



EMU Webmail 7.x Software Manual

**Updated 28/11/2005
Copyright 2005 - EMUMAIL, Inc.**

Table of Contents

Chapter 1: Introduction

- 1.1. What is EMU Webmail?
- 1.2. About this Document

Chapter 2: Installing EMU Webmail

- 2.1. System Requirements
- 2.2. Unix Install Instructions
 - 2.2.1. Instructions for RPM install
 - 2.2.2. Instructions for TarBall install
- 2.3. Windows Install Instructions
- 2.4. Identifying the Web Server Account
- 2.5. Multiple Domains

Chapter 3: System Structure

Chapter 4: Configuring EMU Webmail

- 4.1. Configuration File Overview
- 4.2. Configuration File Format
- 4.3. Special Values in Configuration Files
- 4.4. Adding Custom Variables
- 4.5. Precedence Among Files
- 4.6. Licensing the Product
- 4.7. Restricting Access
- 4.8. User Quotas
- 4.9. site.emu variables
 - 4.9.1. Common Options
 - 4.9.2. User Defaults
 - 4.9.3. Basic Interface Configuration

Chapter 5: Customizing the Interface

- 5.1. Adding a New Interface
- 5.2. Modifying the User Interface
 - 5.2.1. Introduction
 - 5.2.2. Skins
 - 5.2.2.1. Skin CSS
 - 5.2.2.2. Skin Images
 - 5.2.2.3. Creating Your Own Skin
 - 5.2.2.4. Browser Compatibility Issues
 - 5.2.3. EmbPerl
 - 5.2.3.1. Calling EmbPerl
 - 5.2.3.2. Meta Commands
 - 5.2.3.3. if, elsif, else, endif
 - 5.2.3.4. foreach, endforeach
 - 5.2.3.5. EmbPerl Resources

Chapter 6: Troubleshooting FAQ

Chapter 7: Additional Support

Chapter 1: Introduction

1.1. What is EMU Webmail?

EMU Webmail is a software package that runs on your web server, allowing your users access to their own POP or IMAP email accounts from any web browser. EMU Webmail has many advantages, including the following:

EMU Webmail is a universal client.

Other desktop-based email clients, like Eudora and Pegasus, require users to have configured software on their local machine. Not the case with EMU Webmail. The only requirement for using EMU Webmail is a web browser.

EMU Webmail is email security.

You shouldn't be forced to rely on someone you don't trust to safeguard sensitive data. Other service-based Webmail companies make you store your email on their computers. Why should you to trust them to make backups of your important data? How easy is it for them to view your files? Very easy. But not with EMU Webmail. EMU Webmail answers these security concerns by running on your own system and allowing you to store email locally, so you never have to worry that your data could get into the wrong hands. And when it is run off a secure server, EMU Webmail provides a very high level of security for email in transit.

EMU Webmail is email for everyone.

EMUMAIL wasn't designed for computer engineers. It was designed to be a simple yet powerful tool for everyone to use. Type in a URL and before you can say "easy, global and secure," you're checking your email.

EMU Webmail is simplicity for system administrators.

EMUMAIL makes life simple for system administrators. A system administrator needs only to install EMU Webmail once for everyone on the network to use the program. When upgrades become available, one install is all that's needed for every user to enjoy the latest enhancements. No client-side software worries can mean fewer headaches for system administrators.

1.2. About this Document

This document contains all the information you need in order to install, run, configure, and customize EMU Webmail.

This document is intended for mainly for system administrators and web designers wishing to install, maintain, or customize EMU Webmail. Users who check their email with EMU Webmail and have questions about their accounts should consult the online help pages, or their system administrator.

Chapter 2: Installing EMU Webmail

This chapter will guide you through the installation of EMU Webmail. Follow the instructions appropriate for your platform (Unix or Windows NT). Unix installs come in two flavors: with RPM and without. Installing with RPM is easier as most of the hard work is done for you. Some administrators may still favor Tarball installations as they offer the greatest flexibility

After you have installed the system, you may wish to tweak the configuration according to your needs. Consult the chapter on Configuration for guidelines.

2.1. System Requirements

We recommend that your system meet the following requirements before you download and install EMU Webmail:

- 1 GB of RAM. 10 MB per instantaneous access with FastCGI (optional).
- 100 MB of available disk space, 2 MB per user for storage.
- Required for Unix only: Perl 5.005_03 or greater.
- Existing POP3 or IMAP mail server populated with email accounts.
- Existing SMTP server.
- WWW server capable of CGI.
- Internet connectivity.

2.2. Unix Install Instructions

2.2.1. Instructions for RPM install

This section contains instructions for installing EMU Webmail via the RPM package management system. These instructions apply to all architectures compatible the RPM format. Note that the installation steps must be executed as the administrator (root) user. You may wish to download a binary distribution or a source distribution, depending on availability. If an RPM is not an option on your operating system, try the tar package format instead.

The EMU Webmail distribution will install the core Webmail files and necessary Perl modules inside the *EMU/* Directory.

The main EMU Webmail installation by default resides under */home/EMU/webmail*

1) Download the Webmail distribution from the EMUMAIL website at:

<http://www.emumail.com/downloads/>

Save the distribution to a temporary directory, and install it using the following command:

```
rpm -ivh <filename>
```

2) Install License Keys.

If you have a licensed version of the EMU software, you will next need to install license keys. If you do not have a licensed version on Webmail, please contact sales@emumail.com

To install a license key:

```
cd /home/EMU/webmail/data
```

Edit the site configuration file, *site.emu*, find the existing license data and replace it with your new license key. A sample license key looks like:

```
KEY__contact=user@EMUMAIL.com  
KEY__expire=970339698  
KEY__full license=yes  
KEY__version=demo  
KEY__revision=7.0  
KEY__master=632e0e3f324f2b8239g55cf72cb5cb67
```

Note: If you are replacing or updating the license data, ensure that you replace all lines, not just the ones that appear to have been changed.

3) Configure Web Server

Note: If you installed the bundled package, skip to Step 4.

Every web server configuration is different, so you may wish to consult your web server's manual for exact details of this step. In general you will need to setup CGI as a file type, provide the ability to access emumail.cgi from the web, and optionally setup fast cgi access.

Setting up .CGI as a file type is synonymous with "allowing emumail.cgi to be executed instead of downloaded." Some web servers require setting execute permissions, others require a mapping such that any files ending in .cgi will get passed to a cgi handler instead of being downloaded.

Under the Apache web server, this is done via the AddHandler configuration directive:

```
AddHandler cgi-script .cgi
```

to the Apache configuration file (httpd.conf). You should also set Options +ExecCGI in the configuration file under the scope of the directory in which the emumail.cgi file is installed. For example:

```
<Directory /home/EMU/webmail/html>  
Options Indexes FollowSymLinks ExecCGI  
AllowOverride None
```

</Directory>

Other web server will require different steps to install CGI capabilities.

4) Verify Installation

After this is done you're ready to point a web browser to the EMU Webmail Installation.

