

[CODE](#) > [PHP](#)

# Build Your Own CAPTCHA and Contact Form in PHP

by [Monty Shokeen](#) 8 Jan 2019Difficulty: Intermediate Length: Medium Languages: English ▼

PHP

Web Development

Captcha



People write code every day to automate a variety of processes. We exploit the fact that computers are a lot faster and more accurate than humans, which lets us simplify a lot of mundane tasks. Unfortunately, these same abilities can be used to program computers to do something malicious like sending spam or guessing passwords. The focus of this tutorial will be on combating spam.

Let's say you have a website with a contact form to make it easy for visitors to contact you. All they have to do is fill out a form and hit the send button to let you know about a problem or request they have. This is an important feature of a public-facing website, but the process of filling out form values can be automated by malicious users to send a lot of spam your way. This type of spamming technique is not limited to just contact forms. Bots can also be used to fill your forums with spam posts or comments that link to harmful websites.

One way to solve this problem is to devise a test which can distinguish between bots which are trying to spread spam and people who legitimately want to contact you. This is where CAPTCHAs come in. They generally consist of images with a random combination of five or six letters written on a colored background. The idea is that a human will be able to read the text inside the image, but a bot won't. Checking the user-filled CAPTCHA value against the original can help you distinguish bots from humans. CAPTCHA stands for "completely automated public Turing test to tell computers and humans apart".

In this tutorial, we will learn how to create our own CAPTCHAs and then integrate them with the contact form we created in the tutorial.

## Creating the CAPTCHA

We will use the PHP GD library to create our CAPTCHA. You can learn more about [writing text and drawing shapes with GD](#) in one of my earlier tutorials. We will also have to write a little bit of code to create our random string to be written on the image that's created. Yet another tutorial, titled [Generating Random Alphanumeric Strings in PHP](#), can help us in this regard.

### Generate a Random String

All the code from this section will go in the **captcha.php** file. Let's begin by writing the function to create the random string.

```
01 <?php
02
03 $permitted_chars = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ';
04
05 function generate_string($input, $strength = 5) {
06     $input_length = strlen($input);
07     $random_string = '';
08     for($i = 0; $i < $strength; $i++) {
09         $random_character = $input[mt_rand(0, $input_length - 1)];
10         $random_string .= $random_character;
11     }
12
13     return $random_string;
14 }
15
16 $string_length = 6;
17 $captcha_string = generate_string($permitted_chars, $string_length);
18
19
20 ?>
```

The `$permitted_chars` variable stores all the characters that we want to use to generate our CAPTCHA string. We are only using capital letters in the English alphabet to avoid any confusion that might arise due to letters or numbers that might look alike. You can use any set of characters that you like to increase or decrease the difficulty of the CAPTCHA.

Our function creates a five-letter string by default, but you can change that value by passing a different parameter to the `generate_string()` function.

Advertisement

### Render the CAPTCHA Background

Once we have our random string, it's time to write the code to create the background of the CAPTCHA image. The image will be 200 x 50 pixels in size and will use five different colors for

the background.

```

01 <?php
02
03 $image = imagecreatetruecolor(200, 50);
04
05 imageantialias($image, true);
06
07 $colors = [];
08
09 $red = rand(125, 175);
10 $green = rand(125, 175);
11 $blue = rand(125, 175);
12
13 for($i = 0; $i < 5; $i++) {
14     $colors[] = imagecolorallocate($image, $red - 20*$i, $green - 20*$i, $blue - 20*$i);
15 }
16
17 imagefill($image, 0, 0, $colors[0]);
18
19 for($i = 0; $i < 10; $i++) {
20     imagesetthickness($image, rand(2, 10));
21     $rect_color = $colors[rand(1, 4)];
22     imagerectangle($image, rand(-10, 190), rand(-10, 10), rand(-10, 190), rand(40, 60), $rect_color);
23 }
24
25 ?>

```

We begin with random values for the variables `$red`, `$green`, and `$blue`. These values determine the final color of the image background. After that, we run a `for` loop to create progressively darker shades of the original color. These colors are stored in an array. The lightest color is the first element of our `$colors` array, and the darkest color is the last element. The lightest color is used to fill the whole background of the image.

