Password Management
Training & Reference Guide

How does it all work?

OVERVIEW

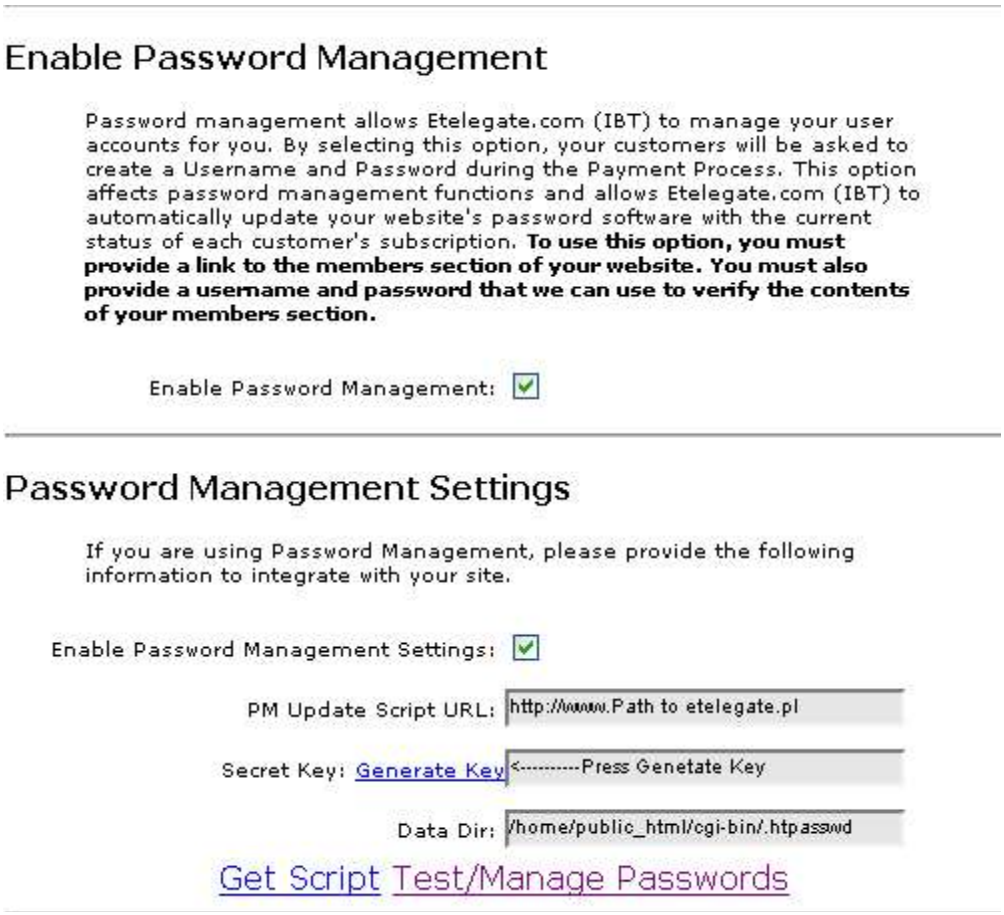
Password Management is one of the proofs of purchase that we currently offer to our clients. A proof of purchase is the method which the customer accesses the client's web site content by means such as hard coding, PIN coding, listed usernames/passwords, etc. Password Management allows the client to use CMI to manage the usernames/passwords on their server making CMI able to handle both billing and access customer service issues.

THE FILES USED FOR PASSWORD MANAGEMENT

Password Management does not work by just by using CMI. A PERL script needs to be uploaded to the client's server to allow the CMI to communicate with the client's server.

The Password Management option consists of three files: **the PERL script**
.htaccess file
.htpasswd file

The PERL script is the only file that was written by us. The other two files, . htaccess and .htpasswd are very common files on Unix servers. These two files are the files that are used for protecting the members' area of a website and can be done by most hosting companies. The following sections contain explanations of these three files and how they work together.



1. Enable Password Management Settings: Add check to enable and settings
PM Update Script URL: <------This is where your etelegate.pl script is located 3.Secret Key: Generate Key <---Use the generate key link
4.Data Dir: <---this is the location of the .htpasswd file
5. After you do these steps press the "Get Script" Link.
This should download to your desktop or root of c:
It will put all the information you added in the script (above).
Add the file to the directory where you specified in step 2.
Add rights to the script.
755 or RWX R_X R_X

THE .HTACCESS FILE
644

This is the file that is placed within the folder that needs to be password protected. This file causes the pop-up box to come up when someone tries to access any file or page within that folder.

