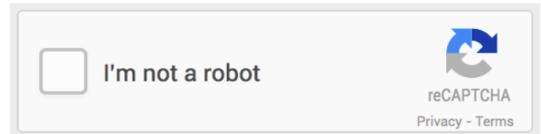


TUTO : Intégrer Google Recaptcha dans vos formulaires

I- Introduction

Recaptcha est un captcha gratuit qui protège les sites web contre les piratages et les spams. Il utilise des techniques d'analyse de risque avancées pour protéger contre les menaces. Recaptcha vient sous forme d'un champ que l'on peut ajouter dans un blog, forum, etc... Allez sur le site suivant pour commencer le service Recaptcha : « www.google.com/recaptcha ».



II- Préparation

Pour utiliser le Recaptcha, on a besoin d'une clé API privées et publiques pour qu'il se communique de façon sécurisée avec le serveur Google. La clé du site est utilisée pour afficher le champ de captcha dans notre site. Et la clé secrète autorise la communication entre l'application (site web) et le serveur Recaptcha pour vérifier la réponse d'utilisateur.

La clé API est unique par domaine. Attention : Par exemple, si vous indiquez la clé API pour « exher.fr » alors le recaptcha va uniquement fonctionner dans ce domaine et ses sous domaines.

Enregistrer un site

Libellé bhuva
Domaines (un par ligne) exher.fr
<input checked="" type="checkbox"/> Envoyer des alertes aux propriétaires ?
Enregistrer

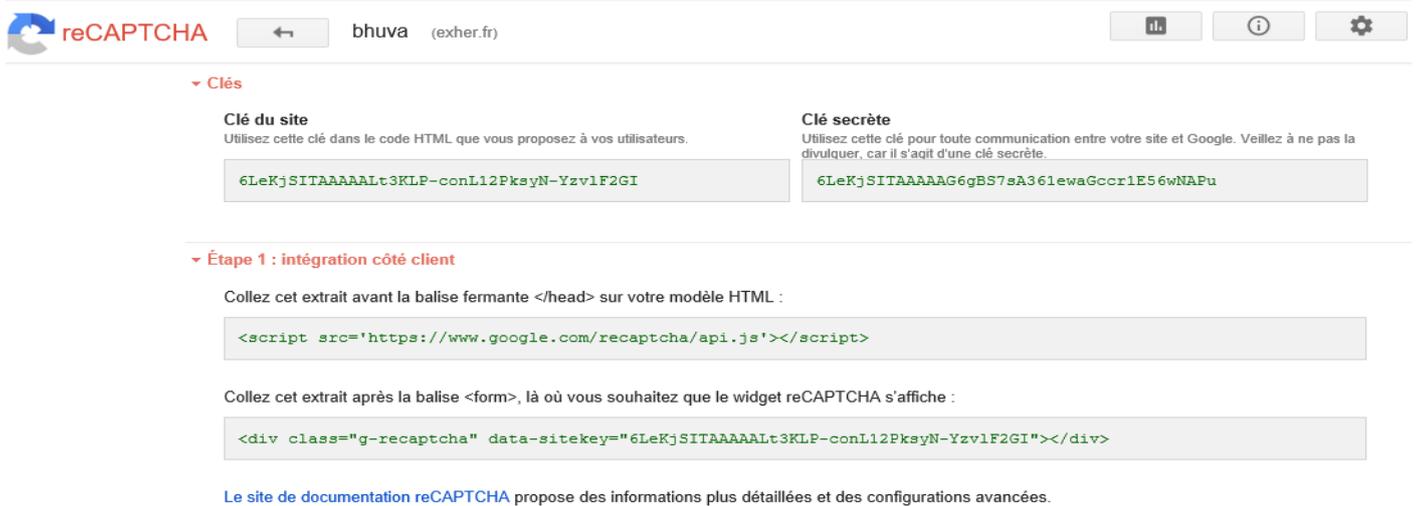
Par défaut, toutes les clés fonctionnent sur localhost (ou 127.0.0.1), donc tu peux développer et tester tes codes sur ton machine local (localhost).

