

1. (3**,5 punts**) Es vol substituir el sistema operatiu instal·lat al disc virtual (VHD) de la màquina de la mHKT 1 seguint les següents especificacions:
   * + Utilitzarem **la versió més actualitzada** de Arch Linux que podem trobar a <https://archlinux.org/>
     + Els directoris /*home/homeA, /usr/local, /var, /tmp i el swap* hauran d’estar en particions separades
     + *El sistema de fitxer es muntarà a l’engegar el sistema*
     + La maquina es dirà **<el teu\_nom>-mhkt24 (ha d'aparèixer al prompt)**
       - * **Ex.:** 
     + la xarxa es detectarà automàticament i l’assignació d'adreces serà per dhcp. Comprova abans de fer cap acció si ja tens connexió a internet.
     + Es permetrà accés amb usuari root.
     + Es crearà un usuari amb user **aso** i password **aso,** amb homea **/home/homeA** de forma manual (modificant les bases de dades d’usuari)
     + S’ha d'instal·lar grub <https://wiki.archlinux.org/title/GRUB_(Espa%C3%B1ol)>. Si l’aplicació grub no existeix, instal·la-la <https://man.archlinux.org/man/pacman.8> (utilitza l’operació Synchronize packages)
   1. (**0,5 punt**) Indica, el mes detallat possible, els passos (algoritme) que seguiràs per fer la instal·lació i que es compleixin totes les especificacions.

Descarreguem l’imatge d’archlinux

La posem dins de virtualbox, i iniciem desde el disc optic.

Iniciem archlinux

Fem l’instalacio a sdb

Modifiquem el fstab per posar cada directori a la seva particio.

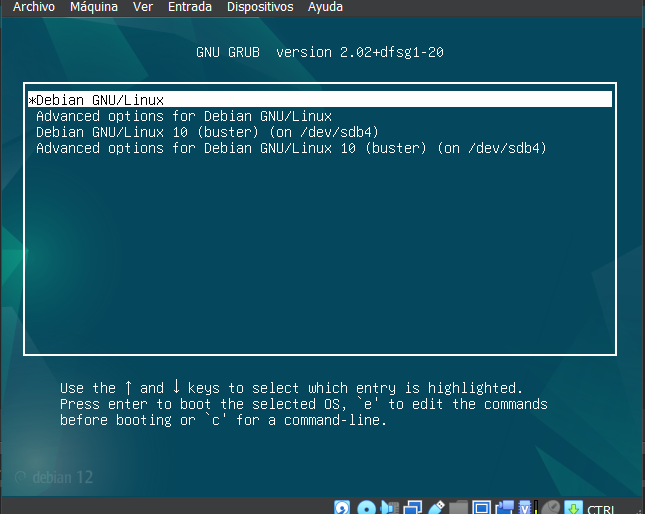
Creem usuari root.

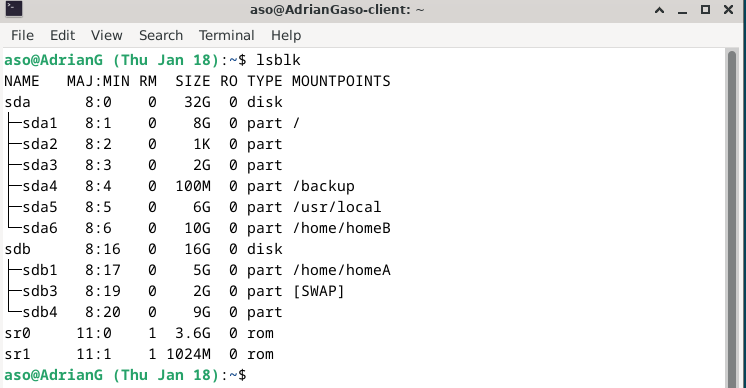
Iniciem archlinux

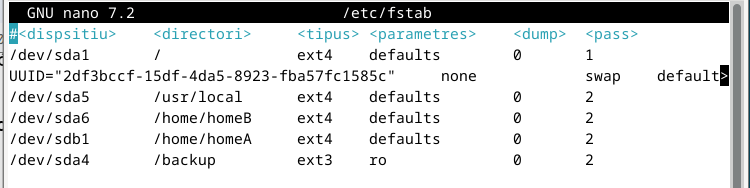
Canviem el nom de la maquina.

Conectem internet.

* 1. (**0,5 punts**) Mostra l’estat actual del teu sistema abans de fer la nova instal·lació (particions, usuaris, fstab, SO instal·lats . Indica les comandes que fas servir.







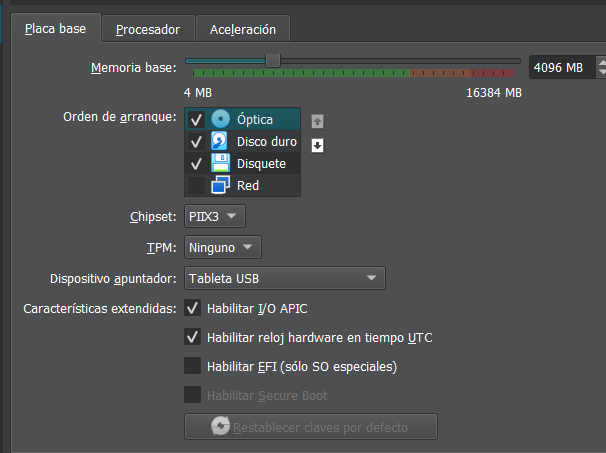
* 1. (**0,5 punt**) Descarrega la imatge de **ArchLinux** i prepara la MV per bootar des d’aquesta imatge. Indica quin es el mode d'instal·lació que has escollit. Mostra la versió a instal·la-lar.

