

Activitat 1: Configuració d'entorn de virtualització

Instal·lació de l'hipervisor:

L'hipervisor que he triat per instal·lar els sistemes operatius ha sigut VirtualBox. No he utilitzat VMWare, ja que em donava problemes a la instal·lació a l'hora de signar els mòduls manualment perquè tinc el Secure Boot activat en la meua màquina.

Després de revisar els docs de virtualbox (Oracle, n.d.-a), he arribat a la conclusió que conté també utilitats com les snapshots pel que em serveix perfectament com a alternativa a VMware.

Per a la instal·lació, he usat el binari oficial dels repositoris de aur:

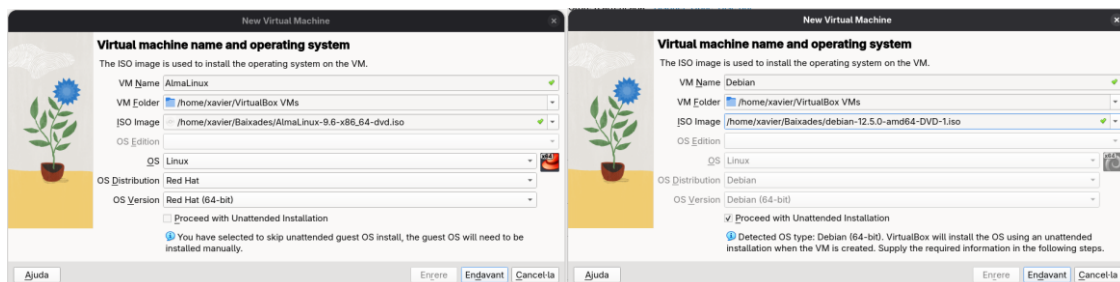
```
1 extra/virtualbox 7.2.0-2 (50.3 MiB 215.7 MiB) (Instal·lat: 7.2.0-1)
   Powerful x86 virtualization for enterprise as well as home use
```

Instal·lació de AlmaLinux i Debian:

Les isos descarregades han sigut:

1. AlmaLinux: [AlmaLinux-9.6-x86_64-dvd.iso](#) (11GB) Versió completa
2. Debian: [AlmaLinux-9.6-x86_64-dvd.iso](#) (3.7 GB)

A la creació de les màquines virtuals, en els dos casos he assignat 4 GB de ram per màquina i 2 cores, cosa que em permetrà tenir-les corrent a l'hora.



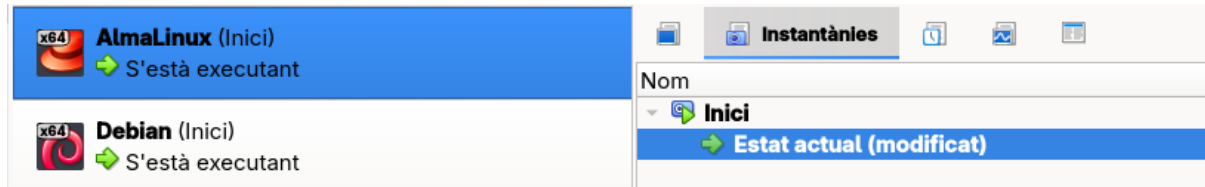
Tant pel cas de AlmaLinux com Debian s'ha instal·lat la versió Server sense gui



Un cop instal·lats dels dos sistemes, els he actualitzat i ja han quedat llestos per utilitzar.

Creació de snapshots:

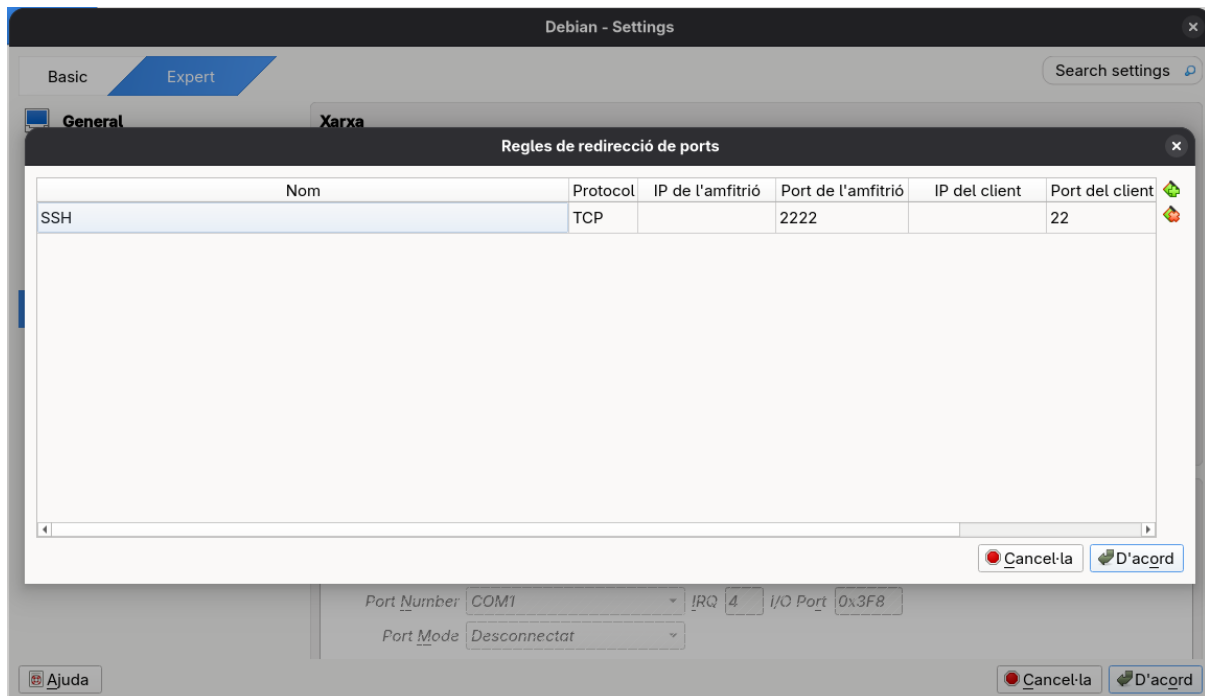
Amb les màquines correctament instal·lades he creat una snapshot de cada una per preservar l'estat inicial.