Specified domain	Website domain	Will reCAPTCHA work?
yoursite.com	yoursite.com	Yes
	www.yoursite.com	Yes
	subdomain.yoursite.com	Yes
	subdomain.yoursite.com:8080	Yes

III- Les clés API

Une fois que vous avez enregistré votre domaine, vous allez recevoir une clé API privés et une clé publique (secrète) lequel vous allez utiliser dans votre ligne de code.

Recap : La clé du site est utilisée pour l’affichage du captcha dans notre site. Et la clé secrète autorise la communication entre l’application (site web) et le serveur Recaptcha pour vérifier la réponse d’utilisateur.

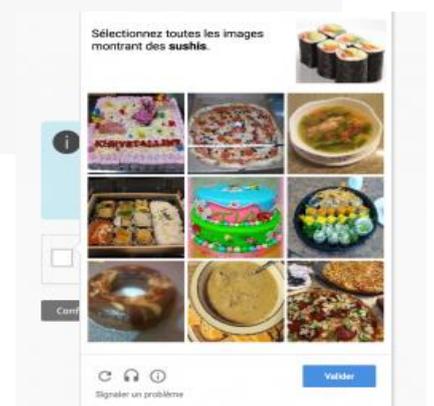


The screenshot shows the reCAPTCHA developer console for the domain 'bhuva (exher.fr)'. It displays two keys: a 'Clé du site' (6LeKjSITAAAAALt3KLP-conL12PksyN-Yzv1F2GI) and a 'Clé secrète' (6LeKjSITAAAAAG6gBS7sA361ewaGccr1E56wNAPu). Below the keys, it provides instructions for integration, including a code snippet for the script tag and another for the form's div tag. A link to the reCAPTCHA documentation is also present.

IV- Affichage et personnalisation du recaptcha

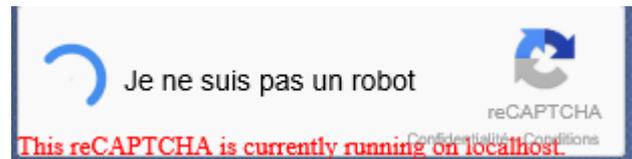
La méthode la plus facile pour utiliser le Recaptcha est d’ajouter la bibliothèque nécessaire tels que l’étiquette « g-recaptcha » au fichier des codes. L’étiquette « g-recaptcha » est un élément de ‘DIV’ avec un class name ‘g-recaptcha’ et votre clé publique dans le paramètre « data-sitekey ».

```
<html>
<head>
  <title>reCAPTCHA demo: Simple page</title>
  <script src="https://www.google.com/recaptcha/api.js" async defer></script>
</head>
<body>
  <form action="?" method="POST">
    <div class="g-recaptcha" data-sitekey="your_site_key"></div>
    <br/>
    <input type="submit" value="Submit">
  </form>
</body>
</html>
```



V- Validation du recaptcha (vérification du réponse)

Une fois que l'internaute clique sur le bouton captcha, un nouveau champ « g-recaptcha-response » va être ouvrir.