In the next step, we use a `for` loop to draw rectangles at random locations on our original image. The thickness of the rectangles varies between 2 and 10, while the color is chosen randomly from the last four values of our `$colors` array.

Drawing all these rectangles adds more colors to the background, making it a little harder to distinguish the foreground of the CAPTCHA string from the background of the image.

Your CAPTCHA background should now look similar to the following image.



## Render the CAPTCHA String

For the final step, we just have to draw the CAPTCHA string on our background. The color, y-coordinate, and rotation of individual letters is determined randomly to make the CAPTCHA string harder to read.

```

01  <?php
02
03  $black = imagecolorallocate($image, 0, 0, 0);
04  $white = imagecolorallocate($image, 255, 255, 255);
05  $textcolors = [$black, $white];
06
07  $fonts = [dirname(__FILE__).'\fonts\Acme.ttf', dirname(__FILE__).'\fonts\Ubuntu.ttf', dirname(
08
09  $string_length = 6;
10  $captcha_string = generate_string($permitted_chars, $string_length);
11
12  for($i = 0; $i < $string_length; $i++) {
13      $letter_space = 170/$string_length;
14      $initial = 15;
15
16      imagettftext($image, 20, rand(-15, 15), $initial + $i*$letter_space, rand(20, 40), $textcolor
17  }
18
19  header('Content-type: image/png');
20  imagepng($image);
21  imagedestroy($image);
22
23  ?>

```

As you can see, I'm using some fonts I downloaded from Google to get variation in the characters. There is a padding of 15 pixels on both sides of the image. The leftover space—170 pixels—is divided equally among all the CAPTCHA letters.

After rendering the text string above the background, your result should look similar to the image below. The characters will be different, but they should be slightly rotated and a mix of black and white.



## Adding the CAPTCHA to Our Contact Form

Now that we have created our CAPTCHA, it's time to add it to our contact form. We will use the contact form from [my previous tutorial on how to create a PHP contact form](#) and add the CAPTCHA just above the **Send Message** button.

We will be using sessions to store the CAPTCHA text and then validating the text entered by website visitors. Here is the complete code of our **captcha.php** file:

```

01 <?php
02
03 session_start();
04
05 $permitted_chars = 'ABCDEFGHJKLMNPQRSTUVWXYZ';
06
07 function generate_string($input, $strength = 10) {
08     $input_length = strlen($input);
09     $random_string = '';
10     for($i = 0; $i < $strength; $i++) {
11         $random_character = $input[mt_rand(0, $input_length - 1)];
12         $random_string .= $random_character;
13     }
14
15     return $random_string;
16 }
17
18 $image = imagecreatetruecolor(200, 50);
19
20 imageantialias($image, true);
21
22 $colors = [];
23
24 $red = rand(125, 175);
25 $green = rand(125, 175);
26 $blue = rand(125, 175);
27
28 for($i = 0; $i < 5; $i++) {
29     $colors[] = imagecolorallocate($image, $red - 20*$i, $green - 20*$i, $blue - 20*$i);
30 }
31
32 imagefill($image, 0, 0, $colors[0]);
33
34 for($i = 0; $i < 10; $i++) {
35     imagesetthickness($image, rand(2, 10));
36     $line_color = $colors[rand(1, 4)];
37     imagerectangle($image, rand(-10, 190), rand(-10, 10), rand(-10, 190), rand(40, 60), $line_color);
38 }
39
40 $black = imagecolorallocate($image, 0, 0, 0);
41 $white = imagecolorallocate($image, 255, 255, 255);
42 $textcolors = [$black, $white];
43
44 $fonts = [dirname(__FILE__).'\fonts\Acme.ttf', dirname(__FILE__).'\fonts\Ubuntu.ttf', dirname(__FILE__).'\fonts\Verdana.ttf'];
45
46 $string_length = 6;
47 $captcha_string = generate_string($permitted_chars, $string_length);
48
49 $_SESSION['captcha_text'] = $captcha_string;
50
51 for($i = 0; $i < $string_length; $i++) {
52     $letter_space = 170/$string_length;
53     $initial = 15;
54
55     imagettftext($image, 24, rand(-15, 15), $initial + $i*$letter_space, rand(25, 45), $textcolors[rand(0, 1)], $captcha_string[$i]);
56 }
57
58 header('Content-type: image/png');
59 imagepng($image);
60 imagedestroy($image);
61 ?>