Configuració d'accés remot amb SSH:

Per tal de facilitar-me la creació de l'script d'autoengegada i connexió per SSH a la màquina, no he volgut canviar el mode de xarxa de la màquina de NAT a adaptador pont, ja que en aquest cas quan canviï de xarxa local (Com per exemple entre la de casa i la de la universitat), les IPs de les MV canviaran deixant així l'script inoperatiu.

La solució que he trobat ha sigut configurar una redirecció de ports (nsrc.org, n.d.) de manera que tot el trànsit per TCP de la màquina virtual en el port 22 (SSH), es redirigeixi al port 2222 en el cas de debian i 2223 en el cas de AlmaLinux a localhost de la meua màquina.



D'aquesta manera em puc connectar per SSH a les màquines independentment de la IP que tinguin només modificant el port:

```
~ ssh -p 2222 -o IdentitiesOnly=yes xavier@127.0.0.1 255 12:30:39
xavier@127.0.0.1's password:
Linux debian 6.1.0-39-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.148-1 (2025-08-26) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sun Sep 14 12:08:41 2025
xavier@debian:~$

~ ssh -p 2223 -o IdentitiesOnly=yes xavier@127.0.0.1 12:40:55
The authenticity of host '[127.0.0.1]:2223 ([127.0.0.1]:2223)' can't be established.
ED25519 key fingerprint is SHA256:2ZyHUr2sn7H3PGmLsupca3HSC9ZCOagKk6LzVRq/ZB8.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[127.0.0.1]:2223' (ED25519) to the list of known hosts.
xavier@127.0.0.1's password:
Activate the web console with: systemctl enable --now cockpit.socket

Last login: Sun Sep 14 12:40:34 2025
[xavier@localhost ~]$
```

Ús de la snapshot:

Per provar l'ús de l'snapshot creada, he instal·lat gcc i he mostrat per pantalla la versió instal·lada. Després he restaurat la instantània i he comprovat que gcc ja no està instal·lat:

Abans:

Després:

```
xavier@debian:~$ gcc --version
gcc (Debian 12.2.0-14+deb12u1) 12.2.0
Copyright (C) 2022 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

xavier@debian:~$ gcc --version
-bash: gcc: no s'ha trobat l'ordre
xavier@debian:~$
```

Script auto connexió SSH:

Per a l'script d'auto connexió, primer mostro un menú demanant a quina màquina em vull connectar. Les opcions són 1 per Debian i 2 per AlmaLinux. Un cop seleccionada la màquina entra en un case per a cada màquina virtual. Primer, amb l'eina VBoxManage (Oracle, n.d.-b) i el paràmetre startvm, inicio la màquina seleccionada amb l'opció de headless

Per saber quan la màquina s'ha iniciat, amb l'ajuda de netcat (blog.alcancellibre.org, n.d.), comprovo si el port 2222 en el cas de debian està disponible. Si encara no ho està, espero un segon i ho torno a comprovar.

Per últim, un cop la màquina ja està arrencada, em connecto per ssh amb la següent comanda:
ssh -p <Port màquina> -o IdentitiesOnly=yes xavier@localhost

```
1)
echo "Iniciant Debian..."
#Arrenco la VM en mode headless
VBoxManage startvm "$DEBIAN_VM" --type headless
echo "Esperant que Debian estigui disponible per SSH..."

#Espero fins que SSH estigui disponible amb netcat
until nc -z localhost $DEBIAN_PORT; do
|   sleep 1
done
echo "Connectant via SSH a Debian..."
ssh -p $DEBIAN_PORT -o IdentitiesOnly=yes xavier@localhost
;;
```

Per últim, he mogut el fitxer al PATH (/usr/local/bin) per poder-lo executar com qualsevol altra utilitat de Linux amb la comanda “startvm”

