

sympa_menu.pl script

Scenario: Our organization had a need for a script to assist with list family modifications and to automate some features. We needed to be able to add/remove lists to families, create new lists based on templates, remove a list completely, customize footers, and more. This script required us to centralize our xml config files into /home/sympa/etc/xmlfiles. (Disclaimer: I don't claim to be a perl expert yet, so improvements are welcome)

```
#!/usr/bin/perl
## Script to add new sympa list to specified family.
## by M.Sallee April-08

# Define your sympa home folder here:
$SYMPAHOME = "/home/sympa";
# Add your domain/robot name here
$ROBOT = "servername.edu";

$USER = getpwuid($>);
if ( $USER ne "sympa" ) { die "You need to run me as sympa! Exiting...\n" }

# Display the menu.
print "\nSympa Lists Menu\n";
print "*****\n";
print "Select an option:\n\n";
print "1. Create a new list\n";
print "2. Add a new list to an existing family\n";
print "3. Add multiple lists to an existing family\n";
print "4. Instantiate a new family, or update changed settings on a
family\n";
print "5. Remove a list, keeping an archive\n";
print "6. Remove a list completely\n";
print "x. Exit this utility\n";

my $option = <STDIN>; chomp $option;

if ($option eq 1)
{ print "* Create new list.*\n";
  &get_listname;
  &select_topic;
  &create_xml_file;
  &run_cmds;
}
elsif ($option eq 2)
{ print "* Add new (or existing) list to a family.*\n";
  &get_listname;
  &select_topic; # should be already defaulted
  &create_xml_file;
  &define_family;
  &run_cmds;
}
```

```
elsif ($option eq 3)
{ print "\n* Add multiple lists to a family.*\n";
  print "The choices of family and topic made now will apply to\n";
  print "all lists you add here. Update your sql query in the config\n";
  print "file $SYMPAHOME/etc/families/familyname/config.tt2 before\n";
  print "beginning this process.\n\n";
  &define_family;
  &select_topic;

while ($listname ne "exit")
{
  &get_listname;
  &create_xml_file;
  &run_cmds;
  print "\n\n";
} # end while
}

elsif ($option eq 4)
{ $family_used = 2;
  print "* Create (instantiate) a new family.*\n";
  print " Note: you will need to point to a specific new or\n";
  print " existing list to instantiate a new family.\n";
  print "Enter a new family name to create. \n";
  $FAMILYNAME = <STDIN>; chomp $FAMILYNAME;
  &get_listname;
  &select_topic;
  &create_xml_file;
  &run_cmds;
}

elsif ($option eq 5)
{ &remove_list_close;
}

elsif ($option eq 6)
{ &get_listname;
  &remove_list_purge;
}

elsif ($option eq x)
{ exit 1;
}

else{ die "You didn't enter a readable option. Exiting\n"; }
#####
#####

sub get_listname
{
  ## Get listname and subject to modify in xml file for sympa.
  print "Please enter your list name, or type exit. \n";
  $listname = <STDIN>; chomp $listname;
  if ($listname eq "exit"){ die "Ending program at user request.\n";}
  print "Enter the subject of the list: \n";
  $subject = <STDIN>; chomp $subject;
} # end get_listname sub
```

```
sub select_topic
{
    print "Select the number for the list topic:\n";
    print "1. internal\n2. massmail\n3. statemassmail\n4.
territorymassmail\n";
    print "5. partnersstate\n6. teachersstate\n";
    $topic = <STDIN>; chomp $topic;
    if ($topic eq 1)
        {$topic = internal;}
    elsif ($topic eq 2)
        {$topic = massmail;}
    elsif ($topic eq 3)
        {$topic = statemassmail;}
    elsif ($topic eq 4)
        {$topic = territorymassmail;}
    elsif ($topic eq 5)
        {$topic = partnersstate;}
    elsif ($topic eq 6)
        {$topic = teachersstate;}
    print "You selected: $topic. You may add new topics at
$SYMPAHOME/bin/etc/topics.conf.\n";
};

}
sub create_xml_file
{
## Define xml files for reading, writing
$xmlfile_orig = "$SYMPAHOME/etc/xmlfiles/listcreation.xml";
$xmlfile_new = "$SYMPAHOME/etc/xmlfiles/listcreation-tmp.xml";
if ($family_used eq 2)
{
    $xmlfile_orig = "$SYMPAHOME/etc/xmlfiles/familytemplate.xml";
}
open XMLFILEREAD, "$xmlfile_orig" or die "Cannot read $xmlfile_orig $! ";
## Put the file's lines into an array
@xmllines = (<XMLFILEREAD>);
chomp @xmllines;

close XMLFILEREAD;

## Open xml file to with write access
open XMLFILEWRITE, ">$xmlfile_new" or die "Cannot read $xmlfile_new $! ";

## Search and replace listname, subject, and topics in file:
foreach my $line (@xmllines)
{
    $line =~ s/<listname>[a-zA-Z0-9-
]*</listname>/<listname>$listname</listname>/g;
    $line =~ s/<subject>[A-Za-z0-9-
]*</subject>/<subject>$subject</subject>/g;
    $line =~ s/<topics>[A-Za-z0-9-]*</topics>/<topics>$topic</topics>/g;
```

```
        print XMLFILEWRITE "$line\n";
        print "$line\n";
    } # end of foreach my line
## Close writeable file.
close XMLFILEWRITE;

} # end of create_xml_file subroutine

sub define_family
{
    $family_used = 1;
# Add your own family names here.
print "Select a family: 1 for partners-xx, 2 for teachers-xx\n";
$FAMILYNAME = <STDIN>; chomp $FAMILYNAME;

if ($FAMILYNAME eq 1)
    {$FAMILYNAME = "partners-xx";}
elsif ($FAMILYNAME eq 2)
    {$FAMILYNAME = "teachers-xx";}
print "... Family $FAMILYNAME selected.\n\n";
} # end of define_family sub

## Define sympa commands to add list(s)
sub run_cmds
{
    print "Running commands...\n";

    if ($family_used eq 1)
    {
        # Create a list added to family
        system `'$SYMPAHOME/bin/sympa.pl --add_list $FAMILYNAME --robot $ROBOT --input_file $xmlfile_new`;
        print "\nFamily config files are located at
$SYMPAHOME/etc/families/$FAMILYNAME/config.tt2. \n
Family xml template is at etc/xmlfiles/familytemplate.xml.\n";
    }

    elsif ($family_used eq 2)
    {
        # Instantiate a new family
        print "system `'$SYMPAHOME/bin/sympa.pl --instantiate_family $FAMILYNAME --robot $ROBOT --input_file $xmlfile_new\n`;
        system `'$SYMPAHOME/bin/sympa.pl --instantiate_family $FAMILYNAME --robot $ROBOT --input_file $xmlfile_new`;
        print "\nFamily config files are located at
$SYMPAHOME/etc/families/$FAMILYNAME/config.tt2.\n";
    }
    else
    {
        # Create a generic list without family
        system `'$SYMPAHOME/bin/sympa.pl --create_list --robot $ROBOT --`
```

```
input_file $xmlfile_new`;
    print "\nConfig file located at $SYMPAHOME/expl/$listname.\n";
}

print "Now running postalias $SYMPAHOME/etc/sympa_aliases\n";
system `/usr/bin/sudo /usr/sbin/postalias $SYMPAHOME/etc/sympa_aliases`;
# system `/usr/bin/sudo /usr/bin/newaliases`; # may have to run this
manually. Sympa user needs permissions.
print "postmap /home/sympa/etc/virtual-regexp\n";
system `/usr/bin/sudo /usr/sbin/postmap /home/sympa/etc/virtual-regexp`;
print "Be sure to run newaliases and postfix reload manually.\n";
# system `/usr/bin/sudo /usr/bin/postfix reload`;

# Call add_footer routine
&add_footer;

} # end of run_cmds subroutine

print "\n";

sub remove_list_close
{
    print "Closing and saving list $listname@$ROBOT...\n";
    system '$SYMPAHOME/bin/sympa.pl --close_list=$listname@$ROBOT';
}

sub remove_list_purge
{
    print "Purging list $listname@$ROBOT...\n";
    system '$SYMPAHOME/bin/sympa.pl --purge_list=$listname@$ROBOT';
    print "Deleting files in $SYMPAHOME/expl/$listname...\n";
    system `rm -iR $SYMPAHOME/expl/$listname/*`;
    system `rmdir $SYMPAHOME/expl/$listname`;
} # end remove_list_purge subroutine

sub add_footer
{
    # Define your footers: template and destination
    $footer_template = "$SYMPAHOME/etc/scripts/footer.master";
    $footer_dest = "$SYMPAHOME/expl/$listname/message.footer";

    # Edit the new template to add listname
#    if (-e $footer_template) {die '$!' };
    open FOOTERREAD, "$footer_template" or die "Cannot read
$footer_template $! ";
    # Add file's lines into an array
    @footer_array = (<FOOTERREAD>);
    chomp @footer_array;
    close FOOTERREAD;

    # Open the destination file for writing
```

```

open FOOTERWRITE, ">$footer_dest" or die "Cannot read $footer_dest $! ";
foreach my $line (@footer_array)
{
    $line =~ s/listname/$listname/g;
    print FOOTERWRITE "$line\n";
}

close FOOTERWRITE;
print "Done: $footer_dest updated for list.\n";
} # end of add_footer subroutine

print "Program ended.\n";