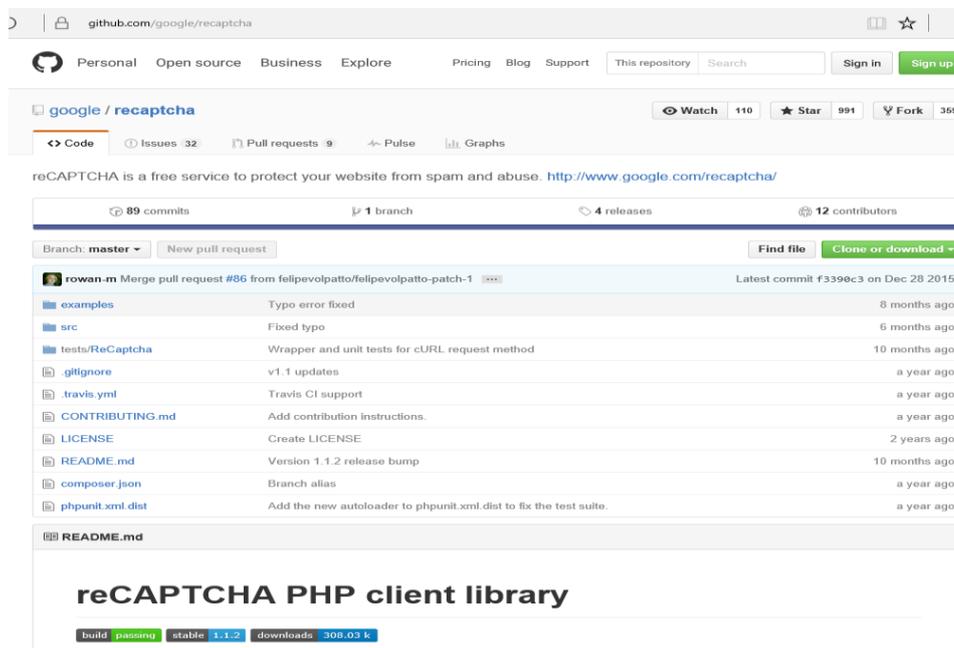


Pour vérifier la réponse de l'internaute, il existe trois méthodes :

- g-recaptcha-response : un paramètre POST lorsque l'internaute envoie la balise « form » dans notre site
- grecaptcha.getResponse(opt_widget_id) : après l'internaute complète le challenge captcha.
 - La méthode Callback function :

1. Téléchargement de la librairie recaptcha client

Dans notre cas, on utilise la méthode « g-recaptcha-response » pour ceci, on doit télécharger la librairie PHP sur le site : 'www.github.com/google/recaptcha' qui va nous permettre de valider le captcha.



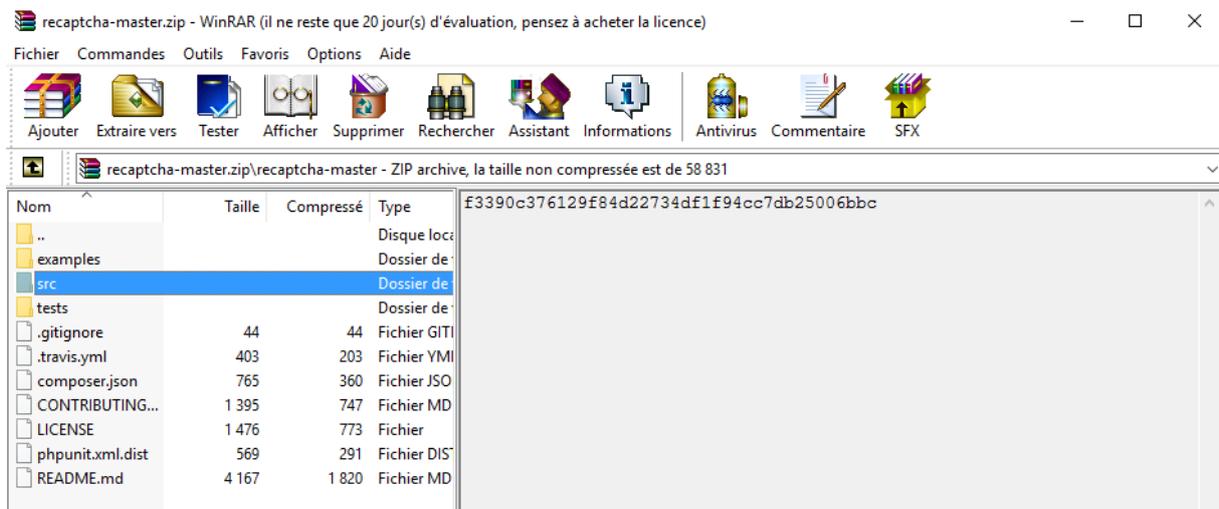
Allez-y sur la partie 'Direct download' pour installer manuellement la librairie et puis cliquez sur **ZIP File** pour télécharger le dossier ZIP. Pour la commodité, ils ont fourni un script « autoload.php (d'autochargeur) » dans src/autoload.php que on va utiliser dans notre script au lieu de vendor/autoload.php de Compositeur.