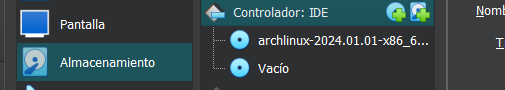
Entrem a la web d’archlinux i veiem que hi ha diferents link per d’escarregar limatge, escollim el següent.

<https://ftp.rediris.es/mirror/archlinux/iso/2024.01.01/>

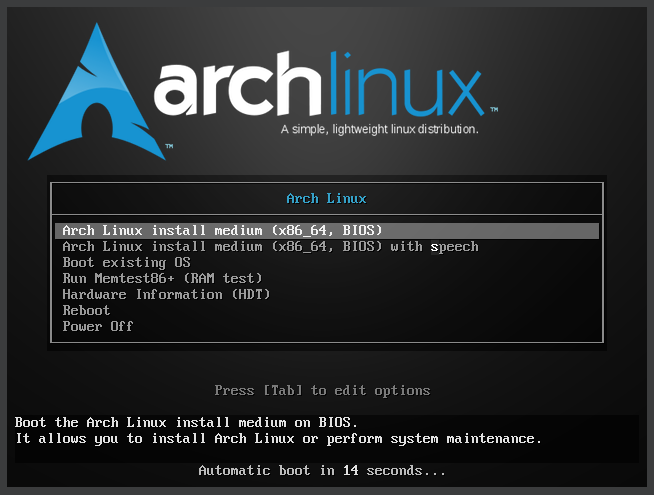
Descarreuguem la version més nova



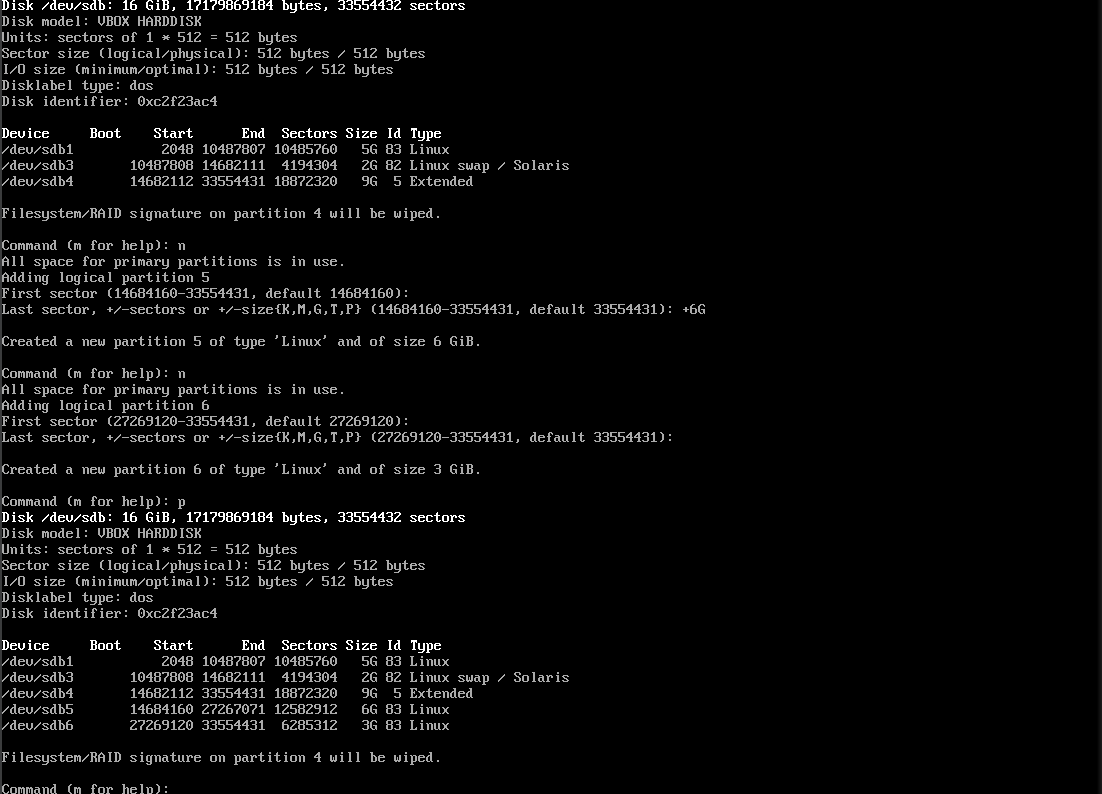




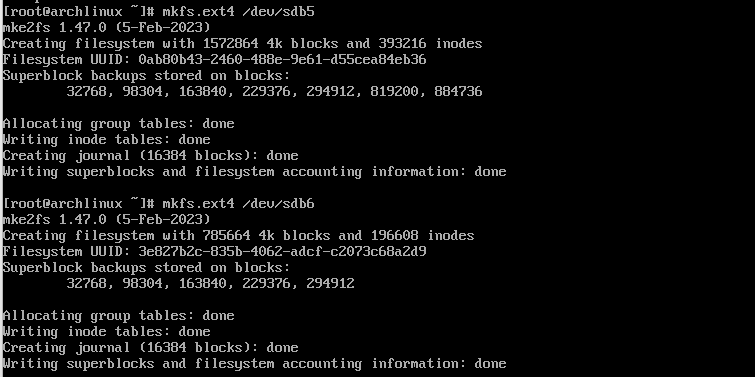
* 1. (1 **punts**) Mostra els passos que vas seguint durant la instal·lació.