```

Sample listcreation.xml file: the default file used for new lists

```

<?xml version="1.0"?>
<list>
    <listname>partners-ar</listname>
    <type>discussion_list</type>
    <subject>Arkansas</subject>
    <status>open</status>
    <shared_edit>editor</shared_edit>
    <shared_read>private</shared_read>
    <language>en_US</language>
    <owner multiple="1">
        <email>postmaster@servername.edu</email>
        <profile>privileged</profile>
    </owner>
    <editor multiple="1">
        <email>postmaster@servername.edu</email>
    </editor>
    <moderator>
        <email>postmaster@servername.eduv</email>
    </moderator>
    <send>editorkeyonly</send>
    <topics>partnersstate</topics>
</list>

```

Sample familytemplate.xml file:

```

<?xml version="1.0"?>
<family>
<list>
    <listname>teachers-ak</listname>
    <type>discussion_list</type>
    <subject>Alaska</subject>
    <status>open</status>
    <shared_edit>editor</shared_edit>

```

```
<shared_read>private</shared_read>
<language>en_US</language>
<owner multiple="1">
    <email>postmaster@servername.edu</email>
    <profile>privileged</profile>
</owner>
<editor multiple="1">
    <email>postmaster@servername.edu</email>
</editor>
<editor multiple="1">
    <email>user2@servername.edu</email>
</editor>
<send>editorkeyonly</send>
<topics>teachersstate</topics>
</list>
</family>
```

Example custom footer:

This text will be placed at the bottom of each outgoing message from sympa. The script will replace the word listname with your list name for each list.

Our mailing lists service. For assistance with this list, contact:
help@servername.edu.
To remove your address from this mailing:
<mailto:sympa@servername.edu?subject=SET%20listname%20NOMAIL>
or send a blank message to sympa@servername.edu with subject: SET listname
NOMAIL

From:
<https://www.sympa.org/> - Sympa mailing list server

Permanent link:
<https://www.sympa.org/contribs/listscreatemodifytool>

Last update: **2009/12/21 19:06**