

- If the email does not match any of the conditions above, then the email will remain unchanged and will arrive at its regular destination.

Note: After reviewing the attachment and deeming it safe, you may rename the file to its former name to run it.

Note: To change the setting, check or uncheck, and then click Save Settings.

Enter extensions of attachments to be blocked (For example: exe, doc) Enter a filename extension, click Add Blocked, and then click Save Settings to add that extension to the **Blocked Extensions** list.

Blocked extensions Any email with an attachment that ends in an extension listed here will be stored in the Emails with known issues file and a warning will be sent to the administrator's email address. Select one or more of the extensions, click on Remove Block, and then click on Save Settings to remove them from the list.

Enter administrator's email address When the program determines that an email is problematic, it will send a warning message to this address.

Emails with known issues will be stored to file When the program determines that an email is problematic, it will store that email in this file. To change the name of the file, you need to highlight the text in the field, enter the new filename, and then click on Save Settings. You can also indicate directories. The directories and filename must be made up of alpha characters, the underscore, or the dash. No other special character can be used, e.g., the period. Once you make the change, make sure to click on Save Settings. Valid Examples: *virus_mail*, *mail/virus_mail*. Invalid Examples: *domain-mail/virus_mail.txt*, *mail/virus.txt*. To empty the contents of the file, click on the Empty link.

Note: You can store e-mails that the program has determined to be problematic to your -mail directory, and then create a user in your mail manager that matches the name of the Emails with known issues file. This will allow you to retrieve an email that you know is not problematic.

Select security level Select the level of tolerance the program uses to determine whether an email is problematic. To change this, click on the menu, select the level, and then click on Save Settings. Please note that this program does not offer full virus protection. For protection against viruses, it will be necessary to acquire a virus protection program for your mail client.

Chapter 15: PGP

Overview

PGP stands for *Pretty Good Privacy*. It is an encryption scheme designed to be unbreakable. PGPMail allows you to encrypt mail sent to you from forms on your website.

Traditional encryption schemes work like this: Alice wants to send a secret message to Bob, so she takes her message and encrypts it using a key. A key is an alphanumeric series used to scramble data, so that the message can't be read by someone just picking it up. Alice sends the message to Bob, and Bob decrypts the message, using his own copy of the key.

However, there are several problems with the above situation, the main one being that Alice, somehow, has to securely get a copy of the key to Bob. If she's sending encrypted messages in the first place, it stands to reason that she doesn't have a secure delivery method, and thus, will always have to worry that someone else will be able to decrypt her messages.

Fortunately, in the 70s, several mathematicians discovered a form of encryption that doesn't require both parties to have the same code. You would think that if Alice encrypts a message to Bob using Alice's key, and then Bob decrypts the message using his own, that the message would come out as gibberish. This isn't the case, because both Alice and Bob actually have two keys: one for encrypting, and one for decrypting. The key that people should use to encrypt messages to Alice is public. She can post it on her door, on her website, or publish it in the local newspaper. It doesn't matter.

She can do this, because with PGP, her key is going to be the product of two very large prime numbers. As it turns out, factoring large numbers is very difficult, and when the only two factors are primes, even more so. Alice's public key will be the product of these two prime factors, and her private key will be the two prime factors themselves.

Bob sends Alice a message encrypted using her public key. It doesn't matter if anyone else gets hold of the message, because only Alice will be able to interpret it. Bob doesn't need to private key, because he's not doing any decrypting.

Alice receives the message, and uses her private key to decrypt the message. If she wants to send a reply to Bob, she'll use the public key that he's posted online, and he'll use his private key to decrypt the reply.

To access the PGP Public Key Manager:

Click the PGP icon in your Control Panel.

Adding a Public Key

To add a public key for PGPMail:

- 1 Create a key using your mail software. The mail software should come with instructions describing how to do this.
- 2 Once the key is created, go to the PGP option in your Control Panel. Copy the public key information and paste it into the text area labeled PGP Mail Public Key.
- 3 Click the Add button; this will give you a public key to use with PGPMail.

Using PGPMail

To use PGPMail after uploading your PGP Key:

Create a form on one of your web pages.

The form action line should be `<FORM ACTION = "/cgi-sys/pgpmail.pl" METHOD = "POST">`

`pgpmail.pl` will do all the programming work for you. You alter the behavior of PGPmail by using hidden fields in your form.

There are three form fields that you must have in your form for PGP mail to work correctly. These are the recipient, username, and keyname fields.

Required Form Fields

recipient

This form field allows you to specify where your form results will be mailed. Most likely you will want to configure this option as a hidden form field with a value equal to that of your email address.

Syntax `<input type=hidden name="recipient" value="anyname@yourdomain.com,othername@yourdomain.com">`

username

This field tells PGPMail where to look for the configuration files to encrypt mail sent to you. You should replace *yourusername* with your Control Panel username.

