

INFORME PRÁCTICA 1

Julio César Velásquez Cárdenas 1397896

David Sánchez González 1401641

Asignatura: Gestió i Administració de Xarxes

Tareas de administración básicas

Crear dos usuarios y verificar su acceso y definición

```
root@Sysetet:~# adduser user1
Adding user `user1' ...
Adding new group `user1' (1001) ...
Adding new user `user1' (1001) with group `user1' ...
Creating home directory `/home/user1' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for user1
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n]
```

```
root@Sysetet:~# adduser user2
Adding user `user2' ...
Adding new group `user2' (1002) ...
Adding new user `user2' (1002) with group `user2' ...
Creating home directory `/home/user2' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for user2
Enter the new value, or press ENTER for the default
    Full Name []:
    Room Number []:
    Work Phone []:
    Home Phone []:
    Other []:
Is the information correct? [Y/n]
```

Como podemos ver en las capturas de pantalla anteriores, hemos creado dos nuevos usuarios, llamados *user1* y *user2*. Ambos tendrán los ids 1001 y 1002 respectivamente.

Como también se puede ver, sus directorios home tendrán se encontrarán en */home/user1* y */home/user2*.

Recursos de la máquina

```
root@Sysetet:~# lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                 2
On-line CPU(s) list:   0,1
Thread(s) per core:    1
Core(s) per socket:    1
Socket(s):              2
NUMA node(s):          1
Vendor ID:              GenuineIntel
CPU family:             6
Model:                  13
Model name:             QEMU Virtual CPU version 1.5.3
Stepping:               3
CPU MHz:                2099.998
BogoMIPS:               4199.99
Hypervisor vendor:      KVM
Virtualization type:    full
L1d cache:              32K
L1i cache:              32K
L2 cache:                4096K
NUMA node0 CPU(s):     0,1
Flags:                  fpu de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pse36 clflush mmx fxsr sse sse2 syscall nx lm
rep_good nopl eagerfpu pni cx16 hypervisor lahf_lm abm
```

Se dispone de una CPU virtual que cuenta con 2 cores, una arquitectura de 64 bits compatible con 32 bits. Una frecuencia de reloj aproximada de 2.1 GHz. Una cachés L0 y L1 de 32 KB cada una más una caché L3 de 4 KB.

```
root@Sysetet:~# free
              total        used        free      shared  buff/cache   available
Mem:           2052556       106720       1525596         3920        420240       1790448
Swap:           604156           0         604156
```

Una memoria RAM total de 2 GB, de las que al momento de la captura de pantalla hay disponibles alrededor de 1.7-1.8 GB. La máquina virtual también dispone de 0.5 GB de espacio Swap aproximadamente.

Discos, particiones y filesystems

```
root@Sysetet:~# fdisk -l
Disk /dev/sda: 20 GiB, 21474836480 bytes, 41943040 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xffb6168d

Device     Boot    Start        End    Sectors    Size Id Type
/dev/sda1  *                2048  19763199  19761152    9.4G 83 Linux
/dev/sda2                19763200  20971519   1208320    590M 82 Linux swap / Solaris
```

La máquina cuenta con un solo disco de 20 GB. Éste cuenta con un total de dos particiones, una de 9.4 GB montada en *root*, y otra de 590 MB montada para Swap.

```

root@master-1-8:~# df -k -BM
Filesystem      1M-blocks    Used Available Use% Mounted on
udev              992M         0M      992M   0% /dev
tmpfs             201M         3M      198M   2% /run
/dev/sda1        9438M    1736M      7279M  20% /
tmpfs            1003M         0M      1003M   0% /dev/shm
tmpfs             5M          0M         5M   0% /run/lock
tmpfs            1003M         0M      1003M   0% /sys/fs/cgroup
tmpfs            201M         1M       201M   1% /run/user/0
/dev/sr0          1M          1M         0M 100% /media/cdrom0

```

Recursos hardware

```

root@master-1-8:~# lspci
00:00.0 Host bridge: Intel Corporation 440FX - 82441FX PMC [Natoma] (rev 02)
00:01.0 ISA bridge: Intel Corporation 82371SB PIIX3 ISA [Natoma/Triton II]
00:01.1 IDE interface: Intel Corporation 82371SB PIIX3 IDE [Natoma/Triton II]
00:01.2 USB controller: Intel Corporation 82371SB PIIX3 USB [Natoma/Triton II] (rev 01)
00:01.3 Bridge: Intel Corporation 82371AB/EB/MB PIIX4 ACPI (rev 03)
00:02.0 VGA compatible controller: Cirrus Logic GD 5446
00:03.0 Ethernet controller: Realtek Semiconductor Co., Ltd. RTL-8100/8101L/8139 PCI Fast Ethernet Adapter (rev 20)
00:04.0 Ethernet controller: Realtek Semiconductor Co., Ltd. RTL-8100/8101L/8139 PCI Fast Ethernet Adapter (rev 20)
00:05.0 Unclassified device [00ff]: Red Hat, Inc Virtio memory balloon