If you are using the bundled package, Webmail may be accessed at:

http://[yourdomain]:1010/

If you are not using a bundled package, access Webmail at the URL you specified in Step 3.

If you get a login page for EMU Webmail, and the title bar in your window says "EMU Webmail -Login," then everything is successful and you can continue on to the Configuration section of this document.

If you don't get a login page, or if the title of the login page contains an error, corrective steps will need to be taken, including re-running and double checking the steps in the installation process. See the Troubleshooting section of this document for more information.

2.2.2. Instructions for TarBall install

This section contains instructions for installing EMU Webmail on a Unix platform. These instructions apply to all flavors of Unix, including Linux, HP-UX, SunOS, Solaris, AIX, and others. Note that the installation steps must be executed as the administrator (root) user.

- 1) Download and unpack the EMU Webmail Distribution. EMU Webmail is available through the Internet at <http://www.emumail.com>

Save the distribution to a temporary directory and unpack the distribution file.

- 2) Run the install script.
Move to the directory unpacked from the distribution file and execute the installation script.

./emuinstall.sh

Ensure that you follow all directions included in the install script.

- 3) Testing the CGI program.
Now it is time to test the program. Move to the */home/EMU/webmail/html/* directory and execute the following command:

perl -c emumail.cgi

If you get a Perl error, you need to make sure that *emuinstall.sh* correctly installed all the modules (running *emuinstall.sh* again won't hurt), that the file permissions are correct, and that the *page root=* line in the *init.emu* file properly points to the absolute path of your *data/* directory. After you have checked all of the above, see the Troubleshooting section of this document, visit the EMUMAIL web site at <http://www.emumail.com/support/>, or email a support technician at emu-tech@EMUMAIL.com.

4) View EMU Webmail from a web browser.

The final step is to view the program in action from a web browser. Point a web browser to the location you placed the *html/* directory. If you get a login page for EMU Webmail, and the title bar in your window says "EMU Webmail -Login," then everything is successful and you can continue on to the Configuration section of this document.

If you don't get a login page, or if the title of the login page contains an error, corrective steps will need to be taken, including re-running and double checking the steps in the installation process. See the Troubleshooting section of this document, visit the EMUMAIL web site at <http://www.emumail.com/support/>, or email a support technician at emu-tech@EMUMAIL.com.

2.3. Windows Install Instructions

EMU Webmail for Windows only runs on Windows NT4, 2000, and XP. It will not run on Windows 95, 98, or Me.

To install EMU Webmail on Windows, first run the InstallShield wizard, which will guide you through most of the steps. After running the wizard, there are a few things that need to be done to complete your EMU Webmail installation:

- 1) Configure the *site.emu* file in the *data* directory. The defaults should work for most installations. The most common settings that you may need to change would be:

```
default_pop=pop.mydomain.com  
smtp_host=smtp.mydomain.com
```

- 2) You are now ready to run EMU Webmail. Load the page in your browser using the URL <http://mydomain.com:8080>.

You can start and stop the service (called "EMU Webmail 7.0") in the Services applet of the Control Panel.

2.4. Identifying the Web Server Account

The most accurate way to find out which account your web server is running under is to look at its configuration file (usually *httpd.conf*, for Apache, or through the Server Administration Page for Netscape servers). However, a useful shortcut (though slightly more prone to error) is to look at who the web

server is running under in your systems process list.

Under Linux, FreeBSD, Digital UNIX, SunOS, and other BSD/OSF versions of Unix, type:

```
ps -aux | grep httpd
```

Under Solaris, IRIX, HP-UX, and other System V versions of Unix, type:

```
ps ef | grep http
```

In either case, you will get a listing of processes, similar to this (the first and last columns may give different names):

```
nobody 2487 0.0 1.7 2056 1076 ? S May 10 0:00 httpd
nobody 2488 0.0 1.6 2036 1056 ? S May 10 0:00 httpd
nobody 2489 0.0 1.4 2048 924 ? S May 10 0:00 httpd
```

The left-hand column is what you want to look at. It may say nobody, or it may say something else. The name that you see in this column, regardless of what it is, is the name of the account that your web server is running as. If, when you execute the ps command, nothing is returned, you can't use the shortcut; your web server is probably not named on a variation of "http". You'll have to dig up the configuration file to find out who your web servers user is. Usually Cobalt 4 runs as "httpd", while most standard Apache installs run as "nobody" or "apache".

2.5. Multiple Domains

Recall that the *init.emu* file tells the EMU Webmail engine where to find the *data/* directory. In fact, it is possible to use different *data/* directories when EMU Webmail is accessed through different URLs. A file named *domain.init* in the *html/* directory, containing the line *page_root=dir* will tell the EMU Webmail engine to use the data directory *dir* when accessed through the URL *domain*. For example, if you wish to use the directory *data2/* specifically when the EMU Webmail engine is accessed through the URL *www.mysite2.com*, you should create a file named *www.mysite2.com.init* in the *html/* directory. This file should contain the same thing as your *init.emu* file, except it should give the absolute path of your *data2/* directory. You may create as many *domain.init* files as you wish. The *init.emu* file will serve as a default for domains not covered by a *domain.init* file.

Chapter 3: System Structure

This chapter will familiarize you with the structure of the EMU Webmail system. An installation of EMU Webmail includes the following directories:

The system is divided into two main directories: *data/* and *html/*. The *html/* directory contains all the files that are accessible from the web server, including the main CGI file and all the images and help files. The *data/* directory is composed of user directories, configuration files, and HTML template data that should not be accessible from the web server.

Below, we will take a look at a sample user home directory, located in the */home/EMU/homes* path. A user home directory contains the following files and subdirectories:

<i>DB_VERSION</i>	database version for this account
<i>LASTSESSION</i>	timestamp of last login
<i>files/</i>	files stored by the user
<i>folders/</i>	folder information database files
<i>folders-ordered/</i>	temporary files with folders with sorted contents
<i>foldmap</i>	database mapping messages to folders
<i>messages/</i>	raw mail messages, 1 file per message
<i>tmp/</i>	temporary session information
<i>userdb</i>	main database with option settings and addressbook

Chapter 4: Configuring EMU Webmail

There are many options that can be set in the EMU Webmail configuration files. Most users will need to change few, if any, of the configuration options. The defaults are meant to work in most cases. Some of the options are meaningful only in the licensed version of EMU Webmail. For more information on purchasing EMU Webmail, see our website at <http://www.emumail.com>, or contact sales@emumail.com.

4.1. Configuration File Overview

There are three main configuration files that control the operation of EMU Webmail: *site.emu*, *lang.emu*, and *conf.emu*. The files *site.emu*, and *lang.emu* are contained in your *data/* directory. A copy of *conf.emu* is contained in each interface directory. Additional copies of *lang.emu* may also be located in the interface directories.

