Week 3 study diary

Exercises:

1. TAR.

   Filesizes of a tarball and both compression methods:

   Command for tarball

   ```
   fuksi@dhcp-asv-103:~$ tar cf tarball.tar bin/*sh
   ```

   Command using gzip compression

   ```
   fuksi@dhcp-asv-103:~$ tar czf gzip_file.tar.gz bin
   /*.sh
   ```

   Command using bzip2 compression

   ```
   fuksi@dhcp-asv-103:~$ tar cjf bzip2_file.tar.bz2
   bin/*.sh
   ```

   Size comparison

   ```
   fuksi@dhcp-asv-103:~$ ls -c -sh *tar *gz *bz2
   ```

```
 4.0K bzip2_file.tar.bz2
 4.0K gzip_file.tar.gz
 20K tarball.tar
```

Bypassing tar’s shortcut

```
 fuksi@dhcp-asv-103:~$ tar cf - bin/*sh | gzip > ex
plicit_gzip_file.gz

 fuksi@dhcp-asv-103:~$ tar cf - bin/*sh | bzip2 > e
xplicit_bzip2_file.bz2

 fuksi@dhcp-asv-103:~$ ls -c -sh *tar *gz *bz2
 4.0K explicit_bzip2_file.bz2  4.0K bzip2_file.tar.
bz2
 20K tarball.tar 4.0K explicit_gzip_file.gz 4.0K gz
ip_file.tar.gz
```

2. Local and network file systems

I’m using my student laptop for this weeks exercises

```
 fuksi@dhcp-asv-103:~$ hostname
 dhcp-asv-103
```

Filesystem on my laptop seems to be ext4 (/home)

```
 fuksi@dhcp-asv-103:~$ mount
 /dev/sda2 on / type ext4 (rw,errors=remount-ro)
 proc on /proc type proc (rw,noexec,nosuid,nodev)
 sysfs on /sys type sysfs (rw,noexec,nosuid,nodev)
 none on /sys/fs/cgroup type tmpfs (rw)
 none on /sys/fs/fuse/connections type fusectl (rw)
 none on /sys/kernel/debug type debugfs (rw)
 none on /sys/kernel/security type securityfs (rw)
 udev on /dev type devtmpfs (rw,mode=0755)
 devpts on /dev/pts type devpts (rw,noexec,nosuid,g
id=5,mode=0620)
 tmpfs on /run type tmpfs (rw,noexec,nosuid,size=10
%,mode=0755)
 none on /run/lock type tmpfs (rw,noexec,nosuid,nod
ev,size=5242880)
 none on /run/shm type tmpfs (rw,nosuid,nodev)
 none on /run/user type tmpfs
 (rw,noexec,nosuid,nodev,size=104857600,mode=0755)
 none on /sys/fs/pstore type pstore (rw)
 /dev/sda6 on /home type ext4 (rw)
 /dev/sda1 on /boot/efi type vfat (rw,utf8,umask=00
7,gid=46)
 systemd on /sys/fs/cgroup/systemd type cgroup

 (rw,noexec,nosuid,nodev,none,name=systemd) gvfsd-f
use on/run/user/1000/gvfs
 type fuse.gvfsd-fuse (rw,nosuid,nodev,user=fuksi)
```

Fetch and extract. Command for uncompression and extraction:

```
     fuksi@dhcp-asv-103:~$ curl https://wiki.helsin
ki.fi/download/attachments/124126879/lost24-monitor
-temps-and-fans-v2.tar.bz2?version=1&modificationDa
te=1383226184357&api=v2 | tar xvj
```

Doing your business somewhere else

Using proxyConfig settings given in material

```
 Host ukko161
 User          rairanta
 HostName      ukko161@hpc.cs.helsinki.fi
 ProxyCommand  ssh rairanta@melkki.cs.helsinki.fi n
c %h %p 2> /dev/null
```

i kept getting error (tried with different nodes too)

```
 ssh_exchange_identification: Connection closed by
remote host
```