```

En la captura de pantalla anterior se pueden ver los recursos hardware que tiene nuestra máquina virtual master. En este caso contamos con un controlador USB, dos controladores de Ethernet, de VGA, etc. Básicos para el correcto funcionamiento de la máquina.

Versión del kernel

```

root@master-1-8:~# uname -a
Linux master-1-8 4.9.0-3-amd64 #1 SMP Debian 4.9.30-2+deb9u3 (2017-08-06) x86_64 GNU/Linux

```

Como se puede ver, la máquina funciona con el kernel 4.9.0.3 para arquitectura de 64 bits.

Canvi del host name

```

root@master-1-8:~# cat /etc/hostname
master-1-8

```

```

root@slave1-1-8:~# cat /etc/hostname
slave1-1-8

```

```

root@slave2-1-8:~# cat /etc/hostname
slave2-1-8

```

Para cambiar el nombre de la máquina que utilizamos, tan solo se debe cambiar el archivo `/etc/hostname` y reiniciar la máquina. Una vez reiniciada se aplicará en nombre leído del fichero mencionado.

Interfaces de red

```
root@master-1-8:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 02:00:0a:0a:0a:35 brd ff:ff:ff:ff:ff:ff
    inet 10.10.10.53/24 brd 10.10.10.255 scope global ens3
        valid_lft forever preferred_lft forever
    inet6 fe80::aff:fe0a:a35/64 scope link
        valid_lft forever preferred_lft forever
3: ens4: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 02:00:14:14:14:2c brd ff:ff:ff:ff:ff:ff
    inet 20.20.20.44/23 brd 20.20.21.255 scope global ens4
        valid_lft forever preferred_lft forever
    inet6 fe80::14ff:fe14:142c/64 scope link
        valid_lft forever preferred_lft forever
```

En master tenemos un total de 3 adaptadores, el primero es un adaptador lógico para la red local de la propia máquina. Los otros dos son los adaptadores que conectan a las interfaces **Internet** y **Middle** respectivamente.

Análisis Boot Manager

Los términos *stop*, *terminate* y *undeploy* de OpenNebula tiene diferente significado.

Stop: Para la máquina, pero guardando el estado y todos los recursos siguen asociados con esta agilizando la reanudación del sistema cuando el usuario desee.

Terminate: Apaga el sistema y elimina la VM liberando todos los recursos asociados con esta.

Undeploy: El sistema se apaga y los discos de la VM son transferidos al almacén de datos del sistema.

Esto afecta a que no sería posible usar ningún tipo de orden o comando dentro de la máquina mientras esta misma esté en uno de estos estados.

Como se puede ver en la siguiente captura, el boot manager utilizado es el GRUB version 2.02~beta3-5

```
root@master-1-8:~# grub-install --version
grub-install (GRUB) 2.02~beta3-5
root@master-1-8:~#
```

Este boot manager pasa los parámetros al kernel del Linux en un buffer llamado *kernel command line* que podemos obtener usando el comando *dmesg* que muestra el system log en el momento del booting.

```
[ 0.000000] Kernel command line: BOOT_IMAGE=/boot/vmlinuz-4.9.0-3-amd64 root=UUID=774d7b6e-4e8b-4294-ac60-265769cb4fd3 ro
quiet
```


Para cambiar parámetros del sistema tendríamos que modificar el archivo `/etc/sysctl.conf` y para variables adicionales el `/etc/sysctl.d/` como se ve en la siguiente captura.

```
root@master-1-8:/# cat /etc/sysctl.conf
#
# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf (5) for information.
#
```