- The *site.emu* file is a general configuration file. It is used to control system-wide options, including mail hosts, security, access restrictions, user options, disk quotas, and internal EMU Webmail settings. These settings affect all users.
- The *conf.emu* file focuses on interface-specific options. Since EMU Webmail supports multiple user interfaces, it is possible to set options on a per-interface basis, affecting only users using that particular interface. This is the purpose of the *conf.emu* file. Options such as background and link colors, maximum lengths for different elements, and other interface-related options are in this file. Note: settings in *site.emu* and *conf.emu* can be interchanged, with *conf.emu* taking a higher precedence than *site.emu*.
- The *lang.emu* file contains the phrases that EMU Webmail uses to communicate error, warning, and other messages to the user that may need to be modified because of geographic, aesthetic, or linguistic needs. The *lang.emu* file can be present in each interface, allowing you to have different messages for each interface for example, one for the Spanish version and one for the English. The default *lang.emu* file is located at the same level as the *site.emu* file and will be the default for all interfaces. Per interface *lang.emu* files will override the default.

4.2. Configuration File Format

All configuration files are syntactically similar. A line of the form *NAME=VAL* is used to assign the value *VAL* to the configuration option specified by *NAME*. Each statement must occur on a single line. The order of the statements is not important.

The syntax of the configuration files is case-sensitive, so you will want to make sure that the cases of your options are correct. To define an option, at the beginning of the line type the name of the option, then an equals symbol, and then the value. It is not necessary to enclose the value in quotes. For example:

MSG_AddressEmpty = *Empty Addressbook*.

Many options, such as *mail_hosts* and *allowed_domains* allow multiple values. You may separate multiple values by placing a space, or a comma, between values. Only one delimiting character, space or comma, may be present per option. Other options, such as *default_signature* and almost all entries in *lang.emu*, accept only a single value, but allow you to place spaces between words.

4.3. Special Values in Configuration Files

Some values take special meaning in these three files. The words *false* and *true* (which evaluate to 0 and 1, respectively) are Boolean values which can be used to define options that take on only a *true* or *false* value.

The `\n` character may be used to enter a newline character into a value. For example, if the default signature option was set to:

```
default_signature=EMU\n7.0
```

the resulting signature would be:

```
EMU
7.0
```

You may set similar values in a configuration file by using `[variablename]`. `[variablename]` will get replaced by the value of *variablename*. For example if you set your *default_pop* to be *pop.mydomain.com* then any subsequent occurrence of `[default_pop]` will be replaced with *pop.mydomain.com*.

If an option is left with a blank value or is commented using the `#` character, then that options will be left unset, and at its default internal value. Thus, to have no default signature for any of your users, you would remove anything after the equals symbol: *default_signature=*.

4.4. Adding Custom Variables

If you are customizing your interface, you may sometimes find it useful to create variables whose values are set in *conf.emu*, that you can then access from your interface. To create a variable, simply create an entry for it in your interface's *conf.emu* file. For example, to create a variable called *foo*, and have its value be 1, add this line to *conf.emu*:

```
foo=1
```

You can then access this variable from an HTML template by writing:
`[+ $EMU::c{foo} +]`

4.5. Precedence Among Files

Since EMU Webmail allows for multiple versions of the *lang.emu* and *conf.emu* files, there is an order of precedence for these values. Generally stated, interface specific configuration settings override system-wide configuration

settings. The *lang.emu* file located in an interface's directory contains settings with the highest precedence, and will override any site-wide *lang.emu* settings. Similarly, the *conf.emu* file read in from the current interface would override any settings that are found in the site-wide *site.emu* file. The *site.emu* file is always loaded and serves as a catchall if variables have not been redefined in other configuration files.

4.6. Licensing the Product

To license EMU Webmail so that you can customize the HTML, add your licensing key to the *site.emu* file. Licensing keys may be obtained by contacting <mailto:sales@emumail.com>. The licensed key may be placed anywhere in the *site.emu* file and looks like the following:

```
KEY__contact=user@emumail.com
KEY__expire=970339698
KEY__full license=yes
KEY__version=demo
KEY__revision=7.0
KEY__master=632e0e3f324f2b8239g55cf72cb5cb67
```

4.7. Restricting Access

Through parameters in the *site.emu* or *conf.emu* files, EMU Webmail allows you to control who accesses their email through your server, as well as where they access from. The simplest method of controlling access is to edit the *mail_hosts=* line in *site.emu*. Here, you can enter a space-delimited list of hostnames that you will allow users to check their email on. For example, if you wanted to allow only users with accounts on *pop.x.com* and on *pop.y.com*, your *mail_hosts=* line would look like this:

```
mail_hosts=pop.x.com pop.y.com
```

After defining the list of allowable servers, make *mail_host_input_box=* equal to *false*. This will prevent the "Hostname" box from appearing on the EMU Webmail Login page, and will instead present the user with a select box from which he or she may choose one of your pre-defined mail hosts.

A more sophisticated method of access control is to define a set of mail hosts from which your users may log on to EMU Webmail. This is controlled from the *allowed_domains=* line, where you can place a space-delimited list of domains that you will allow your users to connect from. For example, an *allowed_domains=* line that looks like this:

```
allowed_domains=x.com z.y.com
```

would allow logins only for users connecting from computers with host-names ending in either *x.com* or *z.y.com*, e.g., *abc.x.com*, *tom.z.y.com*. This is useful in an environment such as an Intranet, where you might not want to allow connections from the outside world, but wish to provide access to users within your domain.

4.8. User Quotas

EMU Webmail allows you to set limits on disk usage by your users. Quotas may be enabled for all users, for a set of domains, or on an individual user basis. The *quota_default=* line allows you to set a default quota for all users, measured in bytes. A user whose account does not fall into any other category will have the default quota enforced. To enforce a quota for users with email accounts in a specific domain, you can create a domain-specific quota by adding a line to the *site.emu* file:

quota_@domainname = N

where *domainname* is the domain you wish to place a quota on, and *N* is the size of the quota, in bytes. For example, to place a quota of 2 megabytes on all users whose e-mail accounts are in the domain *x.com*, e.g. *user1@x.com*, or *user2@mail.x.com*, you would enter this line in *site.emu*:

quota_@x.com = 2048000

To make individual exceptions to the system-wide and domain-wide quotas, use the users e-mail address to define their quota, similar to the domain-wide quotas. For example, to give *joe@x.com* a quota of 1 megabyte, add this line to *site.emu*:

quota_joe@x.com = 1024000

Note: Assigning a quota of -1 will disable quotas for that class of users. For example, if you had a system-wide quota of 1 megabyte, but wish to give *jane@x.com* unlimited disk space, you would assign her a quota of -1:

quota_jane@x.com = -1

4.9. site.emu variables

The most common options that affect system-wide operation of EMU Webmail are found in *site.emu*, and are summarized here:

4.9.1. Common Options

- *add_domain_to_user*
Play trick with hostname for user. This will cause the POP user name to have this domain appended to it.
Example: a typical username "joe" and domain "mydomain.com"
With the setting username becomes "joe@mydomain.com"
- *allow_user_config*
If set to true, this will allow a *conf.emu* file to be read in from a users home directory. This allows the configuration to be tweaked down to a per user granularity.
- *allow_user_lang*
Same as the *allow_user_config*, except with the *lang.emu* file.

