

Ham 97A – ASV Control – Shebang 1

Dr. Marc & Rosemary © 230805

- These are advanced programming tips, methods, and locations.
- A script is a file that has operating system commands in a programming language.
 - The common languages are 'shell' with extension *.sh and 'Python' with extension *.py.
 - The most common version of shell is Bash (Bourne-Again Shell).
- The first line of the file tells the OS which interpreter gets invoked. The form is #!/bin/bash or #!/bin/python.
 - # marks a comment. It is pronounced sharp or hash.
 - ! marks the Linux kernel convention. It is pronounced bang. The combination #! becomes a she-bang.
 - /bin is the directory for the interpreter.
 - /python or /bash is the specific interpreter to use.
- The script can be used two ways. Give the file a name 'helloworld'.
 - python helloworld.py, using python ignores the first line because '#' marks a comment.
 - ./helloworld.py is alternative when the file properties were made executable. The interpreter uses the top line.
 - One of the beauties of Linux, Python and Bash is their flexibility, which also makes them more complex.
- Asterisk is the core PBX code. HamVoip uses asterisk. You can invoke either by *rpt.conf* [functions#node].
- HamVoip files store in */usr/local/*.
 - Really cool scripts are included */usr/local/sbin/*
 - For example, */usr/local/sbin/node-ban-allow.sh* allows blacklisting nodes. Likely helpful for a private system.
 - Convert a text string to words.
`/usr/local/sbin/speaktext.sh # speaktext.sh "abc123" node#`
- HamVoip sound message shells are in */usr/local/sbin/sound/*. These are .gsm files like Asterisk messages.
- Asterisk *.gsm messages are in */var/lib/asterisk/sounds/*. Do not include sound extension with command.
`Sfiles="/var/lib/asterisk/sounds" # file location`
`cat $Sfiles/silence/2.gsm > $Outfile.gsm # concatenate multiple files`
`asterisk -rx "rpt localplay $node $Outfile" # run asterisk command call rpt.conf to play this node, no ext'n`
- Convert .wav, etc to .gsm. A sh file also does this.
- Create custom sounds like morse. See *rpt.conf*
- Asterisk files store in */etc/asterisk/*.
 - Rpt.conf, extensions.conf, iax.conf
 - Personal scripts store in */etc/asterisk/local* so they are not overwritten.
- Write a script as a *.txt file or use nano *.sh. Save as mod1.sh
 - Make it executable. Either click properties or chmod.
 - Go to Bash shell. Allows normal read and echo
`cd /etc/asterisk/local #`
`ls -l # ls files with permissions`
`./mod1.sh # execute shell from Linux`
- Executable properties by right click on file. rwx r-x r-x is numerically 755.
`chmod 755 mod1.sh # alternate command line`
- To make an event in asterisk cause the file to execute, put it in *rpt.conf* stanza.
`[functions58000] #controls DTMF ops`
`; Add local ZIP before node for weather also`
`D1=cmd,/usr/local/sbin/saytime.pl 74008 58000 #`
`D2=cmd,python /etc/asterisk/local/ModLed.py -rx 58000 # MOD python script`
- Any bash or python file can be substituted in the cmd instruction.
`D3=cmd,/etc/asterisk/local/mod1.sh -rx 58000 # access Raspberry GPIO`
- Call a shell script from Python.
`subprocess.run(['./ModSayOn.sh'])`
`echo "well we made it"`
`cd /usr/local/sbin`
`speaktext.sh "led on" 58220`
- Life is good. Enjoy!

```
#!/bin/bash
echo "Please enter your name."
#read name #do not use causes wait
#echo "Good day $name, alio"
printf "this is a %s and no %d" string 3
```

```
#!/bin/bash
SW = 23 # gpio.23
if [ -e $FILE ] ; then # File -e(xists)?
    gpio write $SW 1 # turn pin on
else echo "oh no"
fi # endif
#-e $FILE file exists
#-z is if null, -n not null string!
# must space around [] and operators
exit 0 #no errors
```

