

Objectiu 2: Instal·lació d'aplicacions

1 Objectius

Instal·lar correctament software preparat específicament per a un sistema operatiu determinat i instal·lar software a partir del codi font.

2 Abans de començar

2.1 *contestar les següents preguntes abans de començar:*

- Quina comanda es fa servir per connectar-se a un servidor d'ftp?
ftp dirección_servidor
- Amb quina comanda ftp es llisten els continguts d'un directori al servidor?
ls o dir
- Amb quina comanda ftp podem agafar un fitxer del servidor?
GET fichero
- Amb quina comanda ftp podem agafar més d'un fitxer del servidor?

Con el comando mget patrón_ficheros. Por ejemplo mget *.py desgargará todos los ficheros con extensión .py.
- Com es llista el contingut d'un fitxer tar?
tar -tvf fichero.tar
- I si a més es troba comprimit amb gzip?
tar -ztvf fichero.tar.gz
- Com es descomprimeix un fitxer tar?
tar -xvf fichero.tar
- I un tar.gz? I un tar.bz2?

```
tar xzf fichero.tar.gz  
tar jxvf fichero.tar.bz2
```

- Com es feia un link a un fitxer?

```
ln ruta_fichero ruta_enlace
```

- I un softlink?

```
ln -s ruta_fichero ruta_enlace
```

- Per a què serveix la variable d'entorn PATH?

Contiene las rutas separadas por «:» en las que el interprete de comandos (en nuestro caso /bin/sh) buscará ejecutables a utilizar sin tener que especificar la ruta.

3 Introducció

El procés d'instal·lació de software en un sistema operatiu és essencialment la còpia dels fitxers que formen l'aplicació en els directoris adients i, opcionalment, la modificació d'alguns paràmetres de la configuració de l'aplicació.

Des de l'aparició de les llibreries dinàmiques el software ha adquirit dependències amb d'altre software, i com solució han aparegut diversos sistemes de gestió de software i instal·lació que faciliten el procés d'administració aquestes dependències. A més, aquests sistemes mantenen un registre de les aplicacions i llibreries instal·lades, de manera que permeten actualitzar o desinstal·lar una aplicació de forma senzilla.

Concretament, el Linux de l'assignatura està basat en una distribució Debian. Debian organitza el seu software en una sèrie de fitxers comprimits (paquets similars als .tar.gz), que guarden el binari executable, fitxers de configuració, pàgina de manual i altra documentació, i a més a més contenen informació de les dependències amb altres paquets. Els paquets Debian tenen extensió .deb.

3.1 **NOTA: Entorn gràfic de UNIX: El sistema X-Window.**

El sistema *X-window* (o *X11* o *X*) és un protocol de visualització que proveïx un conjunt d'eines estàndard per construir interfícies d'usuari gràfiques (GUI). *X* prové el marc de referència bàsic però no defineix la interfície d'usuari, aquesta es deixa als programes clients. A més a més, *X* fa ús d'un model client servidor amb el qual el servidor *X* es comunica, localment o en xarxa, amb programes clients. El servidor accepta sol·licituds de sortida gràfica (finestres) i envia de tornada al usuari les entrades (del teclat, ratolí, o altres).

El sistema *X* no conté especificacions de la interfície d'usuari de les aplicacions com poden ser: botons, menús, etc. En lloc d'això el software d'usuari

s'encarrega de l'aparença de les finestres. Per donar una aparença semblant a totes les finestres i aplicacions gràfiques existeixen programes com els gestors de finestres (*window-managers*) o els ambients d'escriptori (*desktop-environments*)

Hi ha diferents implementacions del sistema X-window para Linux (i altres sistemes UNIX). La més comú i que nosaltres utilitzarem és diu *X.org*.

A més del servidor X són necessaris altres paquets per tal de tenir un entorn gràfic complet com són els gestors de finestres, de pantalles i els ambients d'escriptori.

