Operating Systems II 05/18/2012

OS-Lab-6: Automating Tasks

Services at boot up
In the CLI enter: systemctl enable “servicenamed”.service and press enter.
Replace “servicenamed” with the name of the service you want.
Now restart the system and that service is restarted.

Start / Stop services
To start a service, type: systemctl start “systemnamed.service and that service is started.
To stop a service, type: systemctl stop “systemnamed”.service and that service is halted.

Scheduling
To schedule a task, type: crontab -e – this will enter you into the vi editor where you can write a bash script that will start when you tell it too.
Example of a script:

```
#!/bin/bash
18 12 * * * /path-to-script
```

Shell Control
Doing a task in the shell from the CLI.

Pattern matching
grep is a command that will show all instances of a certain item within a file or directory.
Example: grep -w you <file/dirname> = output of instances of “you” within file/dir.

Redirecting input/output
To redirect the output of a command simply use the “>” and input is “<”.

```
 ls -l myfile.txt > test.txt
```

Here we see the output is redirected from the console into the file test.txt.

Piping data
example pipe command: ls -l | cat will long list a directory into a cat view.

Background processing
A process that will run in the background leaving space available for other tasks.

Schedule a script
Develop a script to execute on a schedule. Use your access to the work server. Document the process and turn in with this work.
I wrote a backup.sh script that will back up my user/public_html folder on my worksatation.
Then a set a crontab that will start the script at 1:30am every morning.
The syntax for the following above is as follows:
The backup.sh script:

```
#!/bin/bash
tar czvf backup.gz /home/user/public_html
mv backup.gz /home/user/docs
```

The crontab is: 30 01 * * * /home/user/bin/bash.sh