Entrem al fstab /dev/sdb i creem les particions que necesitarem



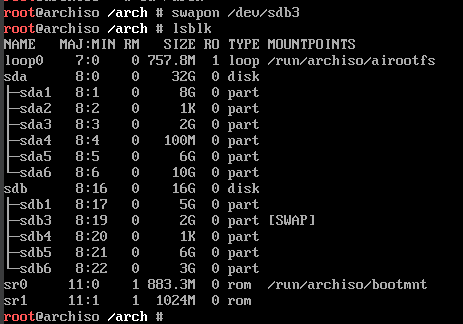
pasem les particions a ext4



creem /arch



Canviem la swap

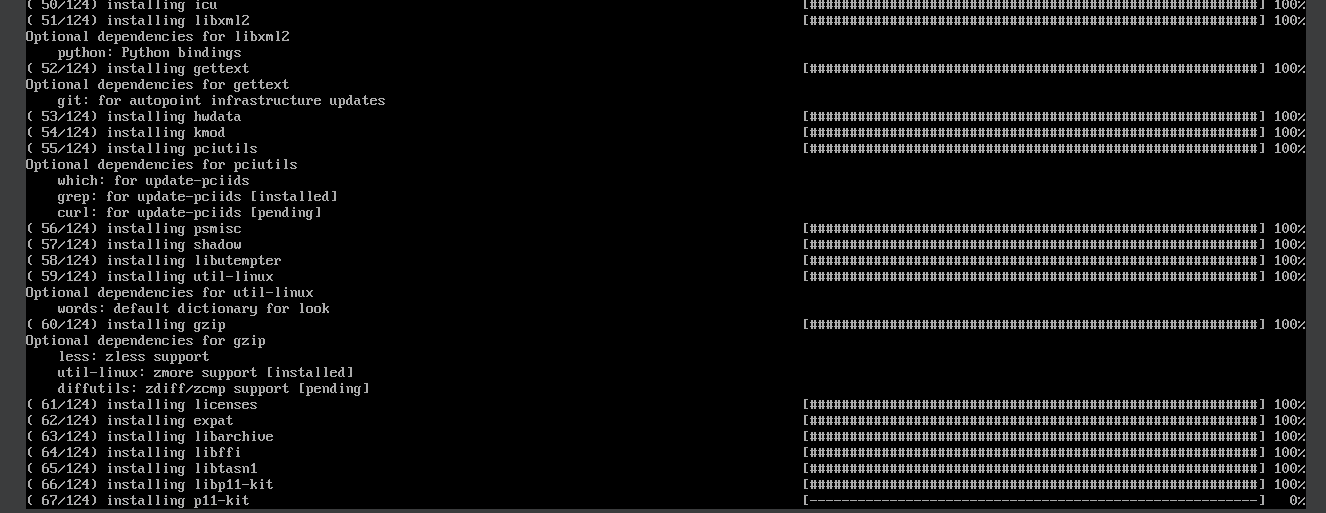


Fem que es monti a sdb1



Instalem amb la següent comanda.

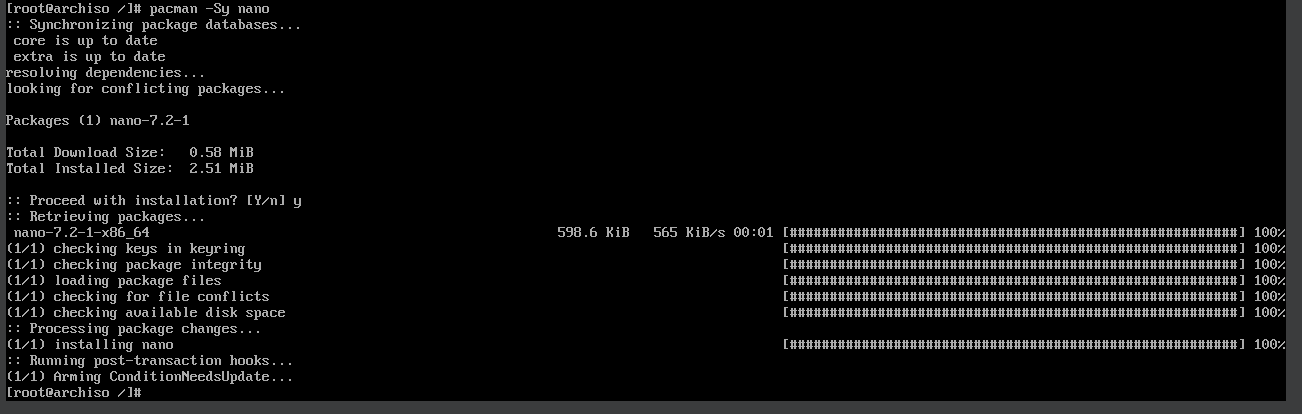
# pacstrap -K /arch base linux linux-firmware