- *compose_dont_save_outgoing*
Set this to true to disable the saving of outgoing messages by default.
- *cookie_path*
When running multiple copies of EMU Webmail from the same domain name, this will be the path that the cookies will be valid for. This option allows multiple CGI files to be run on the same server and have two concurrent EMU Webmail sessions open at the same time. This is an advanced option and is NOT the preferred method of running multiple interfaces. The default of /, should be sufficient for most installations.
- *daylight_savings*
Set to true if daylight savings time is observed.
- *db_perms*
Default permissions for the user databases.
- *dbm_isa*
Specifies the Perl routine to use for the internal hashes.
- *debug_path*
The absolute directory path where the emumail debug file (emudebug) will be written to.
- *default_autoload*
Sets the default autoload of images in messages.
- *default_encoding*
Default encoding of MIME messages parts.
- *default_filter1...N*
Sets up default filtering options when new users login for the FIRST time. Format is:
- *default_imap_prefix*
Some IMAP servers put a users folder hierarchy in a place other than the message root, such as mail/. This option allows the administrator to set this so their users dont have to.
- *default_outbox_host*
When allowing users to save outgoing messages on an IMAP server, this option tells the EMU Webmail engine which host to save on, if it is different from the POP3 server.
- *default_pop*
Use this option to specify the default POP3 or IMAP server that you will connect to when your users do not enter a hostname. The default is localhost, but you should change this to the correct hostname: default pop=pop.yourdomain.com

- *default_send_host*
You may set the default host to send mail to when mail is addressed only with a user name, and not a full email address. This is useful when you have a lot of users sending a lot of mail to different people on a single mail server.
Default_send_host=popular.x.com
- *dictionary*
CGI to use as the dictionary for word lookups. The word to lookup is appended to the end of the URL.
- *disable_account_persistence*
Similar to *remote_only*, however the entire home directory will be deleted on login and logout.
- *disable_caching*
If not enabled [*cache_headers*] will be placed in all headers or Cache-Control: no-store, private if [*cache_headers*] doesn't exist.
- *disable_dictionary*
An online dictionary can be used on the spell check page. This dictionary resides on a remote location and is queried by passing the unknown word to the end of GET HTTP request. To disable the dictionary support, set this value to true.
- *disable_forwarding*
Support for writing a forwarding instruction for a user's mail is available. Contact EMU Webmail technical support for advanced details.
- *disable_ipaddr_check*
If set to false will mandate that each time during a single session that a user connects to the EMU Webmail server it connects using the same IP address. Causes problems if users come in through a proxy server or other means in which the IP addresses are masqueraded, however adds extra security on closed networks.
- *disable_msword*
When a user receives an MS Word file, it is attempted to be converted to text. Setting this value to true can disable this behavior.
- *disable_outbox*
Disables users from saving outgoing messages. Users can still individually choose to have their outgoing messages saved.
- *disable_waitscreen*
Optionally, when the EMU Webmail engine is delayed, it can present a temporary, animated wait screen. This behavior may be removed by setting this option to false.
- *do_realtime_filter*
When set to true this will automatically filter messages as they

come into the server. Set to false will force the user to press the filter messages button to trigger the filtering.

- *emu_debug*
Setting this option to true forces EMU Webmail to create a log file in the data/ directory or to the directory that *debug_path* points, named emudebug. This file contains an entry for most internal actions EMU Webmail performs during a session. Event logging is very useful for investigating problems with EMU Webmail. Since emudebug can grow very quickly, it is recommended that you leave debugging off when you do not absolutely need it not using it.
- *failed_login_counter*
When set to true, EMU Webmail will keep track of the number of incorrect logins a user has had. This option normally is used in conjunction with the *failed_login_max* option.
- *failed_login_max*
When the total number of incorrect logins a user has had exceeds this number, the user is presented with another URL, possibly directing them to further assistance or to a password retrieval page. The *failed_login_counter* option must be set to true in order for this option to work. Default is 6 attempts.
- *failed_login_url*
If the *failed_login_counter* option is enabled, when a user attempts to login after exceeding the *failed_login_max* they will be taken to the page defined in this option.
- *force_mail_local*
To force that all mail be downloaded from the POP/IMAP server and deleted from the mail spool, set this value to true. Use this with caution, once this is turned on, it cannot be reversed.
- *force_protocol*
To force the EMU Webmail engine to use a particular login protocol, set this to the protocol name.
- *forward_file*
The path to the forward file, if it exists.
- *smtp_auth*
Set to true if your SMTP server requires SMTP authentication. Also, you may set the type of authentication (*smtp_auth_type*) available types: DIGEST-MD5, CRAM-MD5, LOGIN, PLAIN
- *lax_mailbox_sync*
Under IMAP don't synchronize when the mailboxes differ.
- *login_protocol*
Allows you to select which protocols are used by default for logging a user in, and in which order the connections are

attempted. The allowed protocols are pop3 and imap. The protocols are used in the order they appear in the options line, and are delimited with a space.

- *map2pop_DOMAINNAME*
Will map a DOMAIN name to a POP server so the users dont have to remember it.
- *max_time*
Maximum time in seconds that a session can remain inactive and not expire.
- *outbox_protocol*
Allows you to select which method to use to save users outgoing messages, and in which order the methods are attempted. The allowed methods are imap and local. An IMAP outbox allows your users to store their outgoing mail on another server. A local outbox stores outgoing messages in users EMU Webmail home directories.
- *perlsub_user_home*
Perl subroutine to calculate where a users home directory should be. Must return a valid path. This option is useful for placing users home directories on various partitions or remote mount points based on some administrator defined load balancing function. The e-mail address will be given to the sub as a parameter.
- *publisher_name*
The publisher name is the string that is printed in the title bar of the Web browser. The value of this option may be changed to anything you like. By default it is set to EMUMAIL.
- *quota_type*
Setting this value to EMUMAIL will allow the EMU Webmail administrator to program their own routine in the *Custom.pm* module of the data/lib/EMU directory in the *quota_check* subroutine. Other values are ignored. This is useful for implementations where the quota is defined by an outside interface. The *quota_check* routine should return (in : delimited fashion) the amount allowed, amount used, and a percentage of usage. EMU Webmail will compare to the used (in bytes) to the *quota_allowed* variable to determine if the user is over quota.
- *redirect_login*
Set this to a URL where the user will be taken when they logout. Useful to switch to https:// for login. Can add *perlsub_* to the beginning to make it into a Perl subroutine that returns the URL.
- *redirect_logout*
Set this to a URL where the user will be taken when they logout.
- *remote_only*
Setting this to true will disallow the creation of long term storage

on the EMU Webmail server. When a user logs in and logs out their home directories will be deleted. Use this option if you wish to provide occasional use of the webmail services, but don't wish to have long term storage. The user's options and personal settings will be retained on the EMU Webmail server, but the mail messages and folders won't be.