- **Gestor de finestres:** (*window manager*) és l'encarregat de controlar l'ubicació i aparença de les finestres de les aplicacions gràfiques. Hi ha molts gestors de finestres amb diferents funcionalitats. Podem mencionar alguns com el Metacity, Kwin, enlightenment, o Compiz.
- **Gestor de pantalla** (*display manager*): permet iniciar una sessió al servidor X des del mateix ordinador o des d'un altre ordinador mitjançant la xarxa. El gestor de pantalla presenta a l'usuari una pantalla de login i fa la validació de la contrasenya, per tant realitza funcions semblats als programes init, getty i login per a terminals de modo caràcter. Alguns gestors de pantalles comuns són: XDM (X Window Display Manager), GDM (GNOME Display Manager), i KDM (KDE Display Manager).

El gestor de pantalla és un servei que es pot arrencar i aturar com la resta de serveis del sistema mitjançant scripts d'inici que són al directori */etc/init.d*

- **Ambient d'escriptori** (*Desktop environment*) Ofereix una interfície d'usuari unificada per a les aplicacions gràfiques amb icones, barres d'eines, fondos, etc. Normalment l'ambient d'escriptori consisteix en un gestor de finestres, un gestor de pantalla i un conjunt propi d'aplicacions i llibreries. Els ambients d'escriptori més habituals són el GNOME i el KDE però hi ha molts més.

Desktop Environment	GNOME	KDE
<i>Window manager</i>	Metacity	Kwin
<i>Display manager</i>	GDM	KDM

4 Instal·lació de paquets binaris

4.1 Instal·lació manual

Volem instal·lar l'aplicació **make** al nostre sistema. Primerament necessitarem aconseguir el software a instal·lar. Els paquets que pugueu anar necessitant els teniu a ahto.epsevg.upc.es.

4.1.1 Accediu al servidor i baixeu del directori */home/public/ads* el paquet corresponent per instal·lar make.

Per instal·lar un paquet *.deb* existeix la comanda **dpkg** (Debian PacKaGe). Feu servir la següent comanda per instal·lar make:

dpkg --install <fitxer.deb>

Llegiu els missatges que surten durant el procés i assegureu-vos que no hi ha hagut cap problema.

La comanda **dpkg** a més de permetre instal·lar paquets serveix també per obtenir informació sobre els paquets i fitxers instal·lats al sistema, i desinstal·lar paquets.

4.1.2 Consulteu l'ajuda de la línia de comandes de dpkg i/o la seva pàgina de manual i completeu la següent taula:

Acció	Opció	Arguments
instal·lar un paquet	-i o --install	nombre_paquete
desinstal·lar un paquet	-r o -- remove	nombre_paquete
purgar un paquet	-p o -- purge	nombre_paquete
l·listar paquets	-- list	nombre_paquete
l·listar fitxers d'un paquet	-c o -- contents	nombre_paquete
trobar a quin paquet pertany un fitxer	-- search	patrón_de_búsqueda

4.1.3 Quina diferència hi ha entre desinstal·lar un paquet i purgar un paquet?

Si purgamos un paquete los ficheros de configuración se eliminan del sistema, al desinstalar los paquetes las ficheros de configuración se mantienen.

4.1.4 Ara volem instal·lar els programes lynx (un navegador web per mode text) i lftp (un client avançat de ftp). Baixeu els paquets corresponents del servidor i instal·leu-los amb dpkg.

```
Instalación de lftp:
root@aso-client:/home/aso/cdrom# dpkg --install lftp_4_3.deb
Selecting previously unselected package lftp.
(Reading database ... 17123 files and directories currently installed.)
Unpacking lftp (from lftp_4_3.deb) ...
Setting up lftp (4.3.1-1) ...
Processing triggers for man-db ...
/usr/bin/man2html: the setuid man user "man" does not exist
root@aso-client:/home/aso/cdrom# _
```