Para obtener información sobre el proceso de boot usamos el comando `dmesg`. Podemos ver una parte de todo este *log* en esta captura.

```
root@master-1-8:/# dmesg
[ 0.000000] Linux version 4.9.0-3-amd64 (debian-kernel@lists.debian.org) (gcc version 6.3.0 20170516 (Debian 6.3.0-18) ) #
1 SMP Debian 4.9.30-2+deb9u3 (2017-08-06)
[ 0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-4.9.0-3-amd64 root=UUID=774d7b6e-4e8b-4294-ac60-265769cb4fd3 ro quiet
[ 0.000000] x86/fpu: Legacy x87 FPU detected.
[ 0.000000] x86/fpu: Using 'eager' FPU context switches.
[ 0.000000] e820: BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000000f0000-0x00000000000fffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000001000000-0x0000000007fffdfff] usable
[ 0.000000] BIOS-e820: [mem 0x0000000007fffe000-0x0000000007ffffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fffc0000-0x00000000ffffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000ffff0000-0x00000000ffffffff] reserved
[ 0.000000] NX (Execute Disable) protection: active
[ 0.000000] SMBIOS 2.4 present.
[ 0.000000] DMI: Red Hat KVM, BIOS 0.5.1 01/01/2011
[ 0.000000] Hypervisor detected: KVM
[ 0.000000] e820: update [mem 0x00000000-0x00000fff] usable ==> reserved
[ 0.000000] e820: remove [mem 0x000a0000-0x000fffff] usable
[ 0.000000] e820: last_pfn = 0x7ffff max_arch_pfn = 0x40000000
[ 0.000000] MTRR default type: write-back
[ 0.000000] MTRR fixed ranges enabled:
[ 0.000000] 00000-9FFFF write-back
[ 0.000000] A0000-BFFFF uncachable
[ 0.000000] C0000-FFFFFF write-protect
[ 0.000000] MTRR variable ranges enabled:
[ 0.000000] 0 base 000080000000 mask 3FFF80000000 uncachable
[ 0.000000] 1 disabled
[ 0.000000] 2 disabled
[ 0.000000] 3 disabled
[ 0.000000] 4 disabled
[ 0.000000] 5 disabled
[ 0.000000] 6 disabled
[ 0.000000] 7 disabled
[ 0.000000] x86/PAT: PAT not supported by CPU.
[ 0.000000] x86/PAT: Configuration [0-7]: WB WT UC- UC WB WT UC- UC
[ 0.000000] found SMP MP-table at [mem 0x000f72e0-0x000f72ef] mapped at [ffff9e0a000f72e0]
[ 0.000000] Base memory trampoline at [ffff9e0a00099000] 99000 size 24576
[ 0.000000] BRK [0x10f2e000, 0x10f2efff] PGTABLE
[ 0.000000] BRK [0x10f2f000, 0x10f2ffff] PGTABLE
[ 0.000000] BRK [0x10f30000, 0x10f30fff] PGTABLE
[ 0.000000] BRK [0x10f31000, 0x10f31fff] PGTABLE
[ 0.000000] BRK [0x10f32000, 0x10f32fff] PGTABLE
[ 0.000000] BRK [0x10f33000, 0x10f33fff] PGTABLE
```

Todos los logs son guardados en el fichero `/var/log` como vemos en esta imagen. Cada uno de estos archivos guardan los mensajes de los diferentes cambios ocurridos en el sistema durante el uso de este.

```

root@master-1-8:/# cd /var/log
root@master-1-8:/var/log# ls
alternatives.log      daemon.log            dpkg.log.1           lastlog              user.log             wtmp
alternatives.log.1    daemon.log.1         dpkg.log.2.gz        lightdm              user.log.1           wtmp.1
apt                  daemon.log.2.gz      faillog              messages             user.log.2.gz        Xorg.0.log
auth.log             daemon.log.3.gz      fontconfig.log        messages.1           user.log.3.gz        Xorg.0.log.old
auth.log.1           debug                installer            messages.2.gz        vboxadd-install.log  Xorg.1.log
auth.log.2.gz        debug.1              kern.log              messages.3.gz        vboxadd-install-x11.log Xorg.1.log.old
auth.log.3.gz        debug.2.gz           kern.log.1            syslog               VBoxGuestAdditions.log
btmtp                debug.3.gz           kern.log.2.gz         syslog.1             VBoxGuestAdditions.log.1
btmtp.1              dpkg.log             kern.log.3.gz         syslog.2.gz          VBoxGuestAdditions-uninstall.log
root@master-1-8:/var/log# █