- *smtp_host*
If your outgoing mail server is different from your POP3 server, enter the SMTP server's hostname here.
- *success_login_sub*
Allows the administrator to execute a subroutine in the *Custom.pm* module whenever a user successfully logs in. This will allow the administrator to set their own options, do advanced reporting, or whatever they like. The value of this variable corresponds to an actual Perl subroutine in the *Custom.pm*, and returns no value.
- *timezone*
Designates your time zone. Use the standard abbreviation for your local timezone.
- *umask*
Default umask for created files.
- *use_fastcgi*
Setting to true will enable the FastCGI hooks to speed up program response. Must have a FastCGI compatible web server.
- *locks_on_nfs*
Enable this option if you're using NFS.

4.9.2. User Defaults

The *site.emu* file allows you to set the defaults for many user options. Users may override these defaults, unless you have customized the interface to prevent users from changing the default options. The configurable options are listed here:

- *default_mail*
Allows you to form the outgoing mail address using the different parts of the user's login address. Using a sample e-mail address, joe@mail.x.com, %u represents joe, %h represents mail.x.com, %1 represents com, %2 represents x, and %3 represents mail. The standard form for an e-mail address is %u@%h, which will result in the user's complete e-mail address. A common form is to have %u@%2.%1 (don't forget the period!), which would result in our sample e-mail address above looking like this on outgoing mail: joe@x.com.
- *default_signature*
Allows you to select a default signature to be appended to your

users outgoing mail. Not to be confused with the *MSG_Tagline* option in *lang.emu*, which may not be overridden; the default signature may be overridden by the user. If you have multiple interfaces, you may, at your option, delete this line from *site.emu* and place it in the *conf.emu* file for each of your interfaces, thereby allowing you different signatures for different interfaces.

- *default_real_name*
Allows you to set the default Full Name for new users.
- *default_organization*
Allows you to set the default organization for new users. Useful within Intranets for placing your company name on outgoing mail.
- *default_checkmail*
Sets the default frequency, in seconds, that EMU Webmail checks a users POP3 or IMAP accounts for new mail while the user is at the Index page.
- *default_compose*
The Compose page has a number of header lines that can be hidden or shown as the user wishes. This sets the header lines that are available initially for the user to fill out. Acceptable values for this options include: attach, cc, bcc, from, replyto, and priority. The To header is always available. To have just the Cc line available by default on the Compose page:

default_compose= cc

- *default_address1...N*
Default addressbook entries for new users. The order of an entry is the nickname, then two vertical bars (|), then the full name, then two more vertical bars, and then the e-mail address. Starting with the number 1, you can have as many default entries as you desire, each on its own line.

default_address1=joe|| Joe Smith|| joe@x.com
default_address2=jane|| Jane Doe|| jane@x.com

4.9.3. Basic Interface Configuration

Each directory that contains a user interface (data/iface/normal, for instance), may contain a file named *conf.emu*, which allows different interfaces to specify different settings for certain options.

- *addressbook_length*
The maximum number of characters to print in the Addressbook pull-down menus on the Compose page.
- *default_folder1...N*
Allows you to specify folders that are automatically created when a user logs into EMU Webmail for the first time. For example, to have every new user have a folder, Company Mail, place this entry

in *conf.emu*: *default_folder1=Company Mail*.

- *default_max_messages*
The maximum number of messages to show in a single page. If the user exceeds this number of messages in the current folder, a link to a next page of messages appears in the Index page. If your users receive a large number of messages frequently, it might be advisable to raise the value of this option a bit so that users have to switch pages less often.
- *folder_namelen*
The maximum allowable length for a new folders name. Folder names over this length will be truncated.
- *index_subject_length*
The maximum number of characters to print for the messages subject in the Index page.
- *index_sender_length*
The maximum number of characters to print for the senders address in the Index page.
- *post_login_page*
If this line is entered into the *conf.emu* file, the file specified by this line will be loaded as an intermediary. This is useful for creating a frame-based interface for EMU Webmail.
- *post_login_page=framebase_html_status0...2*
These three variables allow you to specify what is printed for the three possible states of a message. Status0 is an unread message; Status1 is a read message; and Status2 is a message thats been replied to. HTML may be placed here. This is useful for placing small images, such as checkmarks, next to messages.

Chapter 5: Customizing the Interface

EMU Webmail allows you to customize the user interface by modifying the HTML template files, Webmail "skins", the interface configuration file (*conf.emu*), and the language definition file, (*lang.emu*).

Template files are found under Webmail's data directory, inside the iface directory.

/home/EMU/webmail/data/iface/normal

Licenses are installed in the *site.emu* file and will allow all the customizations described in this document.

5.1. Adding a New Interface

EMU Webmail allows multiple user interfaces to be defined and used by one *emumail.cgi* file. All interfaces must be installed as subdirectories of the */home/EMU/webmail/data/iface/* directory.

The default user interface name is *normal*. Changing the *site.emu* parameter *default_interface=* will change the default interface:

default_interface=French

To allow access to additional interfaces, edit the *site.emu* file, adding interfaces to the *ifaces=* line, as well as setting *multi_interfaces=true* to denote that multiple interfaces are in use.

multi_interfaces=true
ifaces=normal French

Note that there is a site-wide *lang.emu* file in the *data/* directory that contains default values for the interface's language (error) messages. These may be overridden by interface-specific *lang.emu* files in the appropriate *iface/* directory.

5.2. Modifying the User Interface

5.2.1. Introduction

The EMU Webmail interface may be customized in several different ways. Without necessarily editing the templates and configuration files themselves, you may customize Webmail using "skins". You also have the ability to modify the templates to meet any special needs you have. Note: as of Webmail 4.5 the interface used embedded perl.

5.2.2. Skins

Skins offer a simple but powerful method of customizing the look and feel of your Webmail interface with relative ease. Additionally, you may offer your users the ability to select from a list of skins on the Options

page, so they may choose a skin they prefer to use with Webmail. Although your users have the ability to change their skin preference, only Webmail administrators may create, remove, or add skins to Webmail, as well as determine which skins are available for use by others.

Each skin is composed of two components, the first of which is CSS (Cascading Style Sheets). CSS is a simple mechanism used throughout many websites for controlling the “style”, specifically the fonts, colors, and spacing, of web documents and is a technology supported by most mainstream web browsers. Each skin has its own stylesheet (with a .css extension) which is located in *html/css/* under your Webmail installation directory. The default skin that ships with Webmail is *skin_webmail_EMU_Original.css*.

The second component of a skin is the images. Although look and feel is primarily created with HTML and CSS, the Webmail interface uses a handful of images to provide the important finishing touches, like the logo and icons. Each skin has its own directory of images which are located in the *html/emuimages/skins/* directory. The images that are part of the default Webmail skin sit in the *EMU_Original/* directory.

