Scripts

Michael Pobega (pobegam@sunyit.edu)

May 30, 2017

Abstract

Documentation of the scripts worked on during my time at SUNYIT as an nfrastructure intern

Contents

1 File Locations

- **Banner Disbursement Script** avl-sysadm-001:/opt/banner-finaid/disbursement.sh
- **Early Warning System** avl-sysadm-001:/usr/local/adm/scripts/early-warning-system/early_warning_system.pl
- **List Management** avl-syslst-001:

Put the file location here

2 Mailing List

The mailing list generation script is located on **avl-syslst-001** at **/usr/local/adm/scripts/mailer/mailer.pl**. The script creates the file **/etc/postfix/mailer-aliases.cf** which defines aliases and email addresses that receive mail sent to the alias.

%mailers is an associative array that associates a mailing list name (as the left field) with the matching title pattern in LDAP (the right field). The script then iterates over this array and finds all of the users with each title, and adds their email address to the alias file under that mailing list's alias.

```
allstaff: jd_jecko@sunyit.edu,mb_raab@sunyit.edu,[...]
```

Example of a line in mailer-aliases.cf

Lines like the above are generated into the mailer-aliases.cf file, which Postfix respects. When an email arrives for allstaff@sunyit.edu it is forwarded to everyone listed within that alias.

The mailer.pl script is run every evening at 8 PM servertime (as specified in /etc/crontab)

3 Banner Disbursement Script

The banner disbursement script is located on **avl-sysadm-001** at **/opt/banner-finaid/**.

The first thing the script does is check to see if a file named **rpruvfdp_EML.lis** exists in the script's directory. If it doesn't then the script exits, otherwise it continues execution.

```
cd $(dirname "$0")
input='rpruvfdp_EML.lis'
date=`date +_%y%m%d-%H:%M:%S`

# if the input file doesn't exist, exit here
if [ ! -e $input ]; then
    echo "Input file doesn't exist. Exiting" 1>&2
    exit 1
fi
```

Initial preparation

Next, the script iterates over each line in the file and puts all three pipe-separated fields into their own variable, and creates a temporary HTML file that will be e-mailed.

```
while read line; do
    email=‘echo $line | cut -d’|’ -f1‘
    url=‘echo $line | cut -d’|’ -f2‘
    name=‘echo $line | cut -d’|’ -f3‘
    tmp_html=$(mktemp)
```

Grabbing important fields

Next the script will format the email headers. Note that it uses *cat «EOF » $tmp_html* to allow writing multiple lines to the $tmp_html file.

```
    # sets the headers for the email
    cat <<EOF >> $tmp_html
From: "Financial Aid" <finaid@sunyit.edu>
To: "$name" <$email>
Cc: bursar@sunyit.edu
Subject: NOTICE: Disbursement Activity
MIME-Version: 1.0
Content-Type: text/html
EOF
```

Formatting email headers

This is where all the magic happens. The curl call grabs the email template from the SUNYIT website (using whatever URL is provided) and replaces the literal text **[STUDENTNAME]** with the student's name and **[agnEMAIL]** with the student's email in the body of the template, appending it all to the **$tmp_html** file (which if you've been following already has our email headers)

```
# writes the HTML body to $tmp_html
curl -s $url | sed "s/\[STUDENTNAME\]/$name/g" |
       sed "s/\[agnEMAIL\/\]/$email/g" >> $tmp_html
```

Inserting pertinent information into the body of the email

The final step is to actually send the email. We use cat and sendmail to achieve this:

```
    cat $tmp_html | sendmail -i -t
```

Sending the email using ‘sendmail’

The script loops until the input file is out of lines to be read, and then moves the processed file to a subdirectory named **processed**. The files in the processed directory have the date and time appended to them for archival (and possibly legal?) purposes.

3.1 Cronjob

The cronjob is specific in /etc/crontab as follows:

**/10 * * * * itec /opt/banner-finaid/disbursement.sh*

It runs every 10 minutes as the *itec* user, which is also the use that **rpruvfdp_EML.lis** is uploaded as. It is in **/opt/** to avoid permission conflicts in **/usr/local/adm/scripts/**

3.2 Example rpruvfdp_EML.lis

Pipe separated list of values. Three per line. Note that the template URLs are shortened to better fit the width of the page.

```
widricd@sunyit.edu|http://sunyit.edu/1ff72e08|Daniel Widrick
pobegam@sunyit.edu|http://sunyit.edu/1ff72e08|Michael Pobega
email@domain.tld|http://url.to/template|Full Name
```

Example rpruvfdp_EML.lis

4 Early Warning System

The early warning system script is located on **avl-sysadm-001** at **/usr/local/adm/scripts/early-warning-system/early_warning_system.pl**.

This script reads the file **ews.csv** from the script's directory. Each line contains the information for one email to be sent. The order of the fields is as follows: Lastname, Firstname, Fullname, Email, Course, Concerns(7 of them), Recommendations(5 of them), Comments. So fields[0] is the last name, fields[3] is email, and fields 17-$ is comments (because comments can have commas).

The field order is subject to change in the future, but at the time of documentation this is what the csv looks like.

The script grabs the content of a template online and stores it in the $content variable. Then it iterates over each line of the **ews.csv** and replaces the placeholders (such as [STUDENTNAME]) with the actual values from **ews.csv**.

The final step is to send the email; the email header and body are merged together and piped to sendmail using a Perl file handle. **sleep(1)** is used to stop the script from hitting the e-mail server too hard.

5 Kevin's Stuff

5.1 SUNYIT SOAR

This file is owned and run by **banner**, is listed in banner's crontab and it located at **/home/banner/scripts/sunyitsoar.sh**

This script copies all of the SUNYIT_SOAR files out of the $incoming directory, copies them into the $outgoing directory, and moves the original from $incoming into $processed (so that the script doesn't iterate over that specific

file again). If then e-mails Lori and Kevin with the filename and size of each file processed.