```

Sistema de gestión de paquetes

Para poder ver los paquetes instalados en el sistema, usamos el comando *apt list --installed* y para poder actualizar estos usamos el comando *apt-get upgrade*. La imagen siguiente muestra tan solo una parte de los paquetes instalados en la VM Master.

```

root@master-1-8:~# apt list --installed
Listing... Done
acpi-support-base/oldstable,now 0.142-8 all [installed,automatic]
acpid/oldstable,now 1:2.0.28-1+b1 amd64 [installed,automatic]
adduser/oldstable,now 3.115 all [installed]
adwaita-icon-theme/oldstable,now 3.22.0-1+deb9u1 all [installed,automatic]
anacron/oldstable,now 2.3-24 amd64 [installed,automatic]
apt/now 1.4.7 amd64 [installed,upgradable to: 1.4.9]
apt-listchanges/oldstable,now 3.10 all [installed]
apt-utils/now 1.4.7 amd64 [installed,upgradable to: 1.4.9]
avahi-autoipd/oldstable,now 0.6.32-2 amd64 [installed,automatic]
base-files/now 9.9+deb9u1 amd64 [installed,upgradable to: 9.9+deb9u1]
base-passwd/oldstable,now 3.5.43 amd64 [installed]
bash/oldstable,now 4.4-5 amd64 [installed]
bash-completion/oldstable,now 1:2.1-4.3 all [installed]
bind9-host/now 1:9.10.3.dfsg.P4-12.3+deb9u3 amd64 [installed,upgradable to: 1:9.10.3.dfsg.P4-12.3+deb9u5]
binutils/oldstable,now 2.28-5 amd64 [installed,automatic]
bsdmainutils/oldstable,now 9.0.12+nmul amd64 [installed]
bsdutils/now 1:2.29.2-1 amd64 [installed,upgradable to: 1:2.29.2-1+deb9u1]
busybox/oldstable,now 1:1.22.0-19+b3 amd64 [installed]
bzip2/oldstable,now 1.0.6-8.1 amd64 [installed]
ca-certificates/now 20161130+nmul all [installed,upgradable to: 20161130+nmul+deb9u1]
cloud-guest-utils/oldstable,now 0.29-1 all [installed,automatic]
cloud-image-utils/oldstable,now 0.29-1 all [installed,automatic]
cloud-utils/oldstable,now 0.29-1 all [installed,automatic]
console-setup/oldstable,now 1.164 all [installed]
console-setup-linux/oldstable,now 1.164 all [installed,automatic]
coreutils/oldstable,now 8.26-3 amd64 [installed]
cpio/oldstable,now 2.11+dfsg-6 amd64 [installed]
cpp/oldstable,now 4:6.3.0-4 amd64 [installed,automatic]
cpp-6/now 6.3.0-18 amd64 [installed,upgradable to: 6.3.0-18+deb9u1]
cpuid/oldstable,now 20161201-1 amd64 [installed]
cron/now 3.0pl1-128+b1 amd64 [installed,upgradable to: 3.0pl1-128+deb9u1]
curl/now 7.52.1-5+deb9u6 amd64 [installed,upgradable to: 7.52.1-5+deb9u9]
dash/oldstable,now 0.5.8-2.4 amd64 [installed]
dbus/now 1.10.18-1 amd64 [installed,upgradable to: 1.10.28-0+deb9u1]
dbus-user-session/now 1.10.18-1 all [installed,upgradable to: 1.10.28-0+deb9u1]
dbus-x11/now 1.10.18-1 amd64 [installed,upgradable to: 1.10.28-0+deb9u1]
dconf-gsettings-backend/oldstable,now 0.26.0-2+b1 amd64 [installed,automatic]
dconf-service/oldstable,now 0.26.0-2+b1 amd64 [installed,automatic]
debconf/oldstable,now 1.5.61 all [installed]
debconf-i18n/oldstable,now 1.5.61 all [installed]
debian-archive-keyring/now 2017.5 all [installed,upgradable to: 2017.5+deb9u1]
debianutils/oldstable,now 4.8.1.1 amd64 [installed]
desktop-base/now 9.0.2 all [installed,upgradable to: 9.0.2+deb9u1]

```

Para recuperar espacio del caché de los paquetes podemos usar el comando *apt-get clean*

Procesos y servicios

Usamos el comando *ps* que nos dará la información de los procesos activos en el momento. En las siguientes imágenes veremos los resultados de este comando usando las opciones *-e* y *-u* que nos mostrarán todos los procesos o los procesos en ejecución iniciados por el usuario.


```

root@master-1-8:/# ps -u
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         968  0.0  0.0  14536  1584 tty1      Ss+  16:30   0:00 /sbin/agetty --noclear tty1 linux
root      1157  0.1  2.8 357288 59172 tty7      Ssl+  16:30   0:14 /usr/lib/xorg/Xorg :0 -seat seat0 -auth /var/run/lightdm/root
root      2501  0.0  0.1  19908  3628 pts/0    Ss   16:31   0:00 bash
root      4254  0.0  0.1  38304  3252 pts/0    R+   20:38   0:00 ps -u
root@master-1-8:/#