Instalación de lynx:

```
root@aso-client:/mnt/cdrom# dpkg --install lftp_4_3.deb
Selecting previously unselected package lftp.
(Reading database ... 39161 files and directories currently installed.)
Preparing to unpack lftp_4_3.deb ...
Unpacking lftp (4.3.1-1) ...
Setting up lftp (4.3.1-1) ...
Processing triggers for man-db (2.7.6.1-2) ...
/usr/bin/mandb: the setuid man user "man" does not exist
root@aso-client:/mnt/cdrom# dpkg --install lynx_
lynx_2_8.deb lynx_cur.deb
root@aso-client:/mnt/cdrom# dpkg --install lynx_2_8.deb
Selecting previously unselected package lynx.
(Reading database ... 39195 files and directories currently installed.)
Preparing to unpack lynx_2_8.deb ...
Unpacking lynx (2.8.8dev.9-2) ...
dpkg: dependency problems prevent configuration of lynx:
 lynx depends on lynx-cur (>= 2.8.8dev.9-2); however:
  Package lynx-cur is not installed.

dpkg: error processing package lynx (--install):
 dependency problems - leaving unconfigured
Errors were encountered while processing:
 lynx
root@aso-client:/mnt/cdrom# _
```

No se puede instalar lynx, tenemos un problema de dependencias, necesitamos instalar primer el paquete lynx-curl.

```
root@aso-client:/mnt/cdrom# dpkg --install lynx_cur.deb
Selecting previously unselected package lynx-cur.
(Reading database ... 39199 files and directories currently installed.)
Preparing to unpack lynx_cur.deb ...
Unpacking lynx-cur (2.8.8dev.9-2) ...
Setting up lynx-cur (2.8.8dev.9-2) ...
update-alternatives: using /usr/bin/lynx to provide /usr/bin/www-browser (www-br
rowser) in auto mode
Processing triggers for man-db (2.7.6.1-2) ...
/usr/bin/mandb: the setuid man user "man" does not exist
Processing triggers for menu (2.1.47+b1) ...
Processing triggers for mime-support (3.60) ...
root@aso-client:/mnt/cdrom# _
```

Ahora que tenemos la dependencia instalada volvemos a ejecutar el comando dpkg para instalar el paquete lynx.

```
root@aso-client:~# dpkg --install lynx_2_8.deb
(Reading database ... 22634 files and directories currently installed.)
Preparing to unpack lynx_2_8.deb ...
Unpacking lynx (2.8.8dev.9-2) over (2.8.8dev.9-2) ...
Setting up lynx (2.8.8dev.9-2) ...
root@aso-client:~# _
```

Para solucionar el warning de man al instalar paquetes debemos dar permisos 644 al fichero `/etc/passwd`

```
chmod 644 /etc/passwd
```

4.1.5 Executeu les comandes lynx i lftp per assegurar-vos que funcionen correctament.

Ejecución de lftp:

```
root@aso-client:/home/aso/cdrom# lftp
lftp :~> _
```

Ejecución lynx (<http://www.google.es>)

```
Google
Busqueda_Imagenes Maps Play YouTube Noticias Gmail Drive Mas >>
Historial web | Configuracion | Iniciar sesion

!Feliz Halloween!

-----
Buscar con Google Voy a tener suerte Busqueda avanzada
Herramientas del idioma

Ofrecido por Google en: catal`a galego euskara English

Programas de publicidad Soluciones Empresariales +Google
Todo acerca de Google Google.com

(c) 2017 - Privacidad - Condiciones

(NORMAL LINK) Use right-arrow or <return> to activate.
Arrow keys: Up and Down to move. Right to follow a link; Left to go back.
H)elp O)ptions P)rint G)o M)ain screen Q)uit /=search [delete]=history list
```

4.2 Instal·lació amb un gestor de paquets

Per facilitar la instal·lació d'aplicacions grans (que solen tenir moltes dependències) i també que sigui més fàcil mantenir els sistemes actualitzats varen començar a aparèixer els gestors de paquets.

Debian té un conjunt d'eines, anomenades *Apt* (Advanced front-end for dpkg), que permeten cercar, descarregar i instal·lar software i totes les seves dependències i mantenir el sistema actualitzat d'una manera senzilla i còmoda. Existeixen també diversos front-ends gràfics (**synaptic**, **adept**, ...) que no farem servir.

4.2.1 Configuració dels repositoris de software

Primerament haurem de configurar quins són els repositoris d'on APT pot obtenir els paquets *.deb* per instal·lar al sistema. Aquests repositoris poden estar a servidors remots o fins i tot en el nostre servidor (p.ex. en un cdrom) i en podem tenir tants com vulguem.

