Sisältö

1 Introduction: Exercise 6

In this exercise, we practice iptables and NFS. Instead of moving files around with `scp`, we mount part of the filesystem from KONE2 in KONE1.

Here are links to Ubuntu's documentation about iptables, NFS Setup, and Autofs.

2 Steps

2.1 Make firewall rules with iptables in KONE1

Write rules so that

- Everything is rejected by default,
- All outgoing traffic is allowed anywhere,
- All traffic is allowed from and forwarded to KONE2,
- Incoming ssh and http is allowed from anywhere,
- Forwarded port 8080 is only accessible from JYUnet (130.234.0.0/16). Port 2222 is only accessible from the lonka network (172.21.0.0/16).
- All traffic from any established connections is allowed,
- Log all blocked packets.

Make these rules persistent: install `iptables-persistent` and save your rules in `/etc/iptables/rules.v4`.

Check that everything works, and is loaded correctly in boot. To test that, try connecting to a random port and see if that connection attempt gets blocked and logged by the firewall. In other words, http://KONE1.student.it.jyu.fi should work from your workstation and http://KONE1.student.it.jyu.fi:443 should get blocked and logged.

2.2 Network File System

1. KONE1: install `nfs-common` and `autofs`.
2. KONE2: install `nfs-kernel-server`.
3. Make a folder `/exports` in KONE2 for all things shared (or exported) with NFS. In `/etc/exports`, add the line `/exports <172.21. ...>(rw,crossmnt,no_root_squash)` where `<172.21. ...>` is the IP address of KONE1.
4. Make bind mounts inside `/exports`: this puts part of the filesystem inside the exports. It doesn't copy files, just makes them reappear in a different path [1] – in this case, the exports path. This can be done online with the command `mount --bind /something /exports/something` and as a permanent setting in `/etc/fstab` with a line `/something /exports/something none bind 0 0`.

   Export the paths `/etc/nginx`, `/home`, `/var/log`, `/backups` at least.
5. Run `exportfs -a` to initialize the exports table.
6. In KONE1: You might have to add a direct route to KONE2 through the lonka (`up route add -net 192.168.12N.0 netmask 255.255.255.0 gw 172.21.208.1N` in `/etc/network/interfaces` [2]). Then you can mount the filesystem with the command `mount 192.168.12N.XXX:/exports /kone2`.

[1] A question for those with too much free time: what is the difference between bind mounts and symlinks?

[2] If you installed 18.04 from scratch without version upgrade, you have to do this in a similiar but distinct configuration with **netplan**.

2.3 Autofs

1. In KONE1, edit the file `/etc/auto.master` to inlucde the line `/nfs /etc/auto.nfs`. Then make a new file `/etc/auto.nfs` to have the line `kone2 192.168.12N.XXX:/exports`.

2. Unmount any existing NFS mounts if you have any. Then, restart autofs daemon. Now you should be able to cd to `/nfs/kone2/homes/TUNNUS` and see your home directory in KONE2.

2.4 Logs to KONE2

1. in KONE1, edit `/etc/rsyslog.conf` like this: `*.* @@192.168.12N.XXX:10514`

2. in KONE2, add this to the same file:

```
$ModLoad imtcp
$InputTCPServerRun 10514
```

Check that the logs are coming to KONE2.

2.5 Finish

- Check updates for both VMs.
- Run the checker.