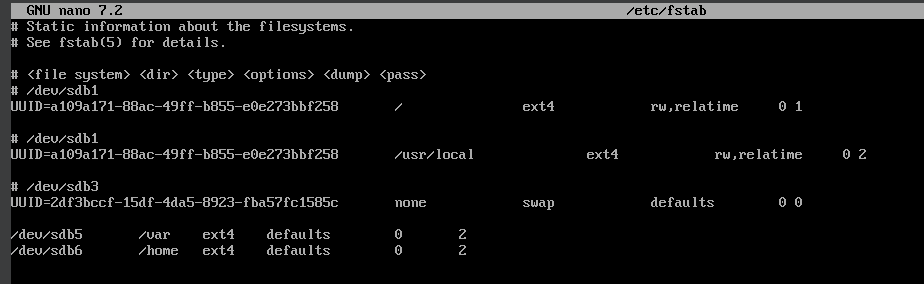


Creem el fstab i entrem al chroot

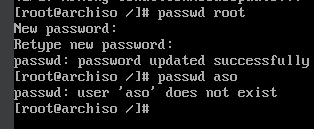


Instalem nano per modificar fstab

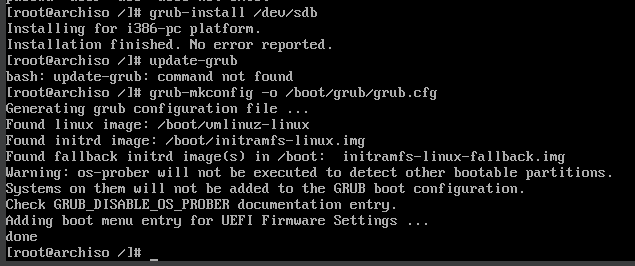




Posem contraseña al root

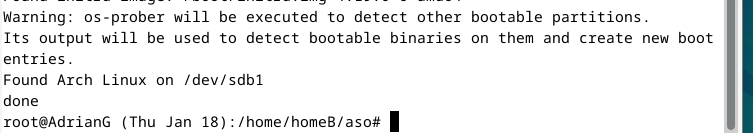


Instalem grub



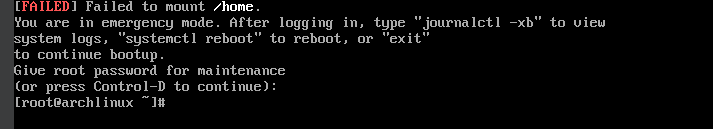
Entrem a debian

Fem update del grub

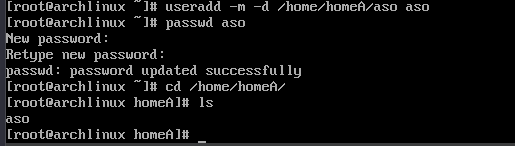


Veiem que ja podem entrar al archlinux

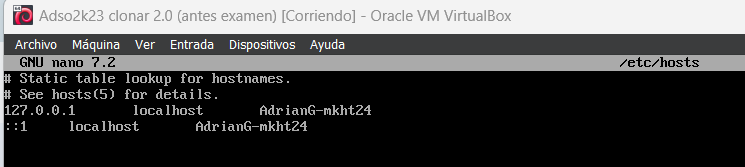




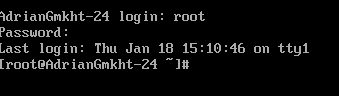
Creem aso



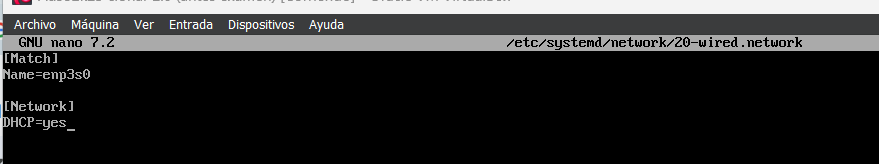
Canviem el nom a la maquina



hostnamectl set-hostname AdrianG-mkht24



configurem internet





* 1. (**1 punt**) Acaba la instal·lació. Entra al nou sistema amb l’usuari aso i mostra, des de terminal, els resultats (prompt, particions, punts de muntatge i versió del kernel)

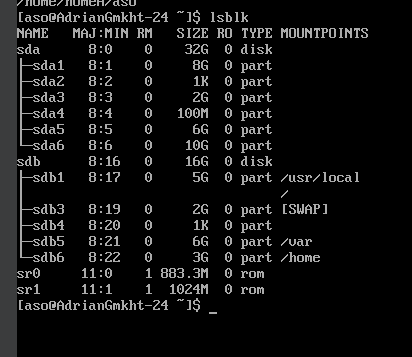


Entrem amb aso

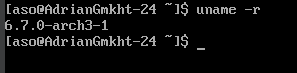




Fem lsblk per mirar on esta tot muntat



mirem la versio de kernel



1. **(3,5 punts)** Es vol crear un mecanisme per gestionar usuaris dins del sistema de tal forma que un administrador de sistemes **novell** (**sense permisos de root**) s’encarregui d’aquesta gestió. Aquest administrador s’encarregarà de modificar un arxiu de text, anomenat **usuaris.txt**, a on ficarà informació dels nous usuaris a donar d’alta o dels usuaris que s’han de donar de baixa. El nostre mecanisme utilitzarà, de forma periòdica, aquest fitxer per crear/eliminar usuaris. S’ha de complir les següents condicions:
   * + Els usuaris del sistema estan organitzats en equips de treball i, per tant, el seu grup serà el nom de l’equip. (veure adjunt: «**equips\_usuaris.xls**»)
     + L’username de cada usuari correspon al seu cognom. La repetició de cognoms ha d’estar solucionada
     + Cada equip tindrà un directori propi, penjat del **/home**, amb el nom de l’equip i a dins cada usuari tindrà el seu directori propi
     + Cap usuari pot accedir al directori de qualsevol altre usuari
     + El directori de l’equip s’esborrarà quan no hi hagi cap usuari a l’equip
     + Per tal de compartir informació, dins de cada directori d’equip hi hauran dos directoris afegits: el directori «grup» (**/home/nom\_equip/grup**) que serà accessible únicament per a tots els membres de l’equip i el directori «public» (**/home/nom\_equip/public**) que serà accessible per a tothom