Els fitxers de configuració d'APT es troben a */etc/apt*. Dintre d'aquest directori crearem el fitxer *sources.list* amb el següent contingut:

```
deb http://ftp.es.debian.org/debian/ stable main non-free contrib
```

mostra el contingut de *source.list*.

```
GNU nano 2.2.6      File: /etc/apt/sources.list
deb http://ftp.es.debian.org/debian stable main non-free contrib
```

4.2.2 Ara, hem de fer que el sistema obtingui la llista de paquets disponibles als repositoris i la informació relacionada amb ells. Ho farem mitjançant la següent comanda:

apt-get update

```
W: Ignoring Provides line with DepCompareOp for package php-psr-log-implementation
W: Ignoring Provides line with DepCompareOp for package php-seclib
W: Ignoring Provides line with DepCompareOp for package php-sabre-http
W: Ignoring Provides line with DepCompareOp for package php-math-biginteger
W: Ignoring Provides line with DepCompareOp for package pkg-config
W: Ignoring Provides line with DepCompareOp for package pypy-cffi
W: Ignoring Provides line with DepCompareOp for package pypy-cffi-backend-api-max
W: Ignoring Provides line with DepCompareOp for package pypy-cffi-backend-api-min
W: Ignoring Provides line with DepCompareOp for package python-cffi-backend-api-max
W: Ignoring Provides line with DepCompareOp for package python-cffi-backend-api-min
W: Ignoring Provides line with DepCompareOp for package python3-cffi-backend-api-max
W: Ignoring Provides line with DepCompareOp for package python3-cffi-backend-api-min
W: Ignoring Provides line with DepCompareOp for package sagenb-export
W: Ignoring Provides line with DepCompareOp for package sagenb-export
W: Duplicate sources.list entry http://ftp.es.debian.org/debian/ stable/main i386
5 Packages (/var/lib/apt/lists/ftp.es.debian.org_debian_dists_stable_main_binary-i386_Packages)
```

Nos apareixerán warnings como el de l'última imatge, per solucionar-lo fem un

apt-get upgrade

apt-get dist-upgrade.

```
root@aso-client:~# apt-get update
Ign:1 http://ftp.es.debian.org/debian stable InRelease
Hit:2 http://ftp.es.debian.org/debian stable Release
Reading package lists... Done
root@aso-client:~# _
```

L'eina **apt-get** serveix també, entre altres coses, per instal·lar paquets, desinstal·lar-los i actualitzar tots els paquets del sistema si hi ha noves versions disponibles.

4.2.3 Amb quina comanda actualitzem els paquets instal·lats a l'última versió disponible?

Con apt-get upgrade

```
Debian4 - ASO [Corriendo] - Oracle VM VirtualBox
iputils-ping isc-dhcp-client isc-dhcp-common keyboard-configuration
libdevmapper1.02.1 libfreetype6 libgcc1 liblocale-gettext-perl libmount1
libp11-kit0 libpam-modules libpam-modules-bin libpam0g libselinux1
libsemanage-common libsemanage1 libstdc++6 libtext-charwidth-perl
libtext-iconv-perl libuuid-perl login mount passwd perl-base procps rsyslog
sysv-rc sysvinit-utils udev util-linux vim vim-common vim-runtime vim-tiny
wget
The following packages will be upgraded:
adduser base-files bash busybox console-common console-data cpio dash
debian-archive-keyring debian-keyring debianutils diffutils dmidecode
e2fslibs e2fsprogs firmware-linux-free gettext-base groff-base gzip hostname
info insserv install-info kbd klibc-utils kmod libacl1 libattr1 libblkid1
libbz2-1.0 libc-bin libc6 libc6-i686 libcomerr2 libfuse2 libgdbm3
libgpg-error0 libgpm2 libidn11 libklibc libkmod2 liblzma5 libncurses5
libncursesw5 libnewt0.52 libnftnl0 libpam-runtime libpipeline1 libpopt0
libsepoll1 libslang2 libsqlite3-0 libss2 libtext-wrapi18n-perl libtinfo5
libusb-0.1-4 libustr-1.0-1 libuuid1 linux-base logrotate lsb-base make
man-db manpages mawk multiarch-support nano ncurses-base ncurses-bin
net-tools netbase netcat-traditional os-prober psmisc readline-common sed
sensible-utils tar taskset taskset-data traceroute tzdata ucf whiptail
xkb-data xz-utils zlib1g
87 upgraded, 0 newly installed, 0 to remove and 64 not upgraded.
Need to get 55.8 MB of archives.
After this operation, 16.9 MB disk space will be freed.
Do you want to continue [Y/n]? _
```