5.2.2.1. Skin CSS

A skin’s .css file is composed of numerous CSS “rules” which are referenced by a Webmail template each time that template is rendered in your browser. Each rule is built as follows:

```
SELECTOR.CLASS { PROPERTY: VALUE; }
```

The *SELECTOR* and *CLASS* correspond to an html tag in a template that has a class already set inside it. For example, in the default Webmail skin, *skin_webmail_EMU_Original.css*, among the list of rules you might find:

```
font.DefaultText { font-size: 8px; color: black; }
```

This rule lets the browser know that each time it finds a font tag like ** it should make the text 8 pixels tall and black in color.

Note: Modifying the *SELECTOR.CLASS* of a preset rule in a CSS file may affect all templates as well as other skins. Unless you are prepared to modify all templates and .css files, it is strongly recommended that you do not modify the names of the selectors or classes in any preset rules you find.

You may, of course, edit the *PROPERTY* and *VALUE* of any rule as needed. However, the properties and values you set must be compliant with documented CSS standards in order to work properly in your browser. For a list of valid CSS properties and values, as well as further information on CSS in general, please refer to the W3C documentation on CSS at <http://www.w3.org/Style/CSS/>.

Below is a list of the preset CSS rules that are used in Webmail along with a brief explanation of what each rule affects. Note: To help illustrate what rules do, some explanations refer to colors and images that are used in the default Webmail skin, *EMU_Original*.

- *body*
Applies body styles (specifically background color, background image, page margins, etc.) to all Webmail templates, excluding popup windows.
- *body.popup*
Same as the *body* selector, however it only applies to popup windows.
- *input.DefaultButton*
All <input type=button> and <input type=submit> elements on all pages.
- *input.DefaultTextbox*
All <input type=text> elements on all pages.
- *textarea.Default*
All <textarea> tags on all pages.
- *select.DefaultSelect*
All <select> tags (aka. dropdowns) on all pages.
- *input.DefaultFile*
All <input type=file> elements on all pages.
- *tr.RibbonTop*
Table row that contains the ribbon image in the header.html file. Note: the "ribbon" is the red/blue striped image that borders the top and bottom of each template in the default Webmail skin.
- *tr.RibbonBot*
Table row that contains the ribbon image in the footer.html file.
- *tr.Menu*
Table row (blue) in which the left-hand menu is generated.
- *td.Body*
Table cell (white) to the right of the menu in which the main contents of each page are rendered. Note: This class is assigned to a table cell (rather than a row) which is not compatible with Netscape 4.7 and other browsers than have poor implementation of CSS. Please review section 5.2.2.4 *Browser Compatibility Issues* for further information.
- *tr.Body*
Table cell (white) to the right of the menu in which the main contents of each page are rendered. Note: This class is assigned

to a table row and used sparingly throughout the interface to accomodate Netscape 4.7. To keep the skin consistent, ensure the properties and values set in of *tr.Body* match those of *td.Body*.

- *tr.PreBody*
Table row (blue) that spans the width of the page, just above the menu and body, which contains the page title as well as some specific page functions (when looking at the inbox or viewing a message).
- *tr.PostBody*
Table row (blue) that spans the width of the page, just below the Menu and Body, which contains the account quota when viewing the inbox.
- *font.MenuTitle*
The full name, email address and time/date stamp located at the top of the left-hand menu.
- *font.MenuSpacer*
A generic font class to help space menu elements out where needed.
- *font.MenuSubTitle*
The sub categories listed in the menu if multiple user folders exist and the Inbox tree is expanded.
- *a.MenuLink*
All linked text inside the menu.
- *a.MenuLink:hover*
All linked text inside the menu when the user's mouse passes over.
- *tr.MenuLine*
The short, single-pixel line that separates the product menu from the main menu when Webmail and Web Calendar are integrated.
- *font.Copyright*
The short EMU Webmail copyright blurb below the bottom ribbon.
- *a.Copyright*
Linked text in the copyright.
- *a.Copyright:hover*
Linked text in the copyright when the user's mouse passes over.
- *font.Alert*
The (red) text that is automatically generated on page alerting the user to a problem or a specific action taken. For example, "Message Sent" or "Contact Saved to Addressbook".

- *font.PageTitle*
The text in the PreBody table row that indicates the current page being viewed.
- *font.PageOptions*
The text, used only on a couple of pages, that displays other information in the PreBody and PostBody table rows.
- *font.PageOptionsSpacer*
The text used to help visually separate different page options when they appear on in the PreBody and PostBody table rows.
- *a.PageOptions*
Linked text in the PreBody and PostBody table rows.
- *a.PageOptions: hover*
Linked text in the PreBody and PostBody table rows when the user's mouse passes over.
- *font.Login*
The text on the left-hand side of the login page (where the menu is usually generated) that offers instructions on logging in.
- *font.LoginForm*
The text labels next to the form elements on the login page.
- *font.DefaultText*
Text used throughout the Webmail interface for form labels, explanations, descriptions, etc.
- *font.DefaultHeader*
Text used in only a couple locations to highlight the headers of a description or some other area of importance.
- *font.DefaultTextReq*
Text used for a form label in order to indicate the field is required for submission.
- *font.PopupDefault*
Text used in limited fashion in the Webmail pop-ups.
- *a.Default*
Linked, generic text. For example, linking the description headers on the Options page.
- *a.Default: hover*
Linked, generic text when the user's mouse passes over.
- *tr.BodyGridOutline*
Generates the (dark blue) color of the outlines that divide the data rows on some pages. For example, the Inbox, Addressbook, Filters, etc.

- *tr.BodyButtonBar*
Generates the (medium blue) background color of the button rows on pages that have form buttons.
- *tr.GridHeader*
The (light blue) background color of the header bar that contains text labels. For example, the row that contains the From, Subject, and Date labels at the top of the Inbox.
- *a.GridHeader*
Grid column labels that are linked.
- *a.GridHeader: hover*
Grid column labels that are linked when the user's mouse passes over.
- *font.GridHeader*
The font properties of unlinked labels in each column of the header bar. Note: This class is necessary for compatibility with Netscape 4.7. Please see section 5.2.2.4 for further information.
- *tr.GridData*
The (white) background color of the grid rows that contain listed data.
- *a.GridData*
Listed grid data that is linked.
- *a.GridData: hover*
Listed grid data that is linked, when the user's mouse passes over.
- *font.GridData*
The font properties of unlinked grid data. Note: This class is necessary for compatibility with Netscape 4.7. Please see section 5.2.2.4 for further information.
- *td.LogoWindow*
Controls the properties of the table cell that holds the Webmail logo.
- *tr.SubHeaderBar*
The (medium blue) bars that are used in the body to logically separate different fields or functions. For example, the SubHeaderBar appears on the Contact Editor page.
- *input.ContactsTextbox*
A special class for the <input type="text"> tags on the Contact Editor page.
- *font.MessageText*
The message body of an email on the View Message page.

5.2.2.2. Skin Images

As previously mentioned, the images for each individual skin sit in their own directory. For example, the images that are part of the default Webmail skin (*EMU_Original*) sit in the *EMU_Original/* directory.