Activitat 2: Emulació en mode d'usuari amb QEMU

Després de compilar el programa hello world per a ARM i executar-lo en la meva màquina x86, aquest ha sigut el resultat:

```
xavier@debian:~/activitat2$ arm-linux-gnueabi-gcc -o helloarm -static hello.c
xavier@debian:~/activitat2$ ls
helloarm  hello.c
xavier@debian:~/activitat2$ ./helloarm
-bash: ./helloarm: no es pot executar el fitxer binari: L'executable té un format erroni
xavier@debian:~/activitat2$
```

Amb la comanda file helloarm, puc veure que el binari ha estat compilat per processadors ARM, i per això no es pot executar a la meva màquina:

```
helloarm: ELF 32-bit LSB executable, ARM, EABI5 version 1 (SYSV), statically linked, BuildID[sha1]=5879b99
9a8a2a9147024c162671c26c7aa8b8dd23, for GNU/Linux 3.2.0, not stripped
xavier@debian:~/activitat2$
```

Si ho compilo per a la meva arquitectura, es pot veure que el programa s'executa correctament

```
xavier@debian:~/activitat2$ gcc -o hellohost hello.c
xavier@debian:~/activitat2$ ls
helloarm  hello.c  hellohost
xavier@debian:~/activitat2$ ./hellohost
Hello, World!
xavier@debian:~/activitat2$

xavier@debian:~/activitat2$ file hellohost
hellohost: ELF 64-bit LSB pie executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib6
4/ld-linux-x86-64.so.2, BuildID[sha1]=29bc8e664f17526f62fe65fa2ded5ae127f382ac, for GNU/Linux 3.2.0, not
stripped
xavier@debian:~/activitat2$
```

Execució del programa amb QEMU

Un cop instal·lat el programa QEMU ja he pogut executar correctament el binari compilat per a ARM gràcies a l'emulació:

```
xavier@debian:~/activitat2$ qemu-arm helloarm
Hello, World!
xavier@debian:~/activitat2$
```

Preguntes plantejades: Reflexió

- **Investiga quines altres architectures són compatibles amb QEMU en mode d'usuari. Prova de compilar i executar un programa per a una d'aquestes architectures.**

QEMU suporta moltes architectures diferents (*QEMU Wikipedia*, n.d.). Entre aquestes hi ha:

- **x86 i x86_64** (32 i 64 bits)
- **aarch64** (ARM de 64 bits)
- **MIPS** (mips, mipsel, mips64...)
- **PowerPC** (ppc, ppc64)
- **RISC-V** (rv32, rv64)
- **s390x** (IBM Z)
- **SPARC** (sparc, sparc64)

En el meu cas he provat de compilar el Hello World per a RISC-V i executar-lo amb QEMU.

Per fer això primer he instal·lat el compilador creat per a RISC-V:

```
sudo apt install gcc-riscv64-linux-gnu
```

I he compilat el programa per a aquesta arquitectura:

```
riscv64-linux-gnu-gcc -o hellorv -static hello.c
```

Per últim, l'he executat per veure que el missatge es mostra correctament:

```
qemu-riscv64 hellorv
```

```
xavier@debian:~/activitat2$ riscv64-linux-gnu-gcc -o hellorv -static hello.c
xavier@debian:~/activitat2$ ls
helloarm  hello.c  hellohost  hellorv
xavier@debian:~/activitat2$ qemu-riscv64 hellorv
Hello, World!
xavier@debian:~/activitat2$
```

- **Per què la compilació estàtica és important en aquest context? Per què no cal incloure la bandera -static en la compilació normal? Quines avantatges i inconvenients té cada enfocament?**

Quan executem programes compilats per a altres architectures és important fer servir compilació estàtica, ja que ens permet incloure totes les biblioteques necessàries per

executar-lo dins del mateix binari. Això és important perquè el sistema al ser emulat no conté les biblioteques necessàries per executar-lo. En canvi, quan el compilem per a la nostra arquitectura, no cal que utilitzem compilació estàtica, ja que els binaris ja es troben a la màquina. Així reduïm la mida del binari significativament.

(The Differences between a Static Library and a Dynamic Library, n.d.)

Bibliografia:

- blog.alcancelibre.org. (n.d.). *Netcat*. Retrieved September 14, 2025, from <https://blog.alcancelibre.org/staticpages/index.php/como-netcat>
- <https://nsrc.org>. (n.d.). *VirtualBox lab: Port Forwarding*. Retrieved September 14, 2025, from <https://nsrc.org/workshops/2014/btnog/raw-attachment/wiki/Track2Agenda/ex-virtualbox-portforward-ssh.htm>
- Oracle. (n.d.-a). *Documentació VirtualBox*. Retrieved September 14, 2025, from <https://docs.oracle.com/en/virtualization/virtualbox/7.2/user/index.html>
- Oracle. (n.d.-b). *VBoxManage*. Retrieved September 14, 2025, from <https://www.virtualbox.org/manual/ch08.html>
- QEMU Wikipedia*. (n.d.). Retrieved September 14, 2025, from <https://en.wikipedia.org/wiki/QEMU>
- The differences between a static library and a dynamic library*. (n.d.). Retrieved September 14, 2025, from <https://medium.com/@alex24dutertre/the-differences-between-a-static-library-and-a-dynamic-library-714840bea128>