Una vez ejecutado los comandos escribimos apt autoremove para eliminar paquetes que ya no son necesarios y que aún están en el SO.

```
The following package was automatically installed and is no longer required:
libasprintf0c2
Use 'apt autoremove' to remove it.
```

```
root@aso-client:~# apt autoremove
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be REMOVED:
  libasprintf0c2
0 upgraded, 0 newly installed, 1 to remove and 0 not upgraded.
After this operation, 63.5 kB disk space will be freed.
Do you want to continue? [Y/n] y
(Reading database ... 18233 files and directories currently installed.)
Removing libasprintf0c2:i386 (0.18.1.1-9) ...
root@aso-client:~# apt-get purge
Reading package lists... Done
Building dependency tree
Reading state information... Done
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
root@aso-client:~#
```

Después de reiniciar hemos tenido problemas con el grub, los hemos soluciado ejecutando los comandos grub-install y update-grub, al reiniciar a vuelto a

funcionar correctament.

Comproveu que tots els paquets estan actualitzats a la seva última versió.

4.2.4 Instal·lació del sistema X-Window

Ara, feu servir **apt-get** per instal·lar un servidor d'X. El paquet que heu d'instal·lar es diu *x-window-system* o *xorg*. Observeu com s'instal·len també totes les dependències i us fa les preguntes necessàries per configurar el servidor d'X.

Quin comanda heu fet servir?

```
apt-get install x-window-system
```

Al reiniciar es possible que tengamos errores al iniciar el SO,

A més d'un servidor d'X fa falta un gestor de finestres (window manager) i un ambient d'escriptori (desktop environment). Si no en coneixem cap podem buscar-ne un a la base de dades de paquets.

Una altra eina interessant d'APT és **apt-cache** que ens deixa fer consultes a la informació que el sistema ha obtingut dels repositoris en fer l'update. Fent servir **apt-cache** trobareu tots els ambients d'escriptori que tenim disponibles per instal·lar al sistema.

Quina comanda heu fet servir?

Tras decidir que queremos instalar xfce hemos escrito en la terminal «apt-cache search xfce4».

Anoteu alguns dels ambients d'escriptori que heu trobat?

Con el comando «apt-cache search 'desktop environment'»

Escritorio	A favor	En contra
Gnome	Bastante completo, incluye.	Ocupa demasiado espacio de disco y lleva aplicaciones que no nos interesan.
Unity	Bastante completo, incluye.	Ocupa demasiado espacio de disco y lleva aplicaciones que no nos interesan.
Kde	Bastante completo, incluye.	Ocupa demasiado espacio de disco y lleva aplicaciones que no nos interesan.
Cinnamon	Bastante completo, incluye.	Ocupa demasiado espacio de disco y lleva

		aplicaciones que no nos interesan.
Mate	Bastante completo, incluye.	Ocupa demasiado espacio de disco y lleva aplicaciones que no nos interesan.
Lxde	No incluye paquetes, es bastante rápido, ocupa poco espacio de disco.	Es un entorno menos completo que otras alternativas, no incluye cosas como gestor de correos, reproductores multimedia etc.
xfce	No incluye paquetes, es bastante rápido, ocupa poco espacio de disco.	Es un entorno menos completo que otras alternativas, no incluye cosas como gestor de correos, reproductores multimedia etc.

Tanto xfce y lxde son buenas opciones, xfce ocupa menos espacio de disco en comparación con lxde, pero es algo más lento por lo que nos hemos declinado por xfce ya que el tamaño del disco duro está limitado a tan solo 16GB.