Direct download (no Composer)

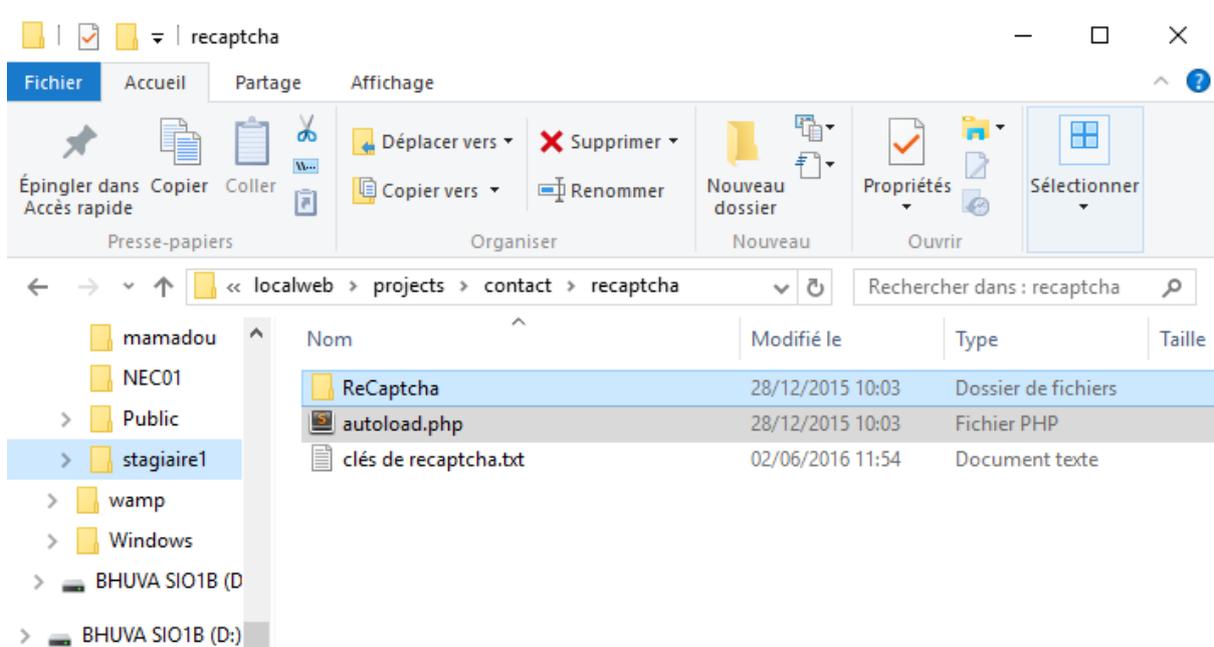
If you wish to install the library manually (i.e. without Composer), then you can use the links on the main project page to either clone the repo or download the [ZIP file](#). For convenience, an autoloader script is provided in `src/autoload.php` which you can require into your script instead of Composer's `vendor/autoload.php`. For example:

```
require('/path/to/recaptcha/src/autoload.php');
```

Dans le dossier que vous avez téléchargé, nous allons intéresser au fichier du sous-dossier « src ».



Copier le dossier « src » dans lequel se trouve le fichier « autoload.php » et puis coller dans le répertoire de votre site pour qu'il prenne en compte le fichier autoload.



Puis on inclut le fichier autoload avec un « require » pour qu'il prenne en compte la librairie de recaptcha qu'on vient de télécharger sur github.

```
require('recaptcha/autoload.php'); // on intègre le fichier autoload pour qu'il prenne en compte le bibliothèque de recaptcha qu'on vient de télécharger sur github
```

2. Utilisation du librairie reCaptcha client

Retourner sur le site www.github.com, dans la partie 'usage' vous allez trouver un modèle du code php pour la validation du captcha que vous allez personnalisé à votre choix.

