Linux Fundamentals-Week 3

Hou, Jue

[jue.hou@helsinki.fi](mailto:jue.hou@helsinki.fi)

Exercise 1

Archive the script in the script directory with bunzip2 and gzip compression:

tar -zcvf compress.tar.gz ./script

tar -jcvf compress.tar.bz2 ./script

The comparison of each archive:

-rwx------ 1 jxhou hyad-all 794 Nov 12 10:03 compress.tar.bz2

-rwx------ 1 jxhou hyad-all 713 Nov 12 10:03 compress.tar.gz

-rwx------ 1 jxhou hyad-all 20480 Nov 12 10:03 non-compress.tar

Bypass tar's shortcut and call bzip2 and gzip explicitly by using a pipeline:

tar -jcv ./script|bzip2> compress.tar.bz2

tar -jcv ./script|gzip> compress.tar.gz

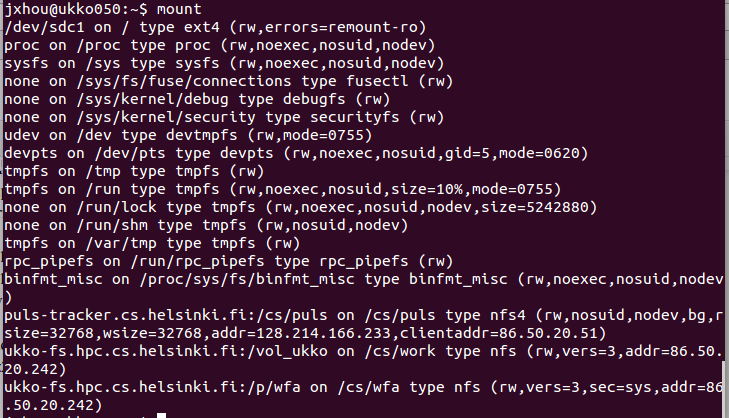
Result:

-rwx------ 1 jxhou hyad-all 1016 Nov 12 10:21 compress.tar.bz2

-rwx------ 1 jxhou hyad-all 817 Nov 12 10:21 compress.tar.gz

Exercise 2

I would like use ukko050 for the tasks of this week



So,”sdc1”,”tmpfs”,”none” may be the local file system

Uncompress and extract the file in a pipeline(without first saving the compressed tarball to disk on local file system):

wget -O- 'https://wiki.helsinki.fi/download/attachments/124126879/lost24-monitor-temps-and-fans-v2.tar.bz2?version=1&modificationDate=1383226184357&api=v2'|tar -jxv

Doing uncompression on ukko and extraction on local file system:

Download the file to ukko050 manually.

Set up proxyConfig file as followed

Host ukko050

User jxhou

HostName ukko050.hpc.cs.helsinki.fi

ProxyCommand ssh jxhou@melkki.cs.helsinki.fi nc %h %p 2> /dev/null

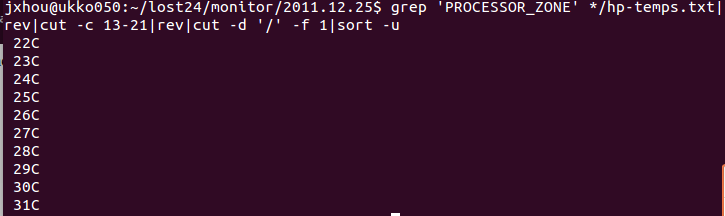
Execute the following command:

ssh -F ./proxyConfig ukko050 'bunzip2 -c lost24-monitor-temps-and-fans-v2.tar.bz2'|tar -xv

Exercise 3

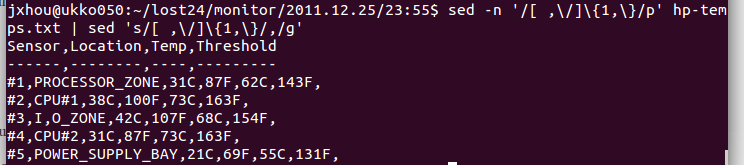
Unique PROCESSOR\_ZONE temperatures (unique temperature) were recorded on 2011.12.25:

grep 'PROCESSOR\_ZONE' \*/hp-temps.txt|rev|cut -c 13-21|rev|cut -d '/' -f 1|sort -u



