

Lab 09 - Network File Systems

1. Note the output line of the above command.

```
root@UWS ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.5.239.36 netmask 255.255.0.0 broadcast 10.5.255.255
    ether 02:2f:be:6d:61:82 txqueuelen 1000 (Ethernet)
    RX packets 62 bytes 6157 (6.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8 bytes 1372 (1.3 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 2 bytes 140 (140.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2 bytes 140 (140.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

2. What permissions for user, group and world are given by the command `chmod 777` to the directory `/media/disk1`?

```
[root@UWS disk1]# ls -lah /media | grep disk1
drwxrwxrwx    3 root    root        1.0K Nov 28 13:31 disk1
```

Full!

3. What are the permission settings in octal representation for the newly created file `file.dat`

```
[root@UWS disk1]# ls -lah /media/disk1/file.dat
-rw-r--r--    1 root    root         16 Nov 28 13:31 /media/disk1/file.dat
```

644

4. Explain the settings, `rw` and `no_root_squash`. Use the man pages and the Internet to find out. Why might `no_root_squash` be a dangerous setting for a shared filesystem?

`rw` allows for users to write to files and read from them. `no_root_squash` will allow the root users of remote machines to interact with the filesystem as though they had local root permissions. `root_squash` should be used instead.

5. Which port is used by `nfs`?

```
[root@UWS disk1]# rpcinfo -p | head -1 ; rpcinfo -p $IP | grep 2049
program vers proto  port  service
100003      3    tcp    2049
100003      3    udp    2049
```

Port `2049`, though the system provides no confirmation of this for some reason/

6. Which different protocol types are supported by `nfs` in your system?

TCP and UDP.

7. Has the `nfs` service restarted?

```
[root@UWS disk1]# /etc/init.d/S60nfs restart
Shutting down NFS mountd: OK
Shutting down NFS daemon: OK
Shutting down NFS services: OK
Stopping NFS statd: OK
Starting NFS statd: OK
Starting NFS services: OK
Starting NFS daemon: OK
Starting NFS mountd: OK
```

Yes. At least, it certainly looks that way.

8. Does the `nfs` daemon connect to the same ports after restarting?

Yes. Given that the default port configuration hasn't been reconfigured I see no reason why it wouldn't be the same.

9. Besides `stop` and `start` what other input options does the shell-script `/etc/init.d/S60nfs` accept?

`restart` and `reload`

10. Note the contents of `file.dat` in your log-book

```
[root@UWS imported_nfs]# cat file.dat
Welcome to socomp-09
```

11. Do you have permission to edit the file `file.dat` yourself from the remote system?

Yes, because I have root permissions and `no_root_squash` was configured earlier.

12. In which system configuration file would you include a remote file system if you wanted it to be mounted at boot time?

`/etc/fstab`

13. Note the throughput of your Client/Server system for the `nfs` mount-options `rsiz=1024` and `wsiz=1024`, by performing the above command on your system and also noting the values for `real`, `user` and `sys` time

```
[root@UWS mnt]# time dd if=/dev/zero of=/mnt/imported_nfs/zeroes.dat bs=1k
count=2k
2048+0 records in
2048+0 records out

real    0m1.997s
user    0m0.019s
sys     0m0.238s
```

This version of `dd` doesn't want to show throughput, but given a `real` time of about two seconds it looks ~1024kb/sec.

- real 0m1.997s
- user 0m0.019s

- sys 0m0.238s

14. Why is the 'user' time so small? Please comment.

`user` is the amount of CPU time in seconds used within the process outside the kernel, i.e. in user-mode code., not making system calls.

15. Which activities will cause a non-zero system time?

Any time a system call needs to be executed this will result in non-zero system time - whenever a process needs to spend CPU time in the kernel.

16. Why do you think `rsz` and `wsz` are optional parameters for `nfs mount`, but when mounting an internal device, these parameters are not provided?

`rsz` and `wsz` determine the size of the data chunks passed between a client and a server. These parameters are meaningless for any mount device other than one which exists across a network.

17. Note the output of the `od` command. Explain, referring to the man pages, the meaning of the asterisk that appears in the 2nd line of the output.

```
[root@UWS mnt]# od /mnt/imported_nfs/zeroes.dat
0000000 000000 000000 000000 000000 000000 000000 000000 000000
*
10000000
[root@UWS mnt]# hexdump /mnt/imported_nfs/zeroes.dat
00000000 0000 0000 0000 0000 0000 0000 0000 0000 0000
*
02000000
```

The default output for `od` is as follows:

```
[root@UWS mnt]# od -A o -t oS -w16 /mnt/imported_nfs/zeroes.dat
00000000 000000 000000 000000 000000 000000 000000 000000 000000
*
10000000
```

Note that it doesn't contain the `-v` flag, `output-duplicates` which suppresses duplicate lines with an asterisk `*`.

```
[root@UWS mnt]# od -A o -t oS -w16 -v /mnt/imported_nfs/zeroes.dat
00000000 000000 000000 000000 000000 000000 000000 000000 000000
00000020 000000 000000 000000 000000 000000 000000 000000 000000
00000040 000000 000000 000000 000000 000000 000000 000000 000000
00000060 000000 000000 000000 000000 000000 000000 000000 000000
00000100 000000 000000 000000 000000 000000 000000 000000 000000
00000120 000000 000000 000000 000000 000000 000000 000000 000000
00000140 000000 000000 000000 000000 000000 000000 000000 000000
00000160 000000 000000 000000 000000 000000 000000 000000 000000
-- snip --
```

18. Use `od -vc /mnt/imported_nfs/zeros.dat` as well and comment on the output.

```
[root@UWS mnt]# od -vc /mnt/imported_nfs/zeroes.dat
00000000 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0
00000020 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0
00000040 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0
00000060 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0
00000100 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0
```

The flags `-vc` force the printing of duplicate lines not suppressed by an asterisk, and each character is displayed with backslash escapes rendering them printable, escapable in some script.

19. Comment on the behaviour of the real throughput time for increasing transfer block sizes, indicated by the sets of different `rsize` and `wsizes` values. What is the optimal value for `rsize`, `wsizes` for which the fastest real time throughput is achieved?

r/wsize =	1024	2048	4192	8192	16384	32768
run: 1 (real)	1.978s	1.235s	0.779s	0.566s	0.461s	0.414s
run: 2 (real)	1.993s	1.155s	0.768s	0.561s	0.456s	0.408s
run: 3 (real)	2.043s	1.189s	0.767s	0.559s	0.453s	0.412s
avg:	2.005s	1.193s	0.771s	0.562s	0.46s	0.411s

The optimal size from the above table is `32768b` as it performs with the fastest average time.

20. Comment on the fact, that the system time (`sys`) stays nearly constant

The time it takes to execute the command is constant. The `rsizes` and `wsizes` has no bearing on execution time.