Si voleu obtenir més informació d'un paquet concret (una descripció, la seves dependències, etc.) podeu fer servir:

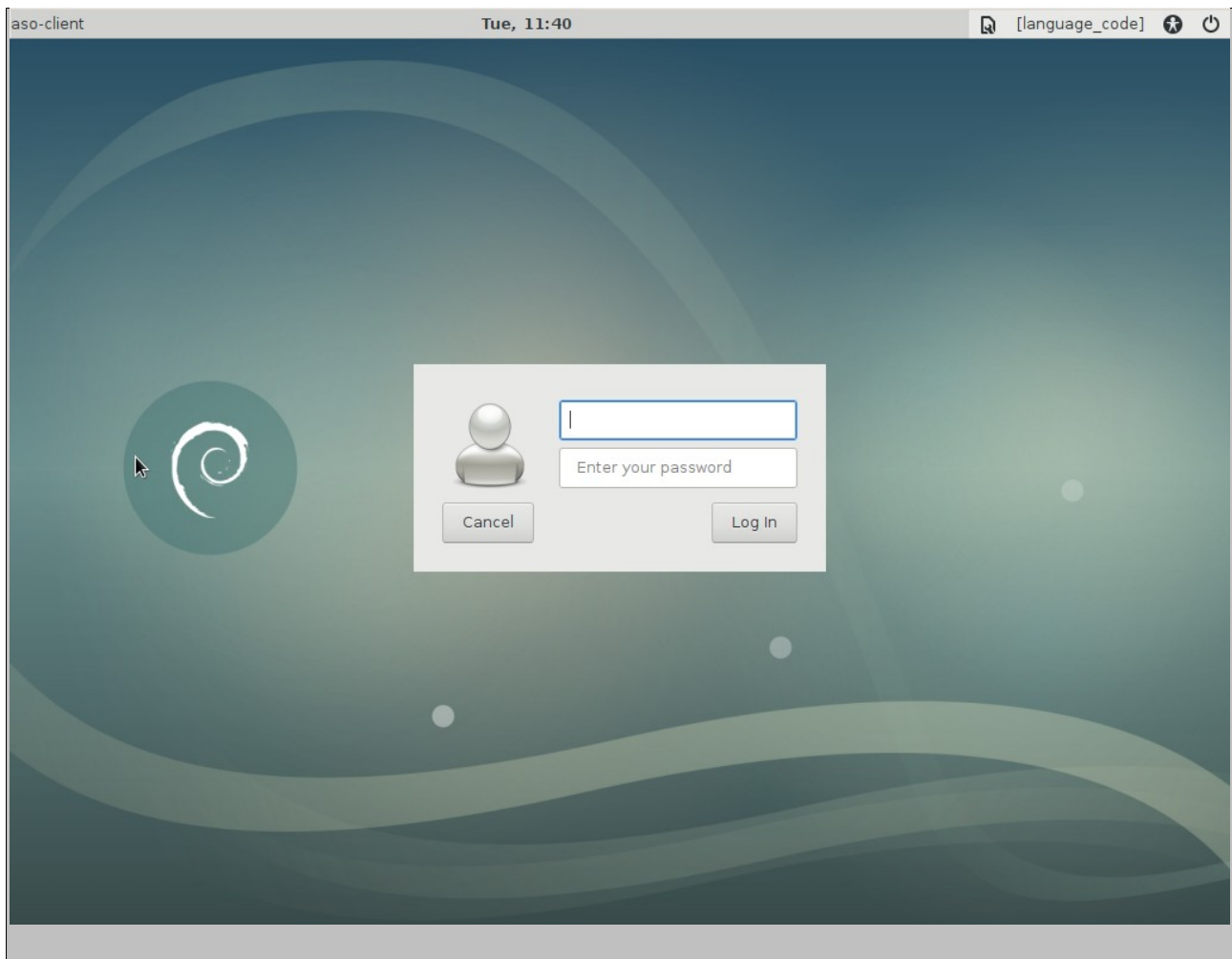
apt-cache show nom_paquet

4.2.5 Escolliu un gestor de finestres i un ambient d'escriptori i instal·leu-los fent servir apt-get.

Quina comanda heu fet servir?