Note: The name of the skin and the image directory must match for the skin to work properly.

All skin image directories can be found at *html/emuimages/skins/*.

If you browse through the handful of preset skins that come packaged with Webmail, you will notice each skin image directory contains the same filenames. This is important to note because if the names of your skin images are not correct, Webmail will not be able to find the images it needs when it renders the page.

For example, if you wanted to replace the default *ribbon.gif* image with any other image, that's fine as long as the new image is named *ribbon.gif*.

Below is a list of the images that are used in the preset skins packaged with Webmail:

- *icon_answered.gif*
Appears in the Inbox and email folders. Indicates email that has been read and replied to.
- *icon_attachment.gif*
Appears in the Inbox and email folders. Indicates email in list that has an attachment.
- *icon_new.gif*
Appears in the Inbox and email folders. Indicates email that is new/unread.
- *icon_read.gif*
Appears in the Inbox and email folders. Indicates email that has been read.
- *menu_bullet.gif*
Appears in the left-hand menu when Webmail and Web Calendar are both installed. Indicates which product is currently selected/in use.
- *menu_minus.gif*
Appears in the menu when the Inbox or Option trees are expanded. Indicates the tree is expanded but may be closed.
- *menu_plus.gif*
Appears in the menu when Inbox or Option trees are closed.

Indicates the tree may be expanded.

- *page_background.gif*
The background for all pages, excluding pop-ups.
- *ribbon.gif*
Tiling image that creates the red and blue striped line along the top and bottom of all pages, excluding pop-ups.
- *subscribed_checkmark.gif*
Appears in the Folder Manager. Indicates the folder is “subscribed” to.
- *webmail_logo.gif*
Appears at the top-left of all pages, excluding pop-ups.

5.2.2.3. Creating Your Own Skin

Creating your own skin is a relatively simple process. The creative side is where you will spend most of your time, having fun building that new, cool, custom skin. However, before you start changing colors and images, you will need a couple of minutes for some technical setup.

- 1) From the root Webmail directory, proceed to *html/html/css/*. Locate the default Webmail skin, *skin_webmail_EMU_Original.css*, and make a copy of it. Name the copy *skin_webmail_My_Cool_Skin.css*

Note: All skin file names and directories are case-sensitive, so ensure you label your new skin appropriately and how you would like it to appear in the skin selectbox on the Options page. Also, use underscores to separate the words in your skin name has multiple words. The underscores in your skin name will not be displayed in the selectbox on the Options page.

- 2) From the root Webmail directory, proceed to *html/emuimages/skins/*. Locate the default Webmail skin directory, *EMU_Original/*, and make a copy of it and its contents. Name the copy the same name as your .css file in step one, minus the “skin_webmail” prefix and the “.css” extension. For example, *My_Cool_Skin/*
- 3) From the root Webmail directory, proceed to *data/* and open *site.emu*. Searching for “skins” will move you to a line that contains:

```
# The skins users may choose from  
skins = EMU_Original Deep_Blue_Sea Industrial_Revolution
```

This line specifies which of the installed skins you want to list in the skin selectbox on the Options page.

Add your skin name, in this example, *My_Cool_Skin*, to the end of the list, ensuring the second line does not wrap. Be sure to separate multiple word skin names with underscores rather than spaces. Save

site.emu and exit.

- 4) Login to Webmail. Your new skin should now appear in the Skin selectbox on the Options page. Select your new skin and click "Save" at the top or the bottom of the page. When you save, the page will refresh with your new skin.

Of course, until you make changes to your new skin, it will look just like *EMU_Original*.

- 5) Head back to your new .css skin file and new skin image directory and begin customizing your new skin at your leisure!

5.2.2.4. Browser Compatibility Issues

The HTML and CSS used to create Webmail Skins is compliant with industry standards and specifications recommended by the World Wide Web Consortium (W3C). While any recently released browser should be compliant with these basic standards, there are a handful of older browser versions still in use today that do not fully comply.

In order to use EmuWebmail 5.2.x properly, we strongly recommend you to use Netscape Navigator 6.x or higher or Internet Explore 5.0 or higher. We do not support version 4.x browsers.

- 1) Splitting table row and font properties into their own CSS rules, due to the lack of support for font properties set inside rules assigned to table rows <tr>. In other words, instead of seeing:

```
<tr class="ABC">
  <td>Subject</td>
</tr>
```

... you will see:

```
<tr class="ABC">
  <td><font class="XYZ">Subject</font></td>
</tr>
```

- 2) Restricting CSS rule assignments for tables to table rows <tr> only, due a lack of support for properties assigned to table cells <td>. While this change seems minimal, it has a significant impact on how the background colors are used in the tables throughout the user interface. The effect of this change as well as the only exception to the change is in #3 below.
- 3) Adding a browser check to the header.tmpl which sets a \$BADCSS var if the user's HTTP_USER_DATA (browser identification data) data matches a 4.x browser other than Internet Explorer or Opera. This \$BADCSS var is then used in all templates where the menu transitions to the webmail body. In Webmail 5.0.0, this transition was simply:

```
</td>
<td class="Body">
```

... whereas now the transition is:

```
[$ if ($args->{BADCSS} eq 1) $]
<td valign='top' width='85%'>
[$ else $]
<td class='Body'>
[$ endif $]
```

The function is necessary because if the `class="Body"` was simply removed from the table cell `<td>`, the body area would have the same background color as the menu area. This function instead ensures the `$BADCSS` var does not equal `'1'` before placing the `Body` class inside the table cell. If the var does equal `'1'` the color is inherited and several cell properties (specifically the vertical alignment and column width) must be hardcoded in.

Note to Webmail 5.0.0 users who are upgrading: The changes described above all involved modifications to the `.css` files. If you added your own custom skins, please ensure that you update each skin where appropriate for it to function correctly.

5.2.3. Embperl

Embperl (embedded Perl) is a powerful scripting language that allows you to execute Perl code from within HTML templates.

5.2.3.1. Calling EmbPerl

There are a couple of methods we use in Webmail to call EmbPerl:

- *[-... -] Execute code*

```
[-$a = 5 -] [-$b = 6 if ($a == 5) -]
```

The code between the `[-` and the `-]` is executed. No output will be generated. This is mainly for assignments, function calls, database queries, etc.

- *[+ ... +] Output the result*

```
[+ $a +] [+ $array[$b] +] [+ "A is $a" +]
```

The code between the `[+` and the `+]` is executed and the return value (the value of the last expression evaluated) is output (sent to the browser).

