

# Setting up Web access using MM list passwords - for Apache

## 3.7. Setting up Web access using MM list passwords - for Apache

The Mailman database stores its user passwords in plain text and by default sends them out to the users once a month, so these passwords are by no means secure. They do however make a convenient database for allowing access to Mailman related web-sites.

Apache uses .htaccess files to limit access to certain sites on a web server. For details, please read:

```
http://httpd.apache.org/docs-2.0/howto/auth.html
```

Basically you place a .htaccess file into a directory on your webserver and in that file point to a password database. The server makes folks login before allowing them access to that part of the site.

The script below takes as input the name of one of your Mailman lists and dumps out each users email address and password. It then plugs those values into a password database /etc/httpd/.htpasswd.<list-name>. You can use a .htaccess file on your website to point to that password database and authenticate against it.

You should be able to read the above link in less than an hour and using the script below as a base, get everything working in another hour.

```
#!/bin/bash
# Script: mm_htaccess
# Updated on Nov 22, 2002 by Linda L.
Julien
# Creates a .htpasswd file based on a
Mailman list
# user = the email address of the user
# password = the Mailman password for
the user
# blank passwords are assigned the
value "password"
#
# Script assumes Mailman was installed to
```

```

the ~mailman/..
#
# Input to the script is the name of the
mailing list
# Output is a .htpasswd file suitable for
web access
#
LIST="`~mailman/bin/list_lists |awk
'{print $1}' | \
        grep -iw ^$1$ 2>/dev/null |tr
[A-Z] [a-z] `"
    if [ "xxx$LIST" = "xxx" ]; then
        echo "mm_htaccess: missing valid
listname from Mailman"
        echo "    mm_htaccess
&lt;list-name&gt;"
        echo "    "
        echo "This program creates a
.htpasswd.&lt;list-name&gt; file in
/etc/httpd"
        echo "It uses the users and passwords
stored in the Mailman list."
        echo "    "
        echo "You must choose a valid Mailman
list on this server:"
        ~mailman/bin/list_lists
        echo "    "
        echo "mm_htaccess command aborted!"
        exit 0
    fi
touch /var/tmp/.htpasswd.$LIST

```

```

CONFIG=~mailman/lists/$LIST/config.db
#
# Isolate the password section of the
config file
PWLN=`~mailman/bin/dumpdb $CONFIG | \
    grep -n "'passwords': {" |cut -f 1
-d: `
~mailman/bin/dumpdb $CONFIG |sed -n
"$PWLN, \ $ p" &gt;/var/tmp/.htlist.$LIST
#
# Retrieve the password info for each
member of the list
for i in `~mailman/bin/list_members $LIST
`
do
    PASS=`grep -i $i /var/tmp/.htlist.$LIST|
head -1 | \
    cut -f2- -d@ | cut -f3 "-d'"`
    if [ "xxx$PASS" = "xxx" ]; then
PASS="password" ; fi
    htpasswd -b /var/tmp/.htpasswd.$LIST $i
$PASS
    # echo $i " : " $PASS
done
#
mv /var/tmp/.htpasswd.$LIST

```

```
/etc/httpd/.htpasswd.$LIST
rm /var/tmp/.htlist.$LIST
```

===

Note that the above script uses the "htpasswd" program which comes with Apache, and is normally installed in /usr/bin/..

Thanks to Jon Carnes for the above script.

===

Ok, building on Linda and Jon's script here, I have come up with the following method of creating the htpasswd databases automagically in the directory /etc/httpd:

1) Here is mm\_make\_htaccess.pl:

```
#!/usr/bin/perl
#####
##
## mm_make_htaccess.pl
##
## Create /etc/httpd/Makefile for Mailman
use.
## version 1.12
## by ges, 10/17/2002
##
## This script is designed to take the
directory names
## from ~mailman/lists and put them into a
Makefile
## which, when run, will create htpasswd
files for
## all your mailman lists, for use in
Apache.
##
```

```
#####  
##  
## Version 1.1 10/15/02: Tests to see if  
$list_dir/$list  
## is a directory or not, before adding the  
entry to  
## the Makefile!  
##  
#####  
##  
## Version 1.11 10/16/02: Update: Made  
printing to the  
## Makefile a little more elegant (thanks,  
Frank!),  
## and now making use of $variables in the  
Makefile.  
##  
#####  
##  
## Version 1.12 10/17/02: Update: Added  
more comments  
## to the code.  
##  
#####  
##  
## First we define where Mailman lists  
reside.  
##  
## Then we define what directory the  
Makefile  
## lives in, as well as where the httpasswd
```

```
files
## will live.
##
## Then we will set a variable to change
for
## admins using Mailman 2.1 or 2.0
##
## Change $configext to be db for 2.0
## or pck for 2.1
##
$list_dir="/etc/mailman/lists";
$file_dir="/etc/httpd";
$configext="pck";
##
##
## First let's open the list directory!
## Once that's done, we'll read the
filenames into @names
## for later use!
##
opendir(LISTDIR,$list_dir) || die("Cannot
Open $list_dir!");
@lists = readdir(LISTDIR);
closedir(LISTDIR);
##
## Now we open the makefile
##
open(FILE,"&gt;$file_dir/Makefile") ||
die("Cannot Open File");
        # This changes print's default to
FILE
```

```

my($oldHandle) = select(FILE);
# Now we start off the Makefile
with two variables
# and some nice spacing.
print "\n\nMAILMAN=$list_dir";
print "\nHTACCESSDIR=$file_dir";
# We ensure that "all" means make
the list htpasswd files.
print "\n\nall: mailmanstuff\n\n";
# H'ok--now we have the work...
print "mailmanstuff: ";
        # This steps through each
entry in @lists, which we read
        # earlier, using readdir
        foreach $list (@lists)
        {
                # Ok--we really
don't want to match
                # current directory
&parent...
                next if ($list eq
".");

```

```
next if ($list eq  
"..");
```

```

# And here we test to make sure that
# the entry is
actually a directory,
# not a file, then
it adds the
# Makefile line to
call each list entry.
if (-d
"$list_dir/$list") {
    print
"\$(HTACCESSDIR)/htpasswd.$list ";
    next;
}
}
# Ok--let's null out $list for
re-use.
$list="";
# Making it pretty...
print "\n\n";
# And here again we step
through...
foreach $list (@lists)
{
    next if ($list eq
".");
    next if ($list eq
"..");
}

```

```

# This time we set up the individual entries
                                # for each list.
                                if (-d $list_dir .
"/" . $list) {
                                print
"\$(HTACCESSDIR)/htpasswd.$list: ";
                                print
"\$(MAILMAN)/$list/config.$configext\n";
                                print
"\t/home/adm/bin/mm_htaccess $list\n\n";
                                next;
                                }
                                }

# Now, let's close the Makefile...
close(FILE)

```

2) Create a shell script-wrapper for ~mailman/bin/newlist (I call it newmaillist):

```

#!/bin/sh
~mailman/bin/newlist
/path-to/mm_make_htaccess.pl

```

2) Create a shell script to run Make in /etc/httpd (I call it htpasswd\_mailman.sh):

```

#!/bin/sh
cd /etc/httpd
make

```

3) Put an entry in root's crontab (this example runs the script every 10 minutes):

```
0,10,20,30,40,50 * * * *  
/path-to/htpasswd_mailman.sh
```

Since this uses make, it only runs on the list databases that have actually been touched since the last make.

I hope people find this useful. 😊

I know that I'm doing the "typical newbie" way of walking a filesystem in Perl, I figured this was easier to port, since it didn't require any modules to be installed in Perl.

If anyone has an idea of getting the perl script to run when lists are created via the GUI interface, that would make this a very handy solution indeed!

Thanks to Jon Carnes and Linda Julien for the original mm\_htaccess script, and the inspiration for this!

*Last changed on Sun Jan 12 04:01:58 2003 by Glenn Sieb*  
Converted from the Mailman FAQ Wizard

This is one of many [Frequently Asked Questions](#).