Voilà mon code de php pour la validation de captcha que j'ai intégré dans un formulaire de contact :

```
1 <?php
2
3 if(isset($_POST['nom']) and isset($_POST['prenom']) and isset($_POST['email']) and isset($_POST['tel']) and
4   isset($_POST['sujet']) and isset($_POST['message']) ) //on vérifie si les variables existent
5 {
6     if(!empty($_POST['nom']) AND !empty($_POST['prenom']) AND !empty($_POST['email']) AND !empty($_POST
7       ['tel']) AND !empty($_POST['sujet']) AND !empty($_POST['message']) )
8         // on vérifie si les variables ne sont pas vides
9     {
10        extract($_POST);
11
12        //Déclaration de l'adresse de destination
13        $destinataire = "ppeinfratech@gmail.com";
14        $email = htmlentities($_POST['email']);
15
16        //on vérifie la syntaxe du mail
17        if(preg_match('#^([a-z0-9!#$%&'\*\+=?^_`{|}~]+\.?)*[a-z0-9!#$%&'\*\+=?^_`{|}~]+)@((
18          a-z0-9-]+\.\.?)*[a-z0-9-]+\.[a-z]{2,}$#i',str_replace('&','&', $email))) //on filtre les
19          serveurs qui rencontrent des bogues
20        {
21            $nom = stripslashes($_POST['nom']);
22            $prenom = stripslashes($_POST['prenom']);
23            $tel = stripslashes($_POST['tel']);
24            $sujet = 'Contact: '.stripslashes($_POST['sujet']);
25            $message = stripslashes($_POST['message']);
26            $headers = "From: <". $email.">\n";
27            $headers .= "Reply-To: " . $email . "\n";
28            $headers .= "MIME-Version: 1.0" . "\r\n";
29
30            // $destinataire="ppeinfratech@gmail.com";
31
32            $headers .= "Content-Type: text/plain; charset=\\"iso-8859-1\\"" ;
33
34            require('recaptcha/autoload.php'); // on intègre le fichier autoload pour qu'il prenne en
35            compte le bibliothèque de recaptcha qu'on vient de télécharger sur github
36
37            if(isset($_POST['g-recaptcha-response']))
38            {
39                //inclure le code secrète de recaptcha dans la variable $recaptcha
40                $recaptcha=new \ReCaptcha\ReCaptcha('6LeikyETAAAAAKRuRnfc7eHc-haxsnfIhtJg3Mod');
41                $resp = $recaptcha->verify($_POST['g-recaptcha-response']);
42                if ($resp->isSuccess()) {
43                    //echo "Captcha valide";
44
45                    //on utilise la fonction mail pour l'envoi de l'email
46                    if(mail($destinataire,/*$nom,$prenom,$tel,*/$sujet,$message,$headers))
47                    {
48                        echo "<strong>Votre message a bien &acute;t&acute; envoy&acute;.</strong>";
49                    }
50                    else
51                    {
52                        echo "<strong style=\\"color:#ff0000;\\">Une erreur c'est produite lors de l'envoi
53                          du message. NOTE: Tous les champs doivent être obligatoirement être remplie</
54                          strong>";
55                    }
56                }
57                else
58                {
59                    $error= $resp-> getErrorCodes();
60                    echo "<strong style=\\"color:#ff0000;\\"> Captcha invalide. Veuillez Remplir le case de
61                      Captcha!! </strong>";
62                    echo $error;
63                }
64            }
65            else
66            {
67                echo "<strong style=\\"color:#ff0000;\\"> Captcha non remplie </strong>";
68            }
69        }
70    }
71}
```

```
72     }
73     else
74     {
75         echo "<strong style=\"color:#ff0000;\">L'email que vous avez entré est invalide.</
76         strong>";
77     }
78     else
79     {
80         echo "<strong style=\"color:#ff0000;\"> Tous les champs doivent etre remplie!!! </strong>";
81     }
82 }
83 else
84 {
85     echo "Formulaire non soumis ou des champs sont vides";
86 }
87 ?>
```