```

```

root@master-1-8:/# ps -e
  PID TTY          TIME CMD
    1 ?           00:00:01 systemd
    2 ?           00:00:00 kthreadd
    3 ?           00:00:00 ksoftirqd/0
    5 ?           00:00:00 kworker/0:0H
    7 ?           00:00:00 rcu_sched
    8 ?           00:00:00 rcu_bh
    9 ?           00:00:00 migration/0
   10 ?           00:00:00 lru-add-drain
   11 ?           00:00:00 watchdog/0
   12 ?           00:00:00 cpuhp/0
   13 ?           00:00:00 cpuhp/1
   14 ?           00:00:00 watchdog/1
   15 ?           00:00:00 migration/1
   16 ?           00:00:00 ksoftirqd/1
   18 ?           00:00:00 kworker/1:0H
   19 ?           00:00:00 kdevtmpfs
   20 ?           00:00:00 netns
   21 ?           00:00:00 khungtaskd
   22 ?           00:00:00 oom_reaper
   23 ?           00:00:00 writeback
   24 ?           00:00:00 kcompactd0
   26 ?           00:00:00 ksmd
   27 ?           00:00:00 khugepaged
   28 ?           00:00:00 crypto
   29 ?           00:00:00 kintegrityd
   30 ?           00:00:00 bioset
   31 ?           00:00:00 kblockd
   32 ?           00:00:00 devfreq_wq
   33 ?           00:00:00 watchdogd
   34 ?           00:00:00 kswapd0
   35 ?           00:00:00 vmstat
   47 ?           00:00:00 kthrotld
   48 ?           00:00:00 ipv6_addrconf
   84 ?           00:00:00 ata_sff
   85 ?           00:00:00 scsi_eh_0
   86 ?           00:00:00 scsi_tmf_0
   87 ?           00:00:00 scsi_eh_1
   88 ?           00:00:00 scsi_tmf_1

