Download and Configure Formmail.PHP

Please note: in order to use a form script on your own server, your host must be running PHP. Check with your host that you have this facility.
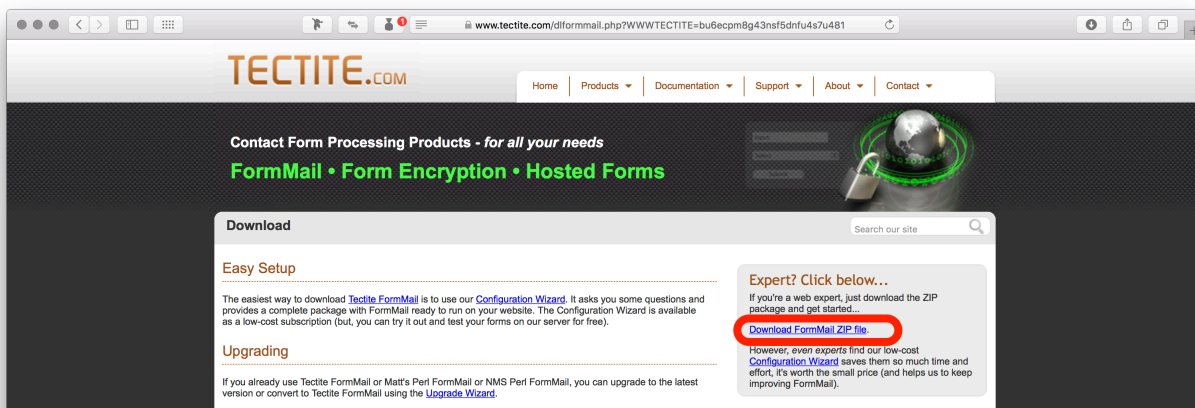
Download the script

Go to http://tectite.com

Click the link highlighted in **Red** in the illustration below:

This will take you to the download page. Click the link highlighted in **Red** in the illustration below.

Clicking the link will download a zip file to your default download location. Locate the zip file and decompress it (*double click to do this*). You will then see a formmail folder. open the folder to see two files. One is the actual script (*highlighted in **Red** in the illustration below*), the other is a sample form which you can ignore.

formmail.php   sampleform.htm
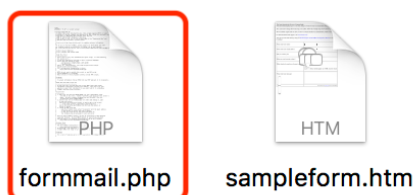
Configure the Script

To configure the script, you will have to make a minimum of two changes. To do this, open the formail.php file in a plain text editor such as notepad or notepad++. In the illustration below, the file has been opened in "sublime text". If you have this app, it colour-codes the elements making it easier to find the areas that need editing. You can download a free copy of sublime text from **https://www.sublimetext.com**.

Scroll down the page to line **427**. This is the start of the configuration area of the script. Make the changes highlighted in **Red** in the above illustration. Refer to the notes below to see how to make the changes.

Note 1: This line reads as follows:

/* Help: http://www.tectite.com/fmdoc/target_email.php */
$TARGET_EMAIL = array($EMAIL_NAME . "@yourhost\.com$");

The only change you need to make here is to enter the email domain of where you want your responses to be delivered. For example, if your form results are to be delivered to recipients at email addresses ending in **mywebsite.com,** you will change the code highlighted in orange to read **"@mywebsite\.com$"** Be sure to include the opening and closing quote marks and make sure there is a backslash preceding the dot before the domain extension and a **$** symbol before the closing quotes. Your script will now be able to forward email responses to ANY address ending in **mywebsite.com.**

> **VARIANTS**
> If you use generic, public domain email addresses, such as gmail or yahoo, you should configure this line to show a FULL EMAIL ADDRESS. If you use this option, you will change the line of code to read.
>
> $TARGET_EMAIL = array("^fredbloggs@gmail\.com$"); In this example the recipients exact email address is fredbloggs@gmail.com. Notice that the string $EMAIL_NAME has been deleted and replaced with the full email address. Also note that full email addresses must be preceded with a caret character (^). Also, any dots within the email name must be preceded with a backslash.

Note 2: This line reads as follows:

/* Help: http://www.tectite.com/fmdoc/def_alert.php */
$DEF_ALERT = "";

Here you will tell the script the email address to which any error messages or other alerts should be sent. Ideally, this should be the email address of the administrator of the website. Simply add an email address between the quotes. e.g. $DEF_ALERT = "administrator@mywebsite.com";

Note 3: This line reads as follows:

/* Help: http://www.tectite.com/fmdoc/at_mangle.php */
$AT_MANGLE = "";

The "AT MANGLE" parameter is a security feature that is highly recommended. It's essentially a string of characters that can replace the "@" symbol in a recipients email address included in your WebPlus forms. When you get to changing your WebPlus forms, you will be adding a hidden field named "**recipients**". This field will tell the script the exact email addresses to where the form results should be sent. Clearly, this will expose the recipients email address in the html code of your web page, where it could become vulnerable to harvesting by a spam bot. By setting up an AT MANGLE string, you can replace the "@" symbol in your email addresses with the AT MANGLE string, thereby disguising the email address from spam bots.

To use this optional feature, simply type any string of characters between the quotes. e.g.

$AT_MANGLE= "bbbccc" Once set up, any recipients email addresses you set up in your form's recipient field would be entered as **mynamebbbcccmywebsite.com**, thereby removing the "@" symbol and making it harder for spam bots to see that it is an email address.

Final Steps

Now that you have made the required changes, you should save your script. It is recommended that you save to a different name that doesn't identify your file as a formmail script. You can give it any name of your choosing, just be sure that it ends with **.php**.