```

The fonts that you want to use will go into the **fonts** directory. Now, you simply have to add the following HTML code above the **Send Message** button from our previous tutorial on [creating a contact form in HTML and PHP](#).

```
1 <div class="elem-group">
2   <label for="captcha">Please Enter the Captcha Text</label>
3   <i class="fas fa-redo refresh-ca
4   <br>
5   <input type="text" id="captcha" name="captcha_challenge" pattern="[A-Z]{6}">
6 </div>
```

Sometimes, the CAPTCHA text will be hard to read even for humans. In these situations, we want them to be able to ask for a new CAPTCHA in a user-friendly manner. The redo icon above helps us do exactly that. All you have to do is add the JavaScript below on the same page as the HTML for the contact form.

```
1 var refreshButton = document.querySelector(".refresh-captcha");
2 refreshButton.onclick = function() {
3   document.querySelector(".captcha-image").src = 'captcha.php?' + Date.now();
4 }
```

After integrating the CAPTCHA in the form and adding a refresh button, you should get a form that looks like the image below.

## Reason For Contacting Us

## Write your message

## Please Enter the Captcha Text





The final step in our integration of the CAPTCHA we created with the contact form involves checking the CAPTCHA value input by users when filling out the form and matching it with the value stored in the session. Update the **contact.php** file from the previous tutorial to have the following code.

```

01  <?php
02
03  session_start();
04
05  if($_POST) {
06      $visitor_name = "";
07      $visitor_email = "";
08      $email_title = "";
09      $concerned_department = "";
10      $visitor_message = "";
11
12      if(isset($_POST['captcha_challenge']) && $_POST['captcha_challenge'] == $_SESSION['captcha_t
13
14          if(isset($_POST['visitor_name'])) {
15              $visitor_name = filter_var($_POST['visitor_name'], FILTER_SANITIZE_STRING);
16          }
17
18          if(isset($_POST['visitor_email'])) {
19              $visitor_email = str_replace(array("\r", "\n", "%0a", "%0d"), '', $_POST['visitor_em
20              $visitor_email = filter_var($visitor_email, FILTER_VALIDATE_EMAIL);

```

```

21
22     }
23
24     if(isset($_POST['email_title'])) {
25         $email_title = filter_var($_POST['email_title'], FILTER_SANITIZE_STRING);
26     }
27
28     if(isset($_POST['concerned_department'])) {
29         $concerned_department = filter_var($_POST['concerned_department'], FILTER_SANITIZE_S
30     }
31
32     if(isset($_POST['visitor_message'])) {
33         $visitor_message = htmlspecialchars($_POST['visitor_message']);
34     }
35
36     if($concerned_department == "billing") {
37         $recipient = "billing@domain.com";
38     }
39     else if($concerned_department == "marketing") {
40         $recipient = "marketing@domain.com";
41     }
42     else if($concerned_department == "technical support") {
43         $recipient = "tech.support@domain.com";
44     }
45     else {
46         $recipient = "contact@domain.com";
47     }
48
49     $headers = 'MIME-Version: 1.0' . "\r\n"
50     . 'Content-type: text/html; charset=utf-8' . "\r\n"
51     . 'From: ' . $visitor_email . "\r\n";
52
53     if(mail($recipient, $email_title, $visitor_message, $headers)) {
54         echo '<p>Thank you for contacting us. You will get a reply within 24 hours.</p>';
55     } else {
56         echo '<p>We are sorry but the email did not go through.</p>';
57     }
58 } else {
59     echo '<p>You entered an incorrect Captcha.</p>';
60 }
61
62 } else {
63     echo '<p>Something went wrong</p>';
64 }
65
66 ?>

```



We updated this file to first check if the CAPTCHA value stored in the session is the same as the value input by the user. If they are different, we simply tell the visitors that they entered an incorrect CAPTCHA. You can handle the situation differently based on what your project needs.