```

En cambio si queremos ver los servicios activos en el momento podemos usar el comando `systemctl list-unit-files --type service --all` que nos mostrará una lista con los servicios y el estado en el que están. Estos estados pueden ser enable (habilitados), disabled (deshabilitados), masked (enmascarados), static (estáticos) y generated (generados).

```

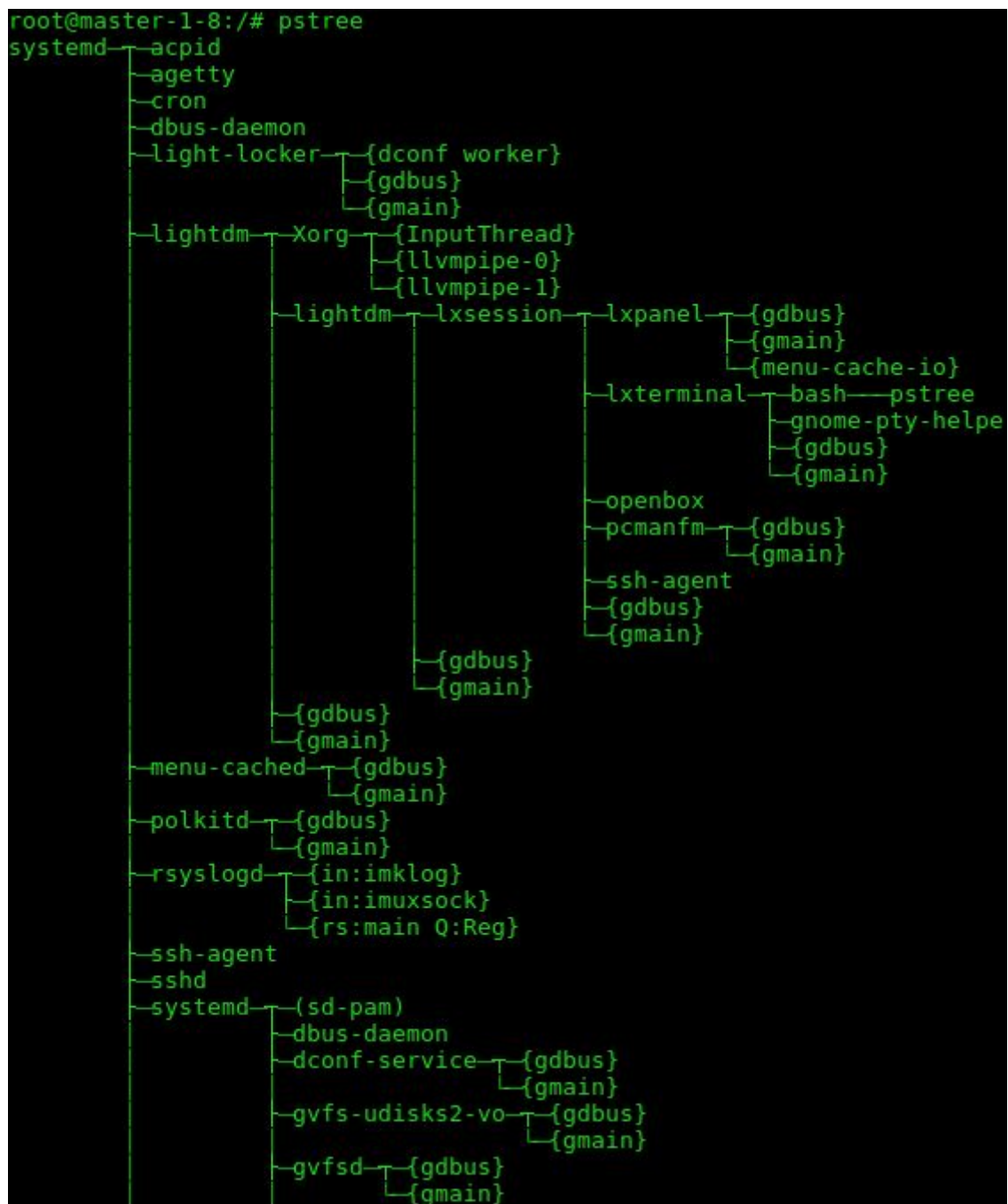
root@master-1-8:/# systemctl list-unit-files --type service --all
UNIT FILE                                     STATE
acpid.service                               disabled
anacron.service                             enabled
apt-daily-upgrade.service                   static
apt-daily.service                           static
autovt@.service                             enabled
bootlogd.service                            masked
bootlogs.service                            masked
bootmisc.service                            masked
checkfs.service                             masked
checkroot-bootclean.service                 masked
checkroot.service                           masked
console-getty.service                       disabled
console-setup.service                       enabled
container-getty@.service                    static
cron.service                                enabled
cryptdisks-early.service                     masked
cryptdisks.service                           masked
dbus-org.freedesktop.hostname1.service      static
dbus-org.freedesktop.locale1.service         static
dbus-org.freedesktop.login1.service          static
dbus-org.freedesktop.network1.service        disabled
dbus-org.freedesktop.resolve1.service        disabled
dbus-org.freedesktop.timedate1.service       static
dbus.service                                 static
debug-shell.service                         disabled
display-manager.service                     enabled
emergency.service                           static
fuse.service                                masked
getty-static.service                         static
getty@.service                               enabled
halt.service                                masked
hostname.service                             masked
hwclock.service                              masked
ifup@.service                                static
initrd-cleanup.service                       static
initrd-parse-etc.service                     static
initrd-switch-root.service                   static
initrd-udevadm-cleanup-db.service            static
keyboard-setup.service                       enabled
killprocs.service                            masked
kmod-static-nodes.service                     static
kmod.service                                 static
lightdm.service                              enabled
module-init-tools.service                     static

```

Systemd

Systemd es un conjunto de daemons de administración del sistema, bibliotecas y herramientas diseñadas como una plataforma para interactuar con el kernel de

Linux. Es el primer daemon en ser ejecutado en el booting y el último en apagarse el sistema, obteniendo el PID 1 como se puede ver cuando usamos el comando *ps tree* que nos da un árbol de los procesos actuales.