5.2.3.2. Meta Commands

Embperl support some meta commands to control the "program flow"

within the Embperl document. This can be compared to preprocessor commands in C. The meta commands take the following form:

```
[$ <cmd> <arg> $]
```

5.2.3.3. if, elsif, else, endif

The *if* command is just the same as in Perl. It is used to conditionally output/process parts of the document. Example:

```
[$ if $EMU::protocol eq 'imap' $]  
Using IMAP  
[$ elsif $EMU::protocol eq 'POP' $]  
Using POP  
[$ else $]  
Not POP or IMAP, must be Local?  
[$ endif $]
```

5.2.3.4. foreach, endforeach

foreach and *endforeach* both create a loop iterating over every element of an array/list. Example:

```
[$ foreach $msg (1..10) $]  
[+ $msg +]  
[$ endforeach $]
```

5.2.3.5. Embperl Resources

More information on Embperl can be found at:
<http://perl.apache.org/embperl/>

Chapter 6: Troubleshooting FAQ

This chapter contains some Frequently Asked Questions about troubleshooting EMU Webmail. The EMU Webmail FAQ is constantly being updated and expanded. The most current and complete version of the FAQ is available at our web page: <http://www.emumail.com/support/>.

▪ **Why do I receive an error "email address invalid" when I send mail?**

Your mailserver (as defined in `smtp_host`) is probably not accepting mail relay from the server, even though they may be the same machine, or perhaps the smtp server is currently inoperative. In order to open the relay to allow mail to be sent offsite, you need to consult your mailserver documentation for instructions. Be sure to open relay to only the server's IP address, and not to the whole Internet. The software also supports backup smtp servers, `smtp_host` can be defined as a space-separated list of mail server hostnames.

Another cause may be an incorrectly formatted email address in the Options page under Account Info. You can control the default setting of this field using `default_email` in `site.emu`.

▪ **How can I set EMU Webmail to use another SMTP port?**

By default, EMU Webmail will use 25 for SMTP. If you need to change the port number, remove the `#` at the beginning of the `#smtp_port=1025` line in your `site.emu` file and set it to the appropriate port.

▪ **What format does EMU Webmail keep its address books in?**

The addressbook files are in GDBM database format, in a simple hash table. The addressbook addresses are stored alongside other data in the `userdb` file. A hash table entry looks like this: the key is `addresses.NICKNAME` and the value is `EMAIL:FULLNAME` where `NICKNAME`, `EMAIL`, and `FULLNAME` are replaced by appropriate data. The hash table can be viewed by running the `showgdbm` tool available for download from

<http://www.emumail.com/support/webmail/>.

▪ **How do I set up CGI in Apache?**

Add, in `access.conf` or `httpd.conf` (which are probably in `/etc/httpd/conf/`):

```
Options Indexes FollowSymLinks ExecCGI
AllowOverride None
```

Of course, change

```
/PATH/TO/emumail.cgi
```


to the appropriate directory path.

In *srm.conf* or *httpd.conf*, add or uncomment:
AddHandler cgi-script .cgi

▪ **On the first click after I log in, the server returns a "Session Expired" error.**

It's likely that your web browser is not accepting cookies. To fix this:

In Netscape, go to the Edit menu: Edit -> Preferences -> Advanced: Cookies: "Accept all cookies"

In Internet Explorer, go to the Tools menu: Tools -> Internet Properties -> Security -> Custom Level -> Cookies -> Allow Per-session cookies -> "Enable"

Also, check the permissions and ownership on the user homes directory. Doing a recursive *chown* and *chmod* may fix it, giving ownership to the web server user and at least 700 permissions.

▪ **How do I set up a frames-based interface for EMU Webmail?**

In order to implement a frames-based interface after logging, you'll have to set a field in your *conf.emu* file (which is in the directory where your interface is located, e.g., *data/iface/normal/conf.emu*). It should look like this (note that we use *frames.html* as a generic filename{ feel free to choose your own}):

```
post_login_page=frames.html
```

In this manner, your file, *frames.html*, can set up a frames-based interface. If you want to use alternate filenames, you can use the "parse" mode of EMU Webmail to access the templates directly. For example, in the default *msgindex.html* file, you can access the *compose.html* page by placing the following link in the page:

[the comments below should be fixed to follow embedperl syntax, no?]

```
<a href="$url?folder=$folder&passed=compose">
```

Alternately, you can use the Parse mode to access the page directly:

```
<a href="$url  
folder=$folder&passed=parse&variable=compose.html">
```

Either method accomplishes the same thing. Note that you declare the name of the file that you wish to access through the *variable=* name/value pair.

To configure EMU Webmail for multiple interfaces, in *site.emu*, set

```
multi_interfaces=false  
ifaces = iface1 iface2 iface3
```

Place the relevant HTML files you create in *iface/iface1*, *iface/iface2*, etc.

▪ **How do I configure EMU Webmail to use IMAP only?**

In *site.emu* set:

```
login_protocol=imap  
force_protocol=imap
```

▪ **I got an "Improper login sequence" error.**

The webmail software uses a sessionID embedded into the login page to provide tighter login security. This sessionID is generated via javascript. It's likely that your web browser does not have Javascript enabled.

For Netscape: Edit -> Preferences -> Advanced: Enable Java & Enable Javascript

For Internet Explorer: Tools -> Internet Options -> Security -> Custom Level -> Scripting -> Enable

The use of sessionID for tighter security can also be disabled, by setting *disable_sessionID=true* in *site.emu*.

▪ **How do I get emails to show the correct address in the From**

In your *site.emu* file, find the line that says *default_email=something*. In that string, the codes are:

```
%u = username (before @).  
%h = full host (e.g. server.treverton.co.za)  
%1 ... %N = reverse order domain, so if you had  
mail.EMUMAIL.com, %1 would be com, %2 would be EMUMAIL,  
and %3 would be mail
```

If EMU is running on *mail.yourdomain.com*, and you want messages to appear to be coming from *yourdomain.com*, you might want to try something like:

```
default_email = %u@%2.%1
```

or
`default_email = %u@yourdomain.com`

- **How do I restrict what servers users can get mail from in EMU Webmail?**

Edit your *site.emu*, and set:

```
default_pop=popservername.com
allowed_domains=popservername.com
hostname_input_box=false
```

and edit the html template to remove the box.

- **How do I stop EMU Webmail from showing my name and mailhost everytime I log in?**

This can be prevented by modifying the *login.html* template file so as not to retrieve this information from the cookie. Delete the references to `[+ $email +]` from the *login.html* file.

- **How can I send special MIME-types (like HTML) in my messages?**

You can allow users to set the MIME-type of their message by adding an INPUT element to the compose page template. You can make all messages HTML messages HTML messages by adding this tag:

```
<input type=hidden name="message_ct" value="text/html">
```

Or you could put a drop down select box with multiple ones:

```
<select name="message_ct">
<option>text/html<option>text/plain</select>
```

or you can create a open text field:

```
<input type=text name="message_ct" value="text/plain">
```

We recommended not making text/html the default, as this might confuse users who use the caret (arrow) symbols in their messages.

Chapter 7: Additional Support

If you have technical question regarding EMU Webmail that is not answered in this document, please consult the searchable FAQ on our web page:
<http://www.emumail.com>.

If you are unable to find the answer to your question on our web page, you may email support@emumail.com for further information. When contacting us with a problem, please include a detailed description of the problem you are having, including any error messages that EMU Webmail returns.