Advertisement



## Final Thoughts

In this tutorial, we created our own CAPTCHA in PHP from scratch and integrated it with a [PHP contact form](#) we built in one of our earlier tutorials. We also made the CAPTCHA more user-friendly by adding a refresh button so that users get a new string with a new background in case the previous one was unreadable.

You can also use the logic from this tutorial to create a CAPTCHA that relies on solving basic mathematical equations like addition and subtraction.

If you want to add a CAPTCHA to your website, you should check out some of the form and CAPTCHA plugins available from CodeCanyon. Some of these have CAPTCHA and many other features like a file uploader built in.

If you have any questions or suggestions, feel free to let me know in the comments. You should also take a look at this list of [best PHP contact forms](#).



PHP

### 12 Best Contact Form PHP Scripts

Nona Blackman



PHP

### Create a Contact Form in PHP

Monty Shokeen

Advertisement



## Monty Shokeen

I am a full-stack developer who also loves to write tutorials in his free time. Other than that, I love learning about new and interesting JavaScript libraries.

 FEED  LIKE  FOLLOW

### Weekly email summary

Subscribe below and we'll send you a weekly email summary of all new Code tutorials. Never miss out on learning about the next big thing.

Update me weekly

### Translations

Envato Tuts+ tutorials are translated into other languages by our community members—you can be involved too!

Translate this post

Powered by



Advertisement

184 Comments

Nettuts+

 Login ▾ Recommend Tweet Share

Sort by Best ▾



Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS **Alpha Wolf** • 7 years ago

Doesn't work. I created a fonts folder and moved arial.ttf to it, but I get told that your image cannot be display because it contains image. I have everything done exactly as said to do. A fix would be nice.

&lt;?php

session\_start();

\$string = &amp;#039&amp;#039;

for (\$i = 0; \$i

11 ^ | ▾ 1 • Reply • Share ›

**sadpanda** ➔ Alpha Wolf • 7 years ago

the captcha image works for, but it doesn't validate for some reason

^ | ▾ • Reply • Share ›

**CodeFinity** • 6 years ago



One reason that the captcha may not be displaying correctly for some folks (might just show a broken img) is b/c if following these directions, the random text string is not implemented into \$image correctly. Why? B/c in Step 2, we have: "\$\_SESSION['random\_code'] = \$string; " Then, in Step 4, we have: "imagettftext(\$image, 30, 0, 10, 40, \$alt, \$dir."arial.ttf", \$\_SESSION['rand\_code']);" Notice that the name of the session variable is inconsistent. Hope this helps!

1 ^ | v • Reply • Share ›



**azeez** → CodeFinity • 2 years ago

Thanks so much

^ | v • Reply • Share ›



**Kul Rain Rine** • 7 years ago

blank image shown

1 ^ | v • Reply • Share ›



**David Moreen** • 10 years ago

Even though this captcha does not seem to secure, I still believe that this is a wonderful starting point.

1 ^ | v • Reply • Share ›



**David Moreen** → David Moreen • 10 years ago

After some time, I checked back at the script and made a few changes because this version was not working. Here is what I came up with:

```
<?php
session_start();
```

```
$string = "";
```

```
for ($i = 0; $i
```

1 ^ | v 1 • Reply • Share ›



**David Moreen** → David Moreen • 10 years ago

Darn, the comment got striped.

^ | v • Reply • Share ›



**Kev** • 8 years ago

Not bad but one problem. It is not responsive design.

Check out our Responsive Contact Forms With Image and Audio Security Captcha, six to choose from.

<http://templatiz.co/contact-...>

2 ^ | v 2 • Reply • Share ›



**Stuart Norman** → Kev • 6 years ago

No good if you have to pay for the scripts. I believe a working captcha should be free for all to use.

2 ^ | v • Reply • Share ›



**bad** → Kev • 7 years ago

your website is horrible

2 ^ | v • Reply • Share ›

**Paul** → Kev • 7 years ago

Not bad, but one problem. They're not free,

^ | v • Reply • Share ›

**nicob** → Kev • 7 years ago

Your forms may be responsive but your website...well...a mother cat wont find her kittens in it

^ | v • Reply • Share ›

**Kev** → nicob • 7 years ago

That's a good one. I have redesigned my site now and things are much easier to find.