Hemos instalado el entorno de escritorio xfce4 el cual ya incorpora un gestor de ventanas propio (xfwm).

```
apt-get install xfce4
```



De vegades la configuració per defecte d'un paquet no funciona i és perquè un error pot fer malbé els fitxers de configuració d'aquest paquet. En aquests casos és necessari reconfigurar el paquet i generar novament els seus fitxers de configuració. El sistema **dpkg** té una comanda per fer això:

dpkg-reconfigure nom-paquet.

Si teniu problemes amb la configuració del sistema gràfic podeu utilitzar aquesta comanda per reconfigurar el servidor X.

4.2.6 instal·leu els següents paquets: gcc (compilador), libc6-dev (llibries de desenvolupament) i iceweasel (navegador web Firefox)

Quan acabeu executeu la següent comanda:

apt-get clean

Què fa aquesta comanda?

Elimina de la cache todos los ficheros .deb que se han descargado de los repositorios al instalar paquetes con el comando apt-get

Quina diferència hi ha amb **apt-get autoclean**?

Es igual que apt-get clean pero solo borra los .deb de versiones anteriores de los programas que tenemos instalados.

4.3 Instal·lació de binaris pre-compilats

De vegades, volem instal·lar un software que no està (pel motiu que sigui) al llistat de paquets dels nostres repositoris. Volem instal·lar varies versions de la SDK de java (JDK).

4.3.1 Per fer-ho baixeu del servidor d'ftp els fitxers d'instal·lació de java que trobareu al directori public/adso de ahto. Per descomprimir cadascun dels fitxers només cal que l'executeu.

Inicialment volem instal·lar la versió 1.6 (*jdk-6-linux-i586.bin*) a */opt/java1.6*.

Quines comandes heu fet servir per descomprimir el fitxer?

Hemos descargado todas las aplicaciones del servidor ahto, python2 y python3 de las paginas oficiales y hemos generado una .iso con el genisoimage para posteriormente montarla en la unidad optica de la máquina virtual.

Se ha montado el cd-rom:

```
modprobe isofs
```

```
mkdir /home/aso/cdrom
```

```
mount -o ro /dev/cdrom /home/aso/cdrom
```

Hemos hecho una copia de .bin que contiene el jdk6

```
cp /home/aso/cdrom/ /opt/jdk_6_li.bin
```

Ejecutamos el .bin

```
cd /opt
```

```
./jdk_6_li.bin
```

```
inflating: jdk1.6.0/man/ja_JP.eucJP/man1/jrunscript.1
inflating: jdk1.6.0/man/ja_JP.eucJP/man1/jsadebugd.1
inflating: jdk1.6.0/man/ja_JP.eucJP/man1/jstack.1
inflating: jdk1.6.0/man/ja_JP.eucJP/man1/jstat.1
inflating: jdk1.6.0/man/ja_JP.eucJP/man1/jstatd.1
inflating: jdk1.6.0/man/ja_JP.eucJP/man1/native2ascii.1
inflating: jdk1.6.0/man/ja_JP.eucJP/man1/rmic.1
inflating: jdk1.6.0/man/ja_JP.eucJP/man1/schemagen.1
inflating: jdk1.6.0/man/ja_JP.eucJP/man1/serialver.1
inflating: jdk1.6.0/man/ja_JP.eucJP/man1/wsgen.1
inflating: jdk1.6.0/man/ja_JP.eucJP/man1/wsimport.1
inflating: jdk1.6.0/man/ja_JP.eucJP/man1/xjc.1
Creating jdk1.6.0/jre/lib/rt.jar
Creating jdk1.6.0/jre/lib/jsse.jar
Creating jdk1.6.0/jre/lib/charsets.jar
Creating jdk1.6.0/lib/tools.jar
Creating jdk1.6.0/jre/lib/ext/localedata.jar
Creating jdk1.6.0/jre/lib/plugin.jar
Creating jdk1.6.0/jre/lib/javaws.jar
Creating jdk1.6.0/jre/lib/deploy.jar
Done.
```

A quin directori us ho ha descomprimit?

En /opt/jdk1.6.0

```
root@aso-client:/opt# ls
jdk1.6.0  jdk_6_li.bin
root@aso-client:/opt# _
```

Nos piden instal·larlo en /opt/java1.6 así que una vez hecha la instalaci3n renombramos el directorio del jdk