For example…..**http://www.mysite.com/join/news/members/index.html**
If the "**members**" folder is protected, I can't access anything below that folder, like the "**index.html**" page without a username/password. I would still be able to access the "**join**" and "**news**" folder without getting a pop-up box because they are above the folder protected by the .htaccess file.

Anyone can create a basic .htaccess file. It can be created by copying the text below, pasting it into notepad and saving the file with the name of .htaccess.

```
AuthName "Password Protected Pages"
AuthType Basic
AuthUserFile /home/public_html/cgi-bin/pass/.htpasswd <---------example

<limit GET POST PUT>
require valid-user
</limit>
```

THE .HTPASSWD FILE
666

This file is just a basic file called ".htpasswd". The only thing in this file is a list of usernames and encrypted passwords. This file has every username/password that will work within the .htaccess file. If the client uses multiple billing companies that share the same members' area, this file will contain the customer's usernames/passwords of the other billing companies.

Below are a few examples of what the usernames/passwords look like:

```
cpl01:s32.t23CQgMcU
test123:pTnd34GFhws
```

The username is followed by an ":**" and then the encrypted password. The usernames in this case are **cpl01** and **test123**. The password is encrypted by the client's server using the date and time of the password creation.**

Think of this file as the key to the .htaccess lock. The .htaccess file knows where to find this file because its location is specified within the .htaccess file. It's referred to as the "AuthUserFile" within the .htaccess section.

THE PERL SCRIPT
755

The PERL script is the file makes communication possible between the CMI and the client's server. This file also writes the username/password that the customer chooses to the .htpasswd file. Below is the section of the script that is modified when the file is configured for a client's server.

```
#!/usr/bin/perl
#vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
#                                                        #
# Name:       Password Management                        #
# Version:    Version 1.2                                #
#          Comments:  Modification of this script is NOT
            Permitted without written Permission
#           from Technical Services.
# Installed:    12/29/05                                  #
#                                                         #
#^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

##########################################################################
##
#
$pwdfile = " /home/public_html/cgi-bin/pass/.htpasswd";    # your password file
$authpwd = "Secret Key";                                   # your authentication code
    ##### DO NOT MODIFY ANYTHING BELOW THIS LINE #####
```

If you look at the above excerpt of the PERL script, you'll notice that the **location of the .htpasswd file** and an **auth code** are located within the script.

- The **location of the .htpasswd** file is required so that the PERL script knows where the customer's username/password needs to be written in order to access the members' area of the web site. The path should match the path located within the .htaccess file, otherwise usernames/passwords will be written to a different file then the one that the pop up box (.htaccess) uses to determine what usernames/passwords will work within the pop up box.
- The **Secret Key** is used to determine that CMI is referring the correct PERL script. Within each Password Management admin in the CMI there is an Secret key and PERL script URL. When Password Management is used, the CMI determines where the script is by this URL and confirms that the Auth code within the script is the same as the one stored within the CMI. Secret Key:Generate Key

*The date in the script is only used to determine when the script was installed on the client's server.

FILE PERMISSIONS

Every file and directory on a Unix server has file permissions. These permissions specify what functions a specific file should have. In order for all the Password Management files to function correctly the Password Management files need to have specific file permissions. Below are the permissions for the Password Management files.

File Name	Permissions
PERL Script	755
.htaccess	644
.htpasswd	666

Every directory and file on the server has an owner and an associated group. The first number represents the 'user' permission level, the second represents the 'group' permission level and the third represents the 'other' permission level. Each group also has a set of permission flags which specify separate read, write and execute permissions. The permission flags have a specific numerical value associated with them.

Read (4)
Write (2)
Execute (1)

You would add up the value for each column and that would give you the permission for that specific group.

- 0 = no permissions whatsoever; this person cannot read, write, or execute the file
- 1 = execute only
- 2 = write only
- 3 = write and execute (1+2)
- 4 = read only
- 5 = read and execute (4+1)
- 6 = read and write (4+2)
- 7 = read and write and execute (4+2+1)

Below is an example of a file that has its permissions set to 756.

USER		**GROUP**		**OTHER**	
Read	(4)	Read	(4)	Read	(4)
Write	(2)			Write	(2)
Execute	(1)	Execute	(1)		

Notice the "user" column equals 7; the "group" column equals 5 and the "other" column equals 6. That is how the 3 digit file permission is determined.
If you view the file permissions of a directory, the permission would be listed as the following: -rwx r-x rw-

Notice that the above string consist of 10 characters, the first position is the directory flag. This position would have a 'd' if it is a directory (folder) and a '-' if it is just a normal file. You will NOT be able to set permissions for a directory. That ability is usually only reserved for the server administrator or the client.