, so i used proxysettings that i set up before to ukko. These are settings in my student laptop in config file:

```
 Host laitos shell.cs.helsinki.fi
```

```
 HostName shell.cs.helsinki.fi
 User rairanta

 Host ukko ukko161.hpc.cs.helsinki.fi
 HostName ukko161.hpc.cs.helsinki.fi
 user rairanta
 ProxyCommand ssh -q laitos nc -q0 ukko161.hpc.cs.h
elsinki.fi 22
```

Essentially these settings do the same thing. e.g. ls from ukko is

```
 ssh ukko "ls"
```

Tarball uncompression on node but extraction to own host. Had big problems with this and my computer didn't accept anything. Ended up using following, which i'm quite sure is not the right way:

```
 ssh ukko "curl https://wiki.helsinki.fi/download/a
ttachments/124126879/lost24-monitor-temps-and-fans-
v2.tar.bz2?version=1&modificationDate=1383226184357
&api=v2" | tar xj
```

3. GREP And CUT

Command and values for required PROCESSOR_ZONE temperatures

```
 fuksi@dhcp-asv-103:~/lost24/monitor/2011.12.25$ gr
ep -r -h "PRO" ~/lost24/monitor/2011.12.25/*/hp-tem
ps.txt | cut -c32-34 | sort -u

 22C
 23C
 24C
 25C
 26C
 27C
 28C
 29C
 30C
 31C
```

If the idea was to find a values that are truly unique and have only one instance, there isn’t any. The command for that would be:

```
 fuksi@dhcp-asv-103:~/lost24/monitor/2011.12.25$ gr
ep -r -h "PRO" ~/lost24/monitor/2011.12.25/*/hp-tem
ps.txt | cut -c32-34 | sort | uniq -u
```

Don’t run with the scissors.

Sed expression in a script file:

```
 #!/bin/bash

 # This script rewrites consecutive spaces and all
slashes as commas so your file looks like a CVS fil
e.

 sed -r 's_[ ]{2,}|\/_,_g' "$@"
```

4. Too long to read. Failing ls:

```
 fuksi@dhcp-asv-103:~/lost24/monitor$ ls */*/hp-tem
ps.txt
 bash: /bin/ls: Argumenttilista on liian pitkä
```

Using find. I will not put the output here, since it’s 8640 lines long, but the command is following:

```
fuksi@dhcp-asv-103:~$ find lost24/monitor/2011.11*
-name 'hp-temps.txt'
```

Escape as a true friend. Constructing an example where find without quotes don’t work, but with quotes it does. I navigated

to directory that has two .txt files in it and now the wildcard expansion can be seen:

```
 fuksi@dhcp-asv-103:~/lost24/monitor/2011.12.25/02:
00$ find . -name '*.txt'
 ./hp-fans.txt
 ./hp-temps.txt
 fuksi@dhcp-asv-103:~/lost24/monitor/2011.12.25/02:
00$ find . -name *.txtfind: polkujen täytyy olla en
nen lauseketta: hp-temps.txt
 Käyttö: find [-H] [-L] [-P] [-Olevel] [-D help|tre
e|search|stat|rates|opt|exec] [polku...] [lauseke]
```

5. The Immelmann. The script for hipsteryfying a .jpg pictures in folder

```
 #!/bin/bash

 # This script hipsterizes photos in folder that is
 given as a parameter

 folder="$@"
 files="${folder}/*.jpg"

 for inputfile in $files
 do
 prefix=${inputfile%.jpg}
```

```
 outputfile=$prefix-hipstah.jpg
 convert -sepia-tone 60% +polaroid $inputfile $outp
utfile
 done
```

6. Hottest day. Script for finding file and temperature:

```
 #!/bin/bash

 # Script for finding maximum processor temperature
 from November 2011 using lost24/monitor dataset.

 files=~/lost24/monitor/2011.11*/*/*temps.txt
 max=0;
 tempfile=0;

 for file in $files
 do
 temp=$(grep 'PRO' $file | cut -c32-33)
 if [ $max -lt $temp ]; then
 max=$temp
 tempfile=$file
 fi

 done
 echo The maximum processor temperature of November
 2011 is
```

```
 $max and it can be found on $tempfile.
```

Output using this:

```
 The maximum processor temperature of November 2011
 is 29 and it can be  found on /home/fuksi/lost24/m
onitor/2011.11.03/12:05/hp-temps.txt .
```