Syntax `<input type=hidden name="username" value="yourusername">`

keyname

This form field allows you to specify the name of your public key. This will be the public key that PGPMail uses to encrypt your mail. You must possess the private key in order to

decrypt the email that is sent. You can get your public key name by going to PGP Manager. It is typically just your email address.

Syntax `<input type=hidden name="keyname" value="publickeyname">`

Optional Form Fields

subject

The subject field allows you to specify the subject that you wish to appear in the email sent to you. If you do not have this option turned on, then the subject will default to: WWW Form Submission .

Syntax If you wish to choose the subject: `<input type=hidden name="subject" value="Your Subject">`

To allow the user to choose a subject: `<input type=text name="subject">`

email

This form field allows the user to specify his or her return address. If you want to be able to respond to form submitted emails, it is strongly suggested that you include this field and allow your users to fill it in. This will be put into the From: field of the message you receive.

Syntax `<input type=text name="email">`

realname

The realname form field allows the user to input his or her name. This field is useful for identification purposes and will also be put into the From: line of the message header.

Syntax `<input type=text name="realname">`

sort

This field allows you to choose the order in which you wish your variables to appear in the email that FormMail generates. You can choose to have the field sorted alphabetically or specify a set order that the fields will appear in mail messages. By leaving this field out, the order will simply default to the order in which the browsers send the information to the script (which isn't always the same order the variables appear in the form.) When sorting by a set order of fields, you should include the phrase order as the first part of your value for the sort field, then follow with the field names you want to be listed in the email message, separated by commas.

Syntax To sort alphabetically: `<input type=hidden name="sort" value="alphabetic">`. To sort by a set field order: `<input type=hidden name="sort" value="order:name1,name2,etc...">`

redirect

If you wish to redirect the user to a different URL, rather than having them see the default response to the form, you may use this hidden variable to send them to another webpage.

Syntax To choose the URL the user will redirected to: `<input type=hidden name="redirect" value="http://your.address/file.html">`. To allow the user to specify a URL he wishes to travel to once the form is filled out: `<input type=text name="redirect">`

require

You can require that users fill in certain fields before the user can successfully submit the form. Simply place all field names that you want to be mandatory into this field. If the required fields are not filled in, the user will be notified that information is missing, and a link back to the form they just submitted will be provided.

Syntax If you want to require that the user fill in the email and phone fields in your form, so that you can reach them once you have received the mail, use a syntax similar to: `<input type=hidden name="required" value="email,phone">`

env_report

You may include environment variables in the email message you receive after a user has filled out your form. This is useful if you wish to know what browser they were using, what domain they were coming from, or any other attributes associated with environment variables. The following is a short list of environment variables you may find useful:

- REMOTE_HOST sends the hostname making a request.
- REMOTE_ADDR sends the IP address of the remote host making the request.
- HTTP_USER_AGENT is the browser the client is using to send the request.

Syntax If you wanted to find the remote host and browser sending the request, you would put the following into your form: `<input type=hidden name="env_report" value="REMOTE_HOST,HTTP_USER_AGENT">`

title

This form field allows you to specify the title and header that will appear on the resulting page if you do not specify a redirect URL.

Syntax If you wanted a title of Feedback Form Results: `<input type=hidden name="title" value="Feedback Form Results">`

return_link_url

This field allows you to specify a URL that will appear on the report page. This field will not be used if you have the redirect field set, but it is useful if you allow the user to receive the report on the following page, but want to offer them a way to get back to your main page. You may set the text of the link in the return_link_title field.

Syntax `<input type=hidden name="return_link_url" value="http://your.host.xxx/main.html">`

return_link_title

This is the title that will be used to link the user back to the page you specify with return_link_url. The two fields will be shown on the resulting form page as: ` return_link_title `

Syntax `<input type=hidden name="return_link_title" value="Back to Main Page">`

background

This form field allows you to specify a background image that will appear if you do not have the redirect field set. This image will appear as the background of the form results page.

Syntax `<input type=hidden name="background" value="http://your.host.xxx/image.gif">`

bgcolor

This form field allows you to specify a background color for the form results page in much the way you specify a background image.

Syntax For a background color of white: `<input type=hidden name="bgcolor" value="#FFFFFF">`

text_color

This field works in the same way as bgcolor, except that it will change the color of your text.

Syntax For a text color of black: `<input type=hidden name="text_color" value="#000000">`

link_color

This field changes the color of links on the result page and works in the same way as text_color. If redirect is defined, then this link will have no effect.

Syntax For a link color of red: `<input type=hidden name="link_color" value="#FF0000">`

vlink_color

This field changes the color of visited links on the result page and works in the same way as text_color. If redirect is defined, then this link will have no effect.

Syntax For a visited link color of blue: `<input type=hidden name="vlink_color" value="#0000FF">`

alink_color

This field changes the color of active links on the result page and works in the same way as text_color. If redirect is defined, then this link will have no effect.

Syntax For a visited link color of blue: `<input type=hidden name="alink_color" value="#0000FF">`

Note: Any other form fields that appear in your script will be mailed back to you and displayed on the resulting page if you do not have the redirect field set.