THE CGI-BIN DIRECTORY

A cgi-bin is a directory of a web server where you can store CGI programs. CGI ("Common Gateway Interface") is an interface of data exchange through which sending and reception of data between the browser and programs residents in the server is organized. The cgi-bin is normally used to run PERL scripts. This directory is created by the hosting company or server administrator and is where we upload the Password Management PERL script.

THE WHEREAMI SCRIPTS

These scripts are used for the purpose of determining the absolute path to the directory where the script is located. These scripts are uploaded into the cgi-bin, given file permissions of "755" and executed by going to the URL of the script. For example….http://www.mysite.com/cgi-bin/whereami.pl Absolute paths and URLs are NOT the same thing.

URL - An acronym for "Uniform Resource Locator," this is the address of a resource on the Internet. For example…http://www.google.com

Absolute path - The full path of an object that begins with the root directory. For example…./home/public_html/cgi-bin/whereami.pl

In the above example, within the "home" directory, there is a "public_html" directory. Within the "public_html" directory, there is a "cgi-bin" directory. Finally, within that "cgi-bin" directory is the whereami PERL script file.

Create a Test.php file

Open a text editor add:

```
<?php
echo getcwd();
 ?>
```

Save this as php. Example "test.php"

(Below is the path to your etelegate.pl file. This is an example)

Open browser>add http://yoursitename/cgi-bin/test.php

This will display the **Absolute path**.

TROUBLESHOOTING PASSWORD MANAGEMENT

This section will help you troubleshoot many of the common problems with existing Password Management setups. The first troubleshooting step if the client is having an issue with the Password Management is to create a free account.

If you can not create a free account, you should receive one of the following errors.

ERROR CODES

("501"); # authentication failed (The authcode in the perl script is wrong)
("502"); # invalid request type
("503"); # failed to locate the password file (The password file is missing)
("504"); # failed to open the password file (The permissions on the password file have changed or the file has been moved)
("505"); # specified user already exists (That username is already in the file)
("506"); # specified user doesn't exist (Username not in the file)
("507"); # invalid username (Choice of username is not allowed)
("508"); # invalid password (Choice of password is not allowed)

If you do not receive one of the previous errors, go to the URL of the script to confirm that the script is on the client's server. You should receive one of the following responses:

"script functioning correctly" (The PERL script is functioning correctly)
"invalid form data post" (The PERL script is functioning correctly. An older version of the script)
"Internal Server Error" (The file permissions of the script are incorrect)
"Page can not be displayed" (The script is missing or the URL used is incorrect)

If you can create the free account, but can't log into the members area it is most likely one of the following problems.

1. The .htaccess file is using the wrong location of the .htpasswd file.
2. Our PERL script is writing the username/password to a different location then the one that the .htaccess file is using.
3. The file permissions of either the .htaccess or .htpasswd are incorrect.

OTHER POSSIBLE PROBLEMS

Before you research these possible problems, you should ALWAYS try to create a free account and access the members' area. The Password Management functionality needs to be working correctly before you continue with any troubleshooting.

Problem: Customer is in the CMI under Customer Service but not in PM.
Reason: Error during signup process (any number of reasons)
Solution: Client can take the transaction number and search for the transaction. Find the Password and username, add those to PM under Password management. Contact support if no user/password is listed.

Problem: My member's area is wide open and no longer protected
Reason: Your htaccess file is either missing or has stopped functioning
Solution: Upload a new htaccess file to your member's directory. If this is not an option for you, request assistance from Technical Support.

Problem: New signups are not being added to the password file.
Reason: Either the permissions on the password file have changed or your file was moved or deleted.
Solution: Verify your password file is still there and if it is, check to make sure the permissions on the password file are 666 (RW-RW-RW-). Contact Technical Support of you need assistance with this.
If 501 shows up as the only password, you need to get new .PL script as this is an old script.

Problem: Customer was billed but could not choose a Username and Password.
Reason: Error during signup process (any number of reasons)
Solution: Client can take the Transaction number, Username and Password and add the customer manually to PM under "Etelegate.com>Tools>Password Management". You will probably have to ask the customer what username and password they tried to select. Contact Technical Support if you need assistance with this.

Problem: No customers can join my site.
Reason: Your server has stopped executing our perl script.
Solution: Verify our script is still there and not deleted. It's usually in a directory called "cgi-bin". If the script is there, make sure the permissions are set to 755. Contact Technical Support if you need assistance with this.