A sed expression that rewrites consecutive spaces as commas and also rewrites all slashes as commas:

sed -n '/[ ,\/]\{1,\}/p' hp-temps.txt | sed 's/[ ,\/]\{1,\}/,/g'



Command “sed -n '/[ ,\/]\{1,\}/p' hp-temps.txt” shows the lines include consecutive spaces and consecutive slashes. Since there will only be one slash appear one time, this command should also work in this case. Its ouput is on the screen.

Command “sed 's/[ ,\/]\{1,\}/,/g'” replace all the consecutive spaces and slashes to commas.

By using stdout, the command mentioned above should be able to turn the plain text file to a file with CSV look as what is required.

Exercise 4

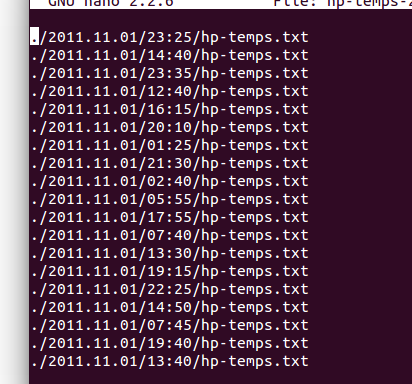
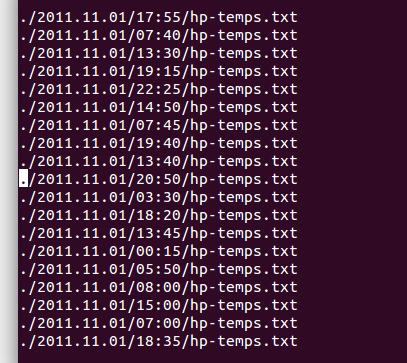
The result of command “ls \*/\*/hp-temps.txt”:

jxhou@ukko050:~/lost24/monitor$ ls \*/\*/hp-temps.txt

-bash: /bin/ls: Argument list too long

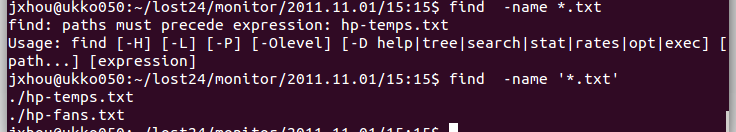
Use find to get the location of each hp-temps.txt file from November, 2011(Part of output):

find ./2011.11.\*/\* -name hp-temps.txt>hp-temps-2011.11.txt

An example where the first type (1.) fails but the second still works:

Find the all the text file in 2011.11.01/15:15/ by using find in two ways, output is as followed:



The first approach failed and the second still worked.

Exercise 5

The script for generating a corresponding -hipstah.jpg version using the code provided with the input of a directory:

#!/bin/bash

for arg in $(find $1 -name '\*.jpg')

do

inputfile=$arg

prefix=${inputfile%.jpg}

outputfile=$prefix-hipstah.jpg

convert -sepia-tone 60% +polaroid $inputfile $outputfile

done

Exercise 6

The script for finding the file (and temperature) which contains the maximum processor temperature from November, 2011:

Enter the /path/to/lost24/monitor/

Run the following script with one parameter ./2011.11.\*

#!/bin/bash

maxfile=./2011.11.01/00:00/hp-temps.txt

prev\_value=$(grep 'PROCESSOR\_ZONE' $maxfile |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

curr\_value=$(grep 'PROCESSOR\_ZONE' $maxfile |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

for arg in $(find $1 -name 'hp-temps.txt')

do

if [ -f $arg ]

then

curr\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

if [ $prev\_value -lt $curr\_value ]

then

maxfile=$arg

prev\_value=$(grep 'PROCESSOR\_ZONE' $arg |rev|cut -c 13-21|rev|cut -d 'C' -f 1)

fi

done

echo $maxfile $prev\_value