L’actualització d’usuaris es farà automàticament el dilluns de cada setmana a les 6:30h, excepte els mesos de juliol i agost

* 1. (**0,5 punts**) Explica com crearàs aquest mecanisme

Per crear aquest mecanisme de gestió d'usuaris en un sistema Linux, seguirem els següents passos:

Script d'Automatització:

Crearem un script en Bash o Python que llegeixi l'arxiu usuaris.txt i realitzi les operacions necessàries.

El script haurà de gestionar la creació i eliminació d'usuaris, la gestió de grups, i la creació i eliminació de directoris corresponents.

Gestió d'Usuaris i Grups:

Per a cada usuari nou, el script crearà un usuari amb el seu cognom com a nom d'usuari, verificant que no hi hagi duplicats.

Assignarà a cada usuari al grup corresponent, basant-se en l'informació de l'arxiu equips\_usuaris.xls.

Crearà un directori per a cada usuari dins del directori de l'equip corresponent, per exemple, /home/equip\_1/cognom.

Gestió de Directoris:

Crearà un directori per a cada equip dins de /home.

Dins de cada directori d'equip, crearà dos directoris: grup i public.

Establirà els permisos adequats per a que els directoris grup siguin accessibles només pels membres de l'equip i public sigui accessible per a tothom.

Permisos:

Configurarà els permisos per a que cap usuari pugui accedir als directoris personals d'altres usuaris.

Ajustarà els permisos del directori grup i public segons les especificacions.

Neteja Automàtica:

El script comprovarà si un equip no té membres i, en aquest cas, eliminarà el directori de l'equip.

Programació del Script:

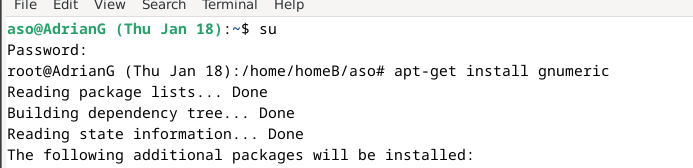
Utilitzarà cron per programar l'execució del script automàticament cada dilluns a les 6:30h, excepte durant els mesos de juliol i agost.

Permisos d'Administrador Novell:

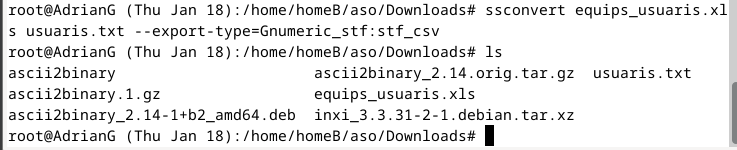
Donat que l'administrador no té permisos de root, es necessitarà configurar els permisos adequadament per a permetre-li executar aquestes tasques. Això pot implicar l'ús de sudo amb restriccions específiques o la delegació de certes tasques a un script amb permisos elevats.

* 1. (**0,5 punts**) Crea l’arxiu **usuaris.txt** a partir del fitxer de dades adjunt i fes que pugui ser modificat per l’administrador novell. Escriu les instruccions que ha de seguir l’administrador novell per a introduir/eliminar usuaris de l’arxiu **usuaris.txt**

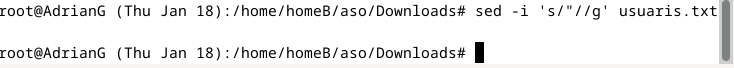
Instalem gnumeric

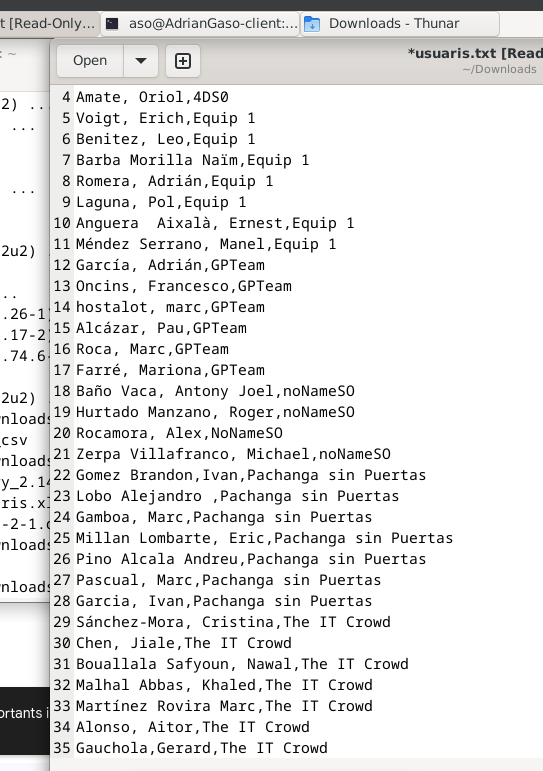


Creem el txt



Borrem les “”





* 1. (**1,5 punts**) Posa en marxa (amb una temporització puntual) el mecanisme i comprova la creació/eliminació d’usuaris, permisos d'accés a directoris i altres usuaris.