Entre los comandos más esenciales de conocer del systemd están el *systemctl status* que nos da el estado del sistema, el *systemctl list-unit* para mostrar los servicios como hemos demostrado antes, *systemctl start servicio* que activa el servicio al momento, *systemctl stop servicio* que detiene el servicio de inmediato y el *systemctl restart servicio* que reinicia el servicio.

```
root@master-1-8:/# systemctl status
● master-1-8
   State: degraded
   Jobs: 0 queued
  Failed: 2 units
   Since: Wed 2020-10-14 16:30:39 BST; 4h 45min ago
  CGroup: /
          └─user.slice
              └─user-0.slice
                  └─session-2.scope
                      ├─2176 lightdm --session-child 14 21
                      ├─2427 /usr/bin/lxsession -s LXDE -e LXDE
                      ├─2463 /usr/bin/ssh-agent x-session-manager
                      ├─2485 openbox --config-file /root/.config/openbox/lxde-rc.xml
                      ├─2490 lxpanel --profile LXDE
                      ├─2491 pcmanfm --desktop --profile LXDE
                      ├─2493 lxterminal
                      ├─2497 /usr/bin/ssh-agent -s
                      ├─2499 light-locker
                      ├─2500 gnome-ptty-helper
                      ├─2501 bash
                      ├─2535 /usr/lib/menu-cache/menu-cached /run/user/0/menu-cached-:0
                      ├─4276 systemctl status
                      └─4277 pager
                  └─user@0.service
                      └─dbus.service
                          ├─2439 /usr/bin/dbus-daemon --session --address=systemd: --nofork --nopidfile --systemd-activation
                          └─2509 /usr/lib/dconf/dconf-service
                      └─gvfs-udisks2-volume-monitor.service
                          └─2518 /usr/lib/gvfs/gvfs-udisks2-volume-monitor
                      └─init.scope
                          ├─2417 /lib/systemd/systemd --user
                          └─2418 (sd-pam)
                      └─gvfs-daemon.service
                          ├─2470 /usr/lib/gvfs/gvfsd
                          ├─2475 /usr/lib/gvfs/gvfsd-fuse /run/user/0/gvfs -f -o big_writes
                          └─2542 /usr/lib/gvfs/gvfsd-trash --spawner :1.4 /org/gtk/gvfs/exec_spaw/0
          └─init.scope
              └─1 /sbin/init
          └─system.slice
```

Red de test

Comenzamos creando las máquinas virtuales que representarán las máquinas A, B y C. Para ello, seguimos las plantillas disponibles en OpenNebula, GAX-large para la máquina A o master, y GAX para las máquinas B y C o slave1 y slave2.

<input type="checkbox"/>	ID	Name	Owner	Group	Status	Host	IPs
<input type="checkbox"/>	4651	slave2_1_8	gax-1-8	GAX	UNDEPLOYED	--	30.30.30.52
<input type="checkbox"/>	4645	slave1_1_8	gax-1-8	GAX	UNDEPLOYED	--	20.20.20.57 30.30.30.48
<input type="checkbox"/>	4634	master_gax_1_8	gax-1-8	GAX	UNDEPLOYED	--	10.10.10.53 20.20.20.44

La máquina A tendrá una interfaz principal **Internet** que le permitirá el acceso a internet más una interfaz secundaria **Middle** que nos servirá para poder conectarla con la máquina B. Esta última tendrá una interfaz principal **Middle** y una secundaria **Deep**, que utilizaremos para conectar con la máquina C. Esta última solo tendrá una interfaz **Deep**.

Como se puede ver en la captura de pantalla anterior, todas las máquinas tienen asignadas unas IPs únicas.

```
if grep -Fq gateway /etc/network/interfaces
then
    exit
fi
```

Para evitar perder los cambios realizados en las máquinas, debemos modificar el fichero `/etc/one-context.d/loc-10-network` añadiendo los comandos de textos que se pueden ver en la captura anterior. De esta forma se guardarán los cambios incluso cuando se reinicien las máquinas o cuando se haga un *redploy* de las mismas.

```
# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1
```

A continuación debemos activar el IP Forwarding en las máquinas A y B. Con esto permitiremos el envío de paquetes entre las tres máquinas. Para ello debemos modificar el archivo `/etc/sysctl.conf` para que quede como se puede ver en la captura anterior.


```
auto lo
iface lo inet loopback

auto ens3
iface ens3 inet static
    address 10.10.10.53
    network 10.10.10.0
    netmask 255.255.255.0
    gateway 10.10.10.1

auto ens4
iface ens4 inet static
    address 20.20.20.44
    network 20.20.20.0
    netmask 255.255.254.0

post-up ip route add 30.30.30.0/24 via 20.20.20.57 dev ens4
```

Una vez activado el IP Forwarding, modificamos el archivo */etc/network/interfaces* para activar el routing entre A y C. Esto se consigue con la última línea que se puede ver en la captura de pantalla.

```
root@master-1-8:~# iptables -t nat -A POSTROUTING -o ens3 -j MASQUERADE
```

Una vez hecho los cambios anteriores, reiniciamos la máquina A para que los cambios se apliquen. Una vez activa de nuevo, activamos el NAT en la máquina A para poder compartir la conexión con el resto de máquinas.