^ | v • Reply • Share ›

**giveup** → Kev • 6 years ago

still terrible

1 ^ | v • Reply • Share ›

**SUMER CHOUDHARY** • 5 days ago

not work this captcha bed

^ | v • Reply • Share ›

**case the man** • 8 months ago

thanks for this wonderful code. i never get it to work though :( i had to change th last part to:

```
=====
ob_start();
imagepng($image);
$contents = ob_get_contents();
ob_end_clean();
imagedestroy($image);

ob_clean();
header('Content-Type: image/png');
echo $contents;
=====
```

and then it worked nicely.

Thanks!

^ | v • Reply • Share ›

**All PHP Tricks** • 2 years ago

Zac this is indeed quite helpful tutorial, easy to understand and implement, i also wrote a tutorial about creating a captcha using php, i hope you will also like this tutorial.

<https://www.allphptricks.co...>

^ | v • Reply • Share ›

**archana sahu** • 2 years ago

how to create font ??bcz my captcha image was not displaying....:(

^ | v • Reply • Share ›

**harpreet kaur** • 4 years agoI want to use captcha for my [websitehttp://immigrationco](http://immigrationco) i have try to



I want to use captcha for my [website http://mimigration100...](http://mimigration100...), I have try to install some plugin also but i am not able to use it properly, Is there anybody suggest me what I do...

^ | v • Reply • Share ›



**Alexey Panin** • 4 years ago • edited

For those who can't get it work. Create file index.php with any html and paste ``. It worked for me.

^ | v • Reply • Share ›



**Anna James Proforbes** • 4 years ago

i want to use it on my client's website <http://www.proforbes.com> contact page. but i don't know how it'll happen my client want to use their own strings in captcha. I don't understood how it'll be done. Don't know its easy or tough because i am a designer but its must to do. Please help if any one can...

thanks in advance

^ | v • Reply • Share ›



**raj** • 4 years ago

thanks for this tutorial very helpfull there is another tutorial which i found helpfull is on [www.talkerscode.com](http://www.talkerscode.com) based on how to create you own captcha system using ajax and php <http://talkerscode.com/webt...>

^ | v • Reply • Share ›



**eldrift** • 5 years ago

Worked like a charm the first time. Used strtoupper to make it more readable.

^ | v • Reply • Share ›



**Mohsen** • 6 years ago

TNX for this useful tutorial.

I have made it more beautiful. the code is as below:

```
<php
session_start();

$string = "";

$dir = 'fonts/';

$image = imagecreatetruecolor(180, 60);
$colors = array(imagecolorallocate($image, 0, 0, 0),imagecolorallocate($image, 200, 100, 90),imagecolorallocate($image, 10, 120, 10),imagecolorallocate($image, 10, 100, 210));
$lights = array(imagecolorallocate($image, 210, 250, 250),imagecolorallocate($image, 250, 220, 250),imagecolorallocate($image, 220, 200, 240),imagecolorallocate($image, 140, 150, 240));
$white = imagecolorallocate($image, 255, 255, 255);

imagefilledrectangle($image,0,0,399,99,$white);

for($i=0; $i<30; $i++){
    $x1 = rand(0,170);
    $y1 = rand(0,50);
```

```

imagefilledrectangle($image,$x1,$y1,$x1+10,$y1+10,$lights[rand(0,count(
}

for($i=$x1=0; $i<6; $i++){
    $x2 = $x1;
    $char =
    substr(str_shuffle('ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz0123456789'),
    $string .= $char;
    $x1 = ($i==0)?20:$i*20+($i*5);
    if($x1-$x2<10)
    $x1 += 20;
    imagefttext ($image, rand(22,30), rand(0,20), $x1, rand(30,40),
    $colors[rand(0,count($colors)-1)], $dir."courbd.ttf", $char);
}

$_SESSION['cap_code']=$string;

header("Content-type: image/png");
imagepng($image);
?>

```

^ | v • Reply • Share ›



**Mohsen** • 6 years ago

TNX for this useful tutorial.