#!/bin/bash

# Comprovar si es proporciona el nom de l'arxiu com a paràmetre

if [ $# -eq 0 ]; then

echo "No s'ha proporcionat el nom de l'arxiu. Ús: ./nomscript.sh usuaris.txt"

exit 1

fi

arxiu\_usuaris=$1

# Si es passa /dev/null com a paràmetre, esborrar usuaris

if [ "$2" == "/dev/null" ]; then

while IFS=',' read -r surname name team

do

# Eliminar espais

surname=${surname// /}

name=${name// /}

team=${team// /}

# Comprovar si l'usuari existeix

if id -u "$surname" >/dev/null 2>&1; then

# Esborrar usuari

userdel "$surname"

echo "Usuari $surname esborrat."

# Esborrar directori de l'usuari

if [ -d "/home/$team/$surname" ]; then

rm -r "/home/$team/$surname"

echo "Directori de l'usuari /home/$team/$surname esborrat."

fi

# Esborrar directori de l'equip si està buit

if [ ! "$(ls -A /home/$team)" ]; then

rm -r "/home/$team"

echo "Directori de l'equip /home/$team esborrat."

fi

else

echo "L'usuari $surname no existeix. S'omet."

fi

done < "$arxiu\_usuaris"

else

declare -A membres\_equip

while IFS=',' read -r surname name team

do

# Eliminar espais

surname=${surname// /}

name=${name// /}

team=${team// /}

# Comprovar si l'usuari ja existeix

if id -u "$surname" >/dev/null 2>&1; then

echo "L'usuari $surname ja existeix. S'omet."

else

# Crear usuari sense directori personal

useradd "$surname"

echo "Usuari $surname creat."

# Establir la contrasenya de l'usuari com a surname

echo "$surname:$surname" | chpasswd

echo "Contrasenya per a l'usuari $surname establerta com $surname."

# Crear directori de l'equip si no existeix

if [ ! -d "/home/$team" ]; then

mkdir "/home/$team"

mkdir "/home/$team/grup"

mkdir "/home/$team/public"

echo "Directori /home/$team creat."

fi

# Afegir usuari al grup de l'equip

groupadd -f "$team"

usermod -a -G "$team" "$surname"

echo "Usuari $surname afegit al grup $team."

# Establir permisos per al directori de l'equip

chown :"$team" "/home/$team/grup"

chmod 770 "/home/$team/grup"

chmod 777 "/home/$team/public"

# Crear directori de l'usuari dins del directori de l'equip

mkdir "/home/$team/$surname"

chown "$surname":"$surname" "/home/$team/$surname"

chmod 700 "/home/$team/$surname"

echo "Directori de l'usuari /home/$team/$surname creat."

# Afegir usuari a l'array de membres de l'equip

membres\_equip["$team"]="$surname ${membres\_equip[$team]}"

fi

done < "$arxiu\_usuaris"

# Esborrar directoris d'equips si estan buits

for team in "${!membres\_equip[@]}"; do

if [ -z "${membres\_equip[$team]}" ]; then

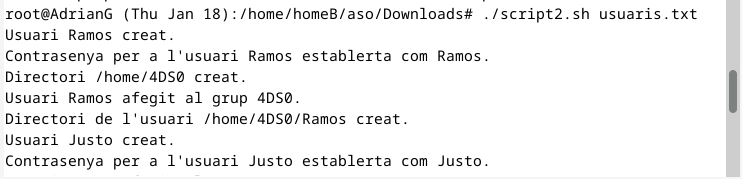
rm -r "/home/$team"

echo "Directori /home/$team esborrat."

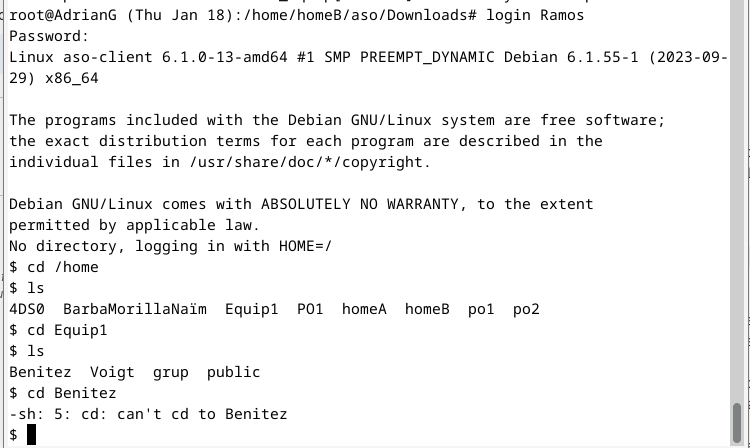
fi

done

fi



Veiem com crea el usuaris.



No podem entrar a altres usuaris.

* 1. (**1 punt**) Temporitza la tasca

entrem al crontab -e



1. **(3 punts)** La comanda **inxi** ens dona informació del nostre sistema. Utilitzant aquesta informació, volem crear un script de protecció del nostre sistema que doni un senyal d’avís quan la temperatura exterior (punt geogràfic a on estem treballant) sigui mes gran que un valor escollit per vosaltres.
   * + Aquest avís ha de contenir informació del nombre de processos en execució i el % de memòria utilitzada.
     + S’ha d’enviar un missatge al terminal de tots els usuaris connectats, tant en local com en remot via ssh. (<https://man7.org/linux/man-pages/man1/wall.1.html>)
     + **Nota: tota la informació s’ha d’extreure de la comanda inxi**
   1. (**0,5 punts**) Explica en format pseudocodi el funcionament del script

1. Obtenir la temperatura exterior utilitzant inxi.

2. Obtenir el nombre de processos en execució i el percentatge de memòria utilitzada utilitzant inxi.

3. Establir un llindar de temperatura màxim permès.

4. Si la temperatura exterior és superior al llindar:

5. Crear un missatge d'avís que inclogui:

* La temperatura exterior
* El nombre de processos en execució
* El percentatge de memòria utilitzada

6. Enviar el missatge d'avís a través de l'ordre 'wall' a tots els usuaris connectats localment.

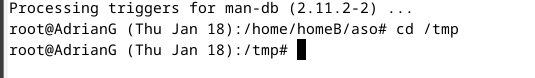
7. Obtenir una llista d'usuaris connectats a través de SSH.