Aunque con esto no sería suficiente, ya que todavía debemos modificar la configuración de la máquina B para que use la máquina A como gateway, así como modificar la configuración de la máquina C para usar como gateway la máquina B.

```
auto lo
iface lo inet loopback

auto ens3
iface ens3 inet static
    address 20.20.20.57
    network 20.20.20.0
    netmask 255.255.254.0
    gateway 20.20.20.44

auto ens4
iface ens4 inet static
    address 30.30.30.48
    network 30.30.30.0
    netmask 255.255.255.0

source /etc/network/interfaces.d/*.cfg
```

Una vez modificado el archivo `/etc/network/interfaces` de la máquina B, podemos reiniciarla y dispondremos de conexión a internet. Como gateway usamos la IP de la interfaz **Middle** de la máquina A, en este caso `20.20.20.44`.

```
auto lo
iface lo inet loopback

auto ens3
iface ens3 inet static
    address 30.30.30.52
    network 30.30.30.0
    netmask 255.255.255.0
    gateway 30.30.30.48

source /etc/network/interfaces.d/*.cfg
```

También deberemos modificar el mismo archivo en la máquina C, que usará como gateway la máquina B, con IP `30.30.30.48` en nuestro caso.

Reiniciamos ambas máquinas y ya dispondremos de conexión a Internet en las mismas, tal como se puede demostrar en las capturas siguientes:

```
root@slave1-1-8:~# wget google.com
--2020-10-14 15:16:02-- http://google.com/
Resolving google.com (google.com)... 172.217.168.174, 2a00:1450:4003:80a::200e
Connecting to google.com (google.com)|172.217.168.174|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://www.google.com/ [following]
--2020-10-14 15:16:02-- http://www.google.com/
Resolving www.google.com (www.google.com)... 172.217.168.164, 2a00:1450:4003:80a::2004
Connecting to www.google.com (www.google.com)|172.217.168.164|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'index.html'

index.html          [ <=> ] 12.91K  --.-KB/s   in 0s

2020-10-14 15:16:02 (74.2 MB/s) - 'index.html' saved [13224]
```

```
root@slave1-1-8:~# ping 158.109.174.1
PING 158.109.174.1 (158.109.174.1) 56(84) bytes of data.
64 bytes from 158.109.174.1: icmp_seq=1 ttl=254 time=1.51 ms
64 bytes from 158.109.174.1: icmp_seq=2 ttl=254 time=1.12 ms
64 bytes from 158.109.174.1: icmp_seq=3 ttl=254 time=1.09 ms
64 bytes from 158.109.174.1: icmp_seq=4 ttl=254 time=1.08 ms
64 bytes from 158.109.174.1: icmp_seq=5 ttl=254 time=1.12 ms
^C
--- 158.109.174.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4005ms
rtt min/avg/max/mdev = 1.086/1.189/1.512/0.166 ms
```

```
root@slave2-1-8:~# wget google.com
--2020-10-15 20:29:53-- http://google.com/
Resolving google.com (google.com)... 172.217.168.174, 2a00:1450:4003:80a::200e
Connecting to google.com (google.com)|172.217.168.174|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: http://www.google.com/ [following]
--2020-10-15 20:29:53-- http://www.google.com/
Resolving www.google.com (www.google.com)... 172.217.168.164, 2a00:1450:4003:80a::2004
Connecting to www.google.com (www.google.com)|172.217.168.164|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'index.html'

index.html          [ <=> ] 12.87K  --.-KB/s   in 0s

2020-10-15 20:29:53 (120 MB/s) - 'index.html' saved [13179]
```

```
root@slave2-1-8:~# ping 158.109.174.1
PING 158.109.174.1 (158.109.174.1) 56(84) bytes of data.
64 bytes from 158.109.174.1: icmp_seq=1 ttl=253 time=2.50 ms
64 bytes from 158.109.174.1: icmp_seq=2 ttl=253 time=1.78 ms
64 bytes from 158.109.174.1: icmp_seq=3 ttl=253 time=1.79 ms
64 bytes from 158.109.174.1: icmp_seq=4 ttl=253 time=1.78 ms
^C
--- 158.109.174.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 1.782/1.966/2.503/0.314 ms
```