>

It should appear a page with infos and you can find if you have gd installed or not.

A few rules :

1. Place an info message to instruct the user what to do.
2. Do not use too many letters! You want to secure your form, not to annoy the user.
3. A hack to this security measure is using OCR systems to "read" the text image. Use different types of fonts, rotate them in different angles and use a background image to prevent that.

Next is presented the code explained.

The code explained:

Create a php file random_image_sample.php. This file will generate the an image with a random 5 letters text.

```
//random_image_sample.php contents

//start a session
session_start();
//this function is called recursively
function random_string($len=5, $str="")
{
    for($i=1; $i<=$len; $i++)
    {
        //generates a random number that will be the ASCII code of the character.
        //We only want numbers (ascii code from 48 to 57) and caps letters.
        $ord=rand(48, 90);
        if(($ord >= 48) && ($ord <= 57) || (($ord >= 65) && ($ord <= 90)))
        $str.=chr($ord);
        //If the number is not good we generate another one
        else
        $str.=random_string(1);
    }
    return $str;
}

//create the random string using the upper function
//if you want more than 5 characters just modify the parameter)
$rand_str=random_string(5);
```

```
//We memorize the md5 sum of the string into a session variable
$_SESSION['image_value'] = md5($rand_str);
```

```

//Get each letter in one variable, we will format all letters different
$letter1=substr($rand_str,0,1);
$letter2=substr($rand_str,1,1);
$letter3=substr($rand_str,2,1);
$letter4=substr($rand_str,3,1);
$letter5=substr($rand_str,4,1);

//Creates an image from a png file. If you want to use gif or jpg images,
//just use the coresponding functions: imagecreatefromjpeg and imagecreatefromgif.
$image=imagecreatefrompng("images/noise.png");

//Get a random angle for each letter to be rotated with.
$angle1 = rand(-20, 20);
$angle2 = rand(-20, 20);
$angle3 = rand(-20, 20);
$angle4 = rand(-20, 20);
$angle5 = rand(-20, 20);

//Get a random font. (In this examples, the fonts are located in "fonts" directory and named from 1.ttf to 10.ttf)
$font1 = "fonts/".rand(1, 10).".ttf";
$font2 = "fonts/".rand(1, 10).".ttf";
$font3 = "fonts/".rand(1, 10).".ttf";
$font4 = "fonts/".rand(1, 10).".ttf";
$font5 = "fonts/".rand(1, 10).".ttf";

//Define a table with colors (the values are the RGB components for each color).
$colors[0]=array(122,229,112);
$colors[1]=array(85,178,85);
$colors[2]=array(226,108,97);
$colors[3]=array(141,214,210);
$colors[4]=array(214,141,205);
$colors[5]=array(100,138,204);

//Get a random color for each letter.
$color1=rand(0, 5);
$color2=rand(0, 5);
$color3=rand(0, 5);
$color4=rand(0, 5);
$color5=rand(0, 5);

//Allocate colors for letters.
$textColor1 = imagecolorallocate ($image, $colors[$color1][0],$colors[$color1][1], $colors[$color1][2]);
$textColor2 = imagecolorallocate ($image, $colors[$color2][0],$colors[$color2][1], $colors[$color2][2]);
$textColor3 = imagecolorallocate ($image, $colors[$color3][0],$colors[$color3][1], $colors[$color3][2]);
$textColor4 = imagecolorallocate ($image, $colors[$color4][0],$colors[$color4][1], $colors[$color4][2]);
$textColor4 = imagecolorallocate ($image, $colors[$color5][0],$colors[$color5][1], $colors[$color5][2]);

Write text to the image using TrueType fonts.
$size = 20;
imaggottext($image, $size, $angle1, 10, $size+15, $textColor1, $font1, $letter1);
imaggottext($image, $size, $angle2, 35, $size+15, $textColor2, $font2, $letter2);
imaggottext($image, $size, $angle3, 60, $size+15, $textColor3, $font3, $letter3);
imaggottext($image, $size, $angle4, 85, $size+15, $textColor4, $font4, $letter4);
imaggottext($image, $size, $angle5, 110, $size+15, $textColor5, $font5, $letter5);

header('Content-type: image/jpeg');
Output image to browser
imagejpeg($image);
Destroys the image
imagedestroy($image);

```

The html file `random_sample.html`

is a sample of how to use the random image.

```

<?php
session_start();
?>
<HTML>
<HEAD>
<TITLE>Random image sample</TITLE>
<META HTTP-EQUIV=\\"Content-Type\"
CONTENT=\\"text/html;
charset=iso-8859-1\">
</HEAD>
<BODY>
<?php
$errors=0;
if(isset($_POST['Submit']))
{
$number = $_POST['number'];
if(md5($number) != $_SESSION['image_value'])
echo '<h1>Validation string not valid!
Please try again!</h1>';
else
{
echo '<h1>Your string is valid!</h1>';
//do what you want with
the data
}
}
?>
<form name="form1" method="post"
action="random_sample.php">
<table cellspacing="0" width="600"
align="center">
<tr><td valign="top"
align="right">Comments</td>
<td><input name="message"
size=40 value="...your message here">
</td></tr>
<tr><td colspan=2 align="center">
<font size="1" face="Geneva,
Arial, Helvetica, sans-
serif"><strong><font
size="2">
Please enter the string shown in the image
in the form.<br> The
possible characters
are letters from A to Z in capitalized form
and the
numbers from 0 to 9.
</font></td></tr>
<tr><td align="center"
colspan=2><input name="number"
type="text" id=\\"number\"></td></tr>
<tr><td colspan=2 align="center"></td></tr>
<tr><td colspan=2 align="center"><input
value="Submit"></td></tr>
</table>
</form>
</BODY>
</HTML>

```

View [demo here](#) If you found this tutorial useful you might be interested to read the following:

- [Installing Apache, PHP, MySQL on a Windows 98/2000/XP Computer](#)
- [PHP Create Image Thumbnail](#)
- [Count Words in a string using PHP](#)
- [PHP Form Image Upload](#)
- [PHP print random image from a folder](#)