8. Enviar el missatge d'avís a tots els usuaris connectats a través de SSH utilitzant SSH i l'ordre 'wall'.

9. Finalitzar l'script.

* 1. (**0,5 punts**) descarrega el fitxer precompilat (.**deb**) de inxi amb la comanda **wget** directament al directori **/tmp** (utilitza opcions de wget). Si tens instal·lat inxi, desinstal·la-ho abans de començar.

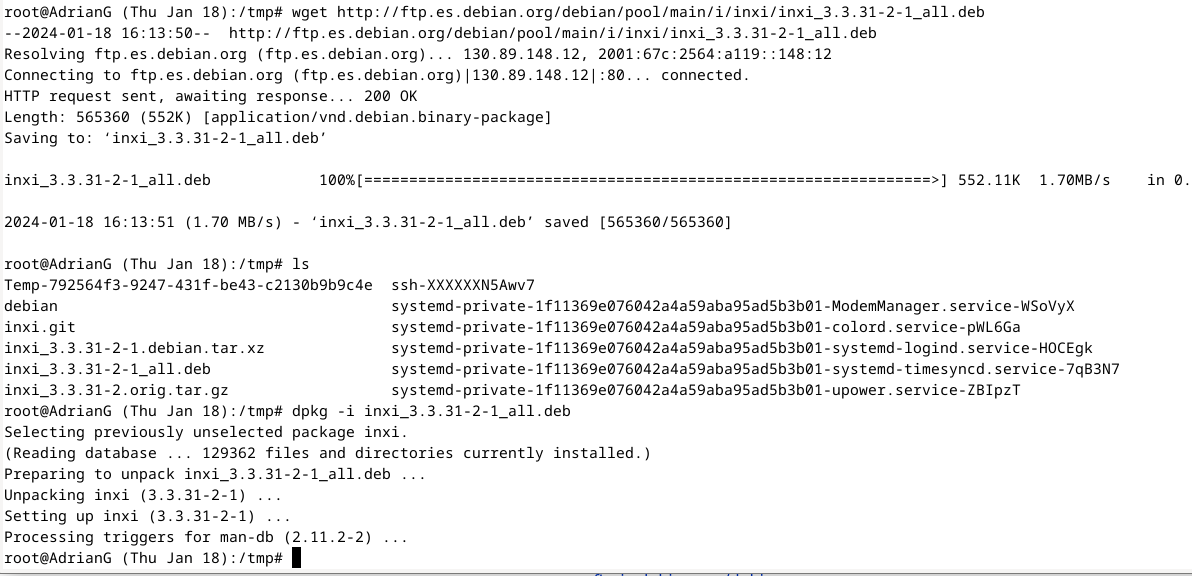
Entrem a /tmp



Fem wget del link de github que ens dona la web de debian sobre inxi.



Instalem amb dpkg -i



* 1. (**0,**5 **punts**)Temporitza la protecció segons el teu criteri d’administrador del sistema.

Obrim el crontab amb crontab -e

Farem que aquest script s’executi cada dimarts a les 12:30





* 1. (**1,5 punts**) Realitza les proves que consideris adients per tal que quedi demostrat el correcte funcionament del script (diferents valor de temperatura, enviament d’avis als usuaris connectats en local i en remot via ssh). Mostra captures de pantalla per a cada cas.

#!/bin/bash

# Extraer la temperatura exterior del resultado de inxi

temperature=$(inxi -w | grep -oP 'temperature:\s\*\K[0-9]+')

# Obtener el número de procesos en ejecución y el porcentaje de memoria utilizada utilizando inxi

process\_count=$(inxi -c 0 | grep -oP 'Procs:\s\*\K[0-9]+')

memory\_usage=$(inxi -m | grep -oP 'Mem:\s\*\K[0-9]+\.[0-9]+\/[0-9]+\.[0-9]+ MiB' | grep -oP '[0-9]+\.[0-9]+')

# Establecer el valor de temperatura límite

temperatura\_limite=15

# Verificar si la temperatura supera el límite

if [ "$temperature" -gt "$temperatura\_limite" ]; then

# Advertencia

echo "Advertencia: La temperatura exterior es superior a $temperatura\_limite°C."

# Mostrar información adicional

echo "Número de procesos en ejecución: $process\_count"

#echo "Porcentaje de memoria utilizada: $memory\_usage%"

# Enviar mensaje a todos los usuarios conectados local y remotamente

wall "Advertencia: La temperatura exterior es superior a $temperatura\_limite°C. Consulte el sistema para más detalles."

wall "Número de procesos en ejecución: $process\_count"

#wall "Porcentaje de memoria utilizada: $memory\_usage%"

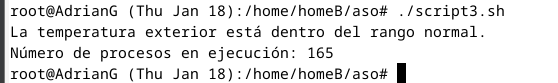
else

echo "La temperatura exterior está dentro del rango normal."

