

tmail.pl

Tmail, for "Template Mail", is a little Perl script I wrote to take the place of the common cgi program *cgiemail*, which was identified as having some security issues. You can read more about those issues in "[A SPAMMER is using my cgiemail, what do I do?](#)" at [Ask-Leo.com](#).

Tmail is executed as the result of an HTML Form, and reads a specified text file as a template. Tokens specified in the template file between square brackets are replaced with the values of parameters by those names entered in the form. The results are then mailed via sendmail.

Parameters (specified as hidden variables in the HTML form):

template=filename: filename of template to process, relative to the perl script's execution environment. Required.

success=page: page to redirect to on successful completion. ../html/index.html if not specified.

failure=page: page to display on failure. IF specified, the page is parsed on output to allow the specific error message to be inserted. If not specified, an extremely simple error page is generated.

The template file is just a text file to be fed to "sendmail -oi -t". Form variables are referenced in [brackets], with substitution as follows:

[plain]: replaced with the value of the passed in parameter "plain". Error if no such param.

[@email]: replaced with the value of the passed in parameter "email", which is pumped through some rudimentary email validity checking. Error if no such param, OR if the email address fails the validity check. (Note that since the error checking only has to happen once, you need only to use this once per variable. Subsequent instances can be [email].)

[\$VAR]: replaced with the value of the server's environment variable "VAR". Silently ignored and removed if no such var.

![prot]: replaced with the value of the passed in parameter "prot". Error if no such param. Prot is "protected", meaning it is restricted in the type of data it may contain. Should it fail that restriction, an error is generated. Right now that restriction causes failures if the data contains a carriage return or line feed (newline) character. (Note that like "@", you really only need to use this once per variable.)

[#optional]: replaced with the value of the passed in parameter "optional", or REMOVED if there is no such parameter.

Note: a line in the template consisting only of

[newmail]

is considered a message boundary. When encountered the lines before it are packaged as email and sent, and lines after it are the start of a **new** mail message, which is sent the next time [newmail] is encountered, or when the template file ends. Essentially this is a simple way for tmail.pl to be able to send multiple, different, emails at a single request.

Recommendation is that the template files be kept in the cgi-bin directory or equivalent, or some other not-web-readable location.

IMPORTANT: if you place an unprotected field in your email header, you are at risk for spammer hijacking. The header of your email - that means everything up to the first blank line - should include only "@" email variables, and "!" protected variables. In general, it's recommended the only user-entered data you allow in the email header be an email address. Anything else should only be placed in the body of the email.

Example

This is a quick form to test tmail.pl:

```
<FORM METHOD="POST" ACTION="/cgi-bin/tmail.pl">
<INPUT NAME="template" TYPE="hidden" VALUE="tmailtest.txt">
<INPUT NAME="success" TYPE="hidden" VALUE="http://ask-leo.com">

<BR>Email: <INPUT NAME="email" TYPE="text" SIZE="45">
<BR>Plain Field: <INPUT NAME="plain" TYPE="text" SIZE="45">
<BR>Protected: <INPUT NAME="prot" TYPE="text" SIZE="45">
<BR>Optional: <input type="radio" name="opt" value="one">One
               <input type="radio" name="opt" value="two">Two
<BR><INPUT TYPE="submit" VALUE="Submit" NAME="Submit1">
</FORM>
```

And this would be the template file **tmailtest.txt**:

```
From: tmail-example@pugetsoundsoftware.com
To: [@email]
Bcc: youremail@yourdomain.com
Subject: tmail.pl test

Plain Field: [plain]
Protected: [!prot]
Optional: [#opt]
Environment: [$HTTP_REFERER]
```

The net result after the files are all installed in their appropriate locations is that values entered on the form are replaced in the specified locations in the template, and the result is sent via email.

Go ahead, try it here:

Email:

Plain Field:

Protected:

Optional: ☐ One ☐ Two

tmail is freeware. It's a server-side Perl script, so it assumes you have CGI capability and Perl on your web server.

Unfortunately because of today's litigious society, I also have to tell you the following:

By using tmail you agree to assume **all** risk for its use. It may or may not meet your needs. It's extremely unlikely, but it might have bugs that could harm or delete files on

your computer. Perhaps all of them. Puget Sound Software assumes **no** liability for any damage caused by your use of tmail in any way.

If you can't accept those terms, then don't use tmail.

With that out of the way, I will also say that if you find a bug in tmail I'll attempt to correct the problem and get you a new version for free. Since tmail is freeware that last part is kinda silly, but it's my policy for all the software I'm providing here, free or not.

You can download tmail.pl [here](#) 6,048 bytes, ZIPped. Also included is tmail-conversion.txt, a quick guide to converting existing cgiemail usages.