I have made it more beautiful. the code is as below:

^ | v • Reply • Share ›



**Stuart Norman** • 6 years ago

Wonderful tutorial. It works like a charm. I've adapted it a little to include several different fonts to make it more harder. I'm not sure how well it'll work against spam bots though. I guess we'll see. I only have one question.

How would I implement a captcha refresh button, like the one on Google's reCaptcha. So that if the image is too hard for a user, they can refresh the image. Is this possible with this captcha script?

^ | v • Reply • Share ›



**Sachin Walvekar** • 6 years ago

Its working.....need to make a small change to the code in captcha.php

change:

```

imagefttext($image, 30, 0, 10, 40, $alt, $dir."arial.ttf",
$_SESSION['rand_code']);

```

to:

```

imagefttext ($image, 30, 0, 10, 40, $color, $dir."arial.ttf", $string);

```

also do make a "fonts" directory and copy arial.ttf (download it, available for free on net) to it.

^ | v • Reply • Share ›



**Muhammad Azeem** • 6 years ago

some people says captch not showing.. copy arial.ttf and past into font folder ,

^ | v • Reply • Share ›



**Farhad** • 6 years ago

Tnkssssss That's Worked!!

IF Any Body Have Problem With This, pm me on yahoo s43l@yahoo.com

^ | v • Reply • Share ›



**idiot** → Farhad • 6 years ago

no it doesnt worked for me. my image is broken . :(

^ | v • Reply • Share ›



**Tom** • 7 years ago

for some reason my cptcha image is broken? Do I need to have anything enabled for PHP on the server side?

thanks

^ | v • Reply • Share ›



**Chuck** • 7 years ago

Zac;

Thanks for such a great "captcha" alternative that is very easy to use. I am using it on my site, now. I have but one suggestion.

You might want to convert the "code" from the user input to lowercase or the test will fail.

`if($_POST['code'] == $_SESSION['rand_code'])` should be

`if(strtolower($_POST['code']) == $_SESSION['rand_code'])`

Without this: If the user inputs one or more capital letters, the form will not work.

Thanks again,

Chuck

^ | v • Reply • Share ›



**camelos** • 7 years ago

but what if i have the same page with captcha on many tabs?  
only one page will work with this code

^ | v • Reply • Share ›



**Alex** • 7 years ago

I cant get the image to show, there was no arial.ttf file in the font folder so I got one from somewhere else but still the image wont show ??

^ | v • Reply • Share ›



**gyit** • 7 years ago

awesome tutorial! this was exactly what i was looking for....

^ | v • Reply • Share ›



**NicoB** • 7 years ago

Please tell me how to send this form details to the clients email and auto respond to user. My php is beginner stage



^ | v

• Reply • Share ›



**steve** • 7 years ago

I downloaded the source files and get a broken image when I upload to my server. Do I just need to put Arial.ttf font in the Font folder?

^ | v

• Reply • Share ›



**axelcureno** → steve • 7 years ago

Yes you do have tu put the Arial.ttf in the font folder in order to get it work.

1 ^ | v

• Reply • Share ›



**Neon** → axelcureno • 7 years ago

Yes I put the arail.ttf font in the font folder but still broken image

1 ^ | v

• Reply • Share ›



**miniContent** • 7 years ago

Thanks for sharing, this is what I am looking for!

^ | v

• Reply • Share ›



**tricajus** • 8 years ago

thanks for sharing this tutorial. but i'm looking tutorial make web form with captcha for blogspot..

^ | v

• Reply • Share ›



**JacobTheDev** • 8 years ago

Not sure that people are still commenting on this, but is there a way to make it pull the form fields from the page itself? I want to be able to add whatever form fields I add to the page dynamically instead of having to specify each field in the script. I build a ton of sites, so having something that's adaptable is very important to me.

QUICK LINKS - Explore popular categories

ENVATO TUTS+	+
JOIN OUR COMMUNITY	+
HELP	+



28,5001,27340,043

TutorialsCoursesTranslations

[Envato.com](#) [Our products](#) [Careers](#) [Sitemap](#)

© 2020 Envato Pty Ltd. Trademarks and brands are the property of their respective owners.

Follow Envato Tuts+  [Facebook](#)  [Twitter](#)  [Pinterest](#)