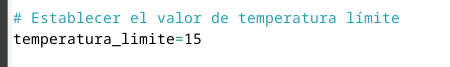
echo "Número de procesos en ejecución: $process\_count"

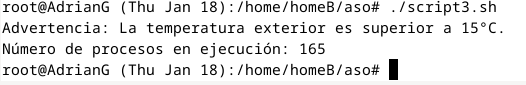
#echo "Porcentaje de memoria utilizada: $memory\_usage%"

fi

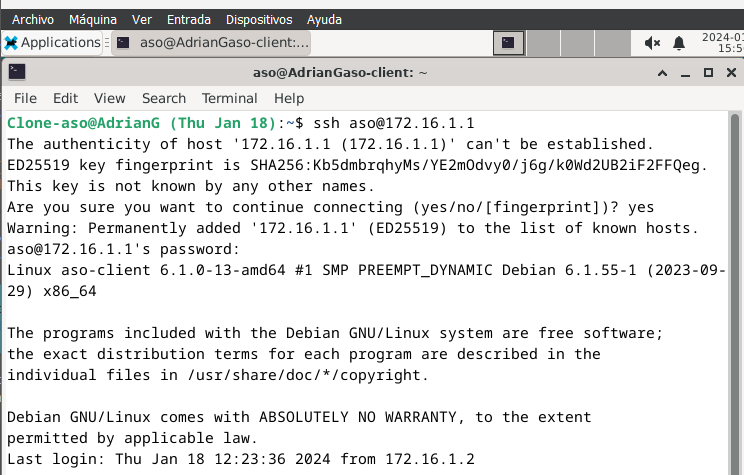
Temperatura límit a 25

Canviem temperatura Limit

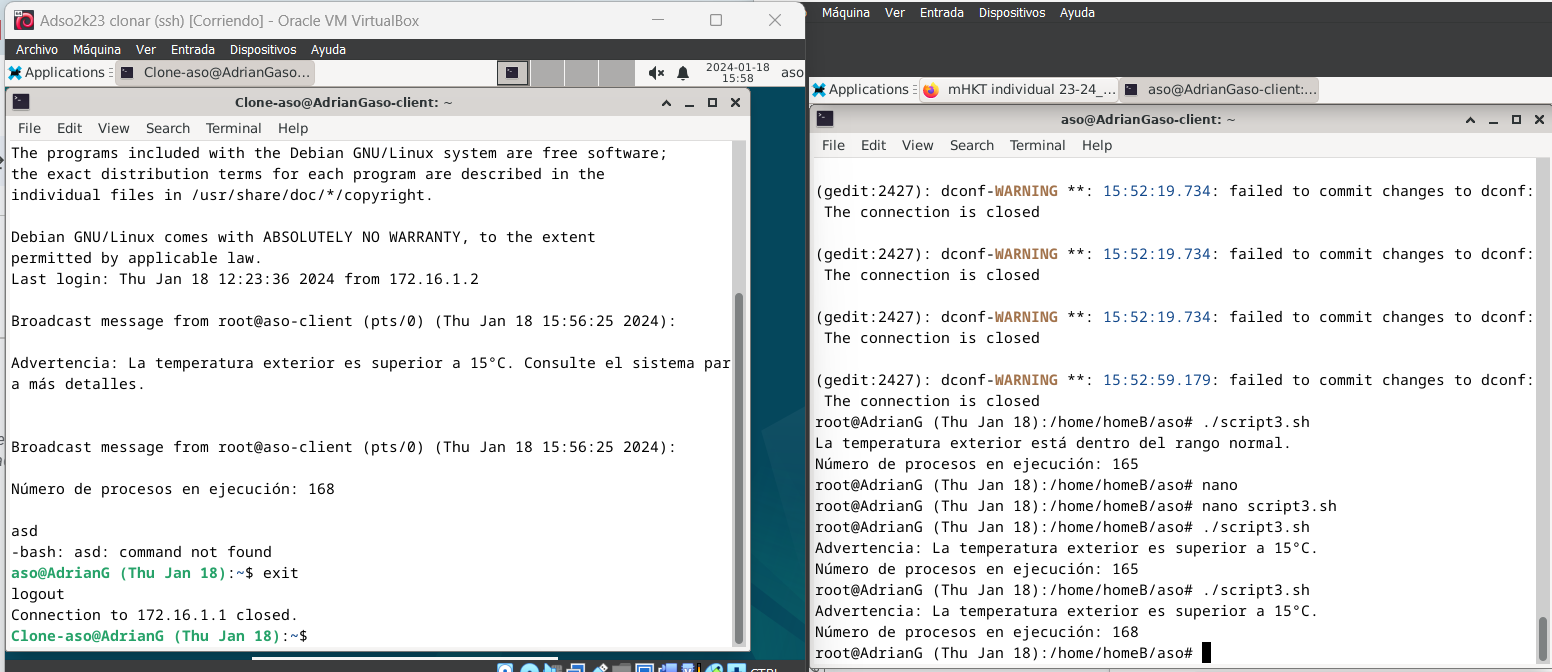




Fem ssh a la maquina remota



amb ssh també surt.



Entrega:

* **mHKT individual 23-24\_final.odt (o .doc)** amb les captures de pantalla inserides a cadascun dels apartats de l’exercici, i els comentaris que creguis convenient per tal de mostrar la resolució dels exercicis
* tots els codis **font.**