At this stage you can upload your script to your server for testing. This is easily done through your web host's control panel. There will normally be an option for uploading individual files to your domain and for creating specific folders within your domain. If you have this option, create a folder named "scripts" or something similar. Then upload your renamed file to that folder. You can, if you wish, simply upload it to the root of your domain, but it's usually better to have it in it's own folder.

If you don't have a control panel option, you can use any FTP application to upload your script file. In a worst case scenario, you can wait until you've configured your WebPlus forms and have the script upload through WebPlus when you update your site.

If you can upload to your site before updating your WebPlus forms, you will be able to test that your script works. To do this, type the following into the url bar of any web browser.

http://www.mywebsite.com/scripts/formmail.php?testalert=1

Obviously, you will change **mywebsite.com** to your own domain name. You will also change the **scripts** folder to the name of the folder where you uploaded your script file (*or omit it altogether if the file is in the root of your domain*) and you will change **formmail.php** to the name you gave your script file.

This will cause a test message to be sent to the administrator's email address set up under the **DEF_ALERT** parameter detailed above. If there is a problem with your configuration, the test email will tell you where the problem is. If the message doesn't contain any errors, your script is good to go.

Setting Up Your Forms in WebPlus

The last thing you have to do is configure your WebPlus forms to use the script. This is very easy to do and only requires the addition of some "hidden fields" in your forms and pointing your forms to your new script file. Check your WebPlus user instructions on how to add hidden fields to your forms. The fields you need to create are as follows:

Field Name	Value
recipients	the email addresses to where form results should be sent.
good_url	the Full URL of your thank you or success page
bad_url	the Full URL of your error or failure page
subject	the text you want to appear in the subject line of emails sent to you

The **recipients** field can contain one or more email address separated by commas. If you've set the AT_MANGLE option in your script, be sure to replace the "@" symbol in your email addresses with the string of characters set in the AT_MANGLE option.

It is important to remember that all email addresses added to this field must match the domain name set in the TARGET_EMAIL parameter in the script. In other words, if you set the TARGET_EMAIL parameter to **"@mywebsite\.com$"** then the email addresses in the recipients field must end with **@mywebsite.com** If you set a full email address in the TARGET_EMAIL parameter, then that is the only email address that can be set up in the recipients field. This is a security feature that prevents your script from being used as a spam gateway.

The **good_url** field should be the thank you or success page of your website. Enter it as a FULL URL e.g. **http://mywebsite.com/thankyou.html** After a form has been successfully submitted, your site visitors will be redirected to this page.

The **bad_url** field should be the URL of a failure page. Visitors will be directed to this page if there is a problem submitting the form. The administrator will be sent an error message by the script if this happens.

The **subject** field can be any text string that you want to appear in the subject line of response emails. e.g. **Website Enquiry from MyWebsite.com**

Once the above hidden fields have been created, you should go to your form settings and change where the form should be sent. If you've been using **Serif Web Resources**, your forms will be set to send to **SWR**. You need to change this to either **A Remote Script**, or **A Script from your hard drive**. If you've already uploaded and tested your script, choose the first option and add the full url of your script. e.g. **http://mywebsite.com/scripts/formmail.php** (*change the domain, path and file name to whatever is relevant in your case*). The two options are outlined in red in the illustration below:
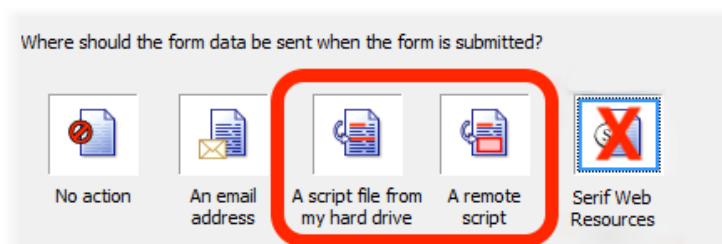
If you haven't yet uploaded your script, you can do so via WebPlus by changing the submit option to **A Script on my hard drive**. This will allow you to browse for your saved script file and add it to your WebPlus project. The script will then be uploaded by WebPlus when you update your site.

Things to Remember

Once your script is set up and running, you can make as many forms as you wish and point them all to the same script. You do not need a separate script for every form. Each form can have it's own recipients and can all have different success or failure redirects - just set them up in hidden fields on each form.

If you have multiple websites, you can configure the script to mail recipients on any number of domains by simply adding more domains to the TARGET_EMAIL parameter. By setting up this option, your one script will act like a personal SWR service that can be used by any number of domains - you don't need to have the script uploaded to every domain for this to work. Below is an example of how to add multiple domains:

```php
$TARGET_EMAIL = array($EMAIL_NAME."@mywebsite\.com$",
        $EMAIL_NAME."@another website\.com\.au$",
        $EMAIL_NAME."@yet another website\.com$",
        $EMAIL_NAME."@somewebsite\.org$",
        $EMAIL_NAME."@timetabling-scheduling\.com$",
        $EMAIL_NAME."@freindswebsite\.com$"
        );
```

Note that domains with double extensions such as **.com.au** or **.co.uk** should have a **** before the dot. Also, make sure that all **" , ()** and **$** symbols are in the correct places as shown.

By setting up the script to work with multiple domains, only the domain hosting the script has to have PHP running.

There are a number of built-in security functions in the script that helps to combat spam email. Therefore, it isn't necessary to have captchas and re-captchas. Should you start receiving a large amount of spam email through your website, there are other security features that can be switched on within the script. You can check the Tectite website for further information on how to turn these additional features on or off.

It is well worth looking through the tectite documentation for full details on the numerous other configuration options available, including handling file uploads through your form and creating formatted custom HTML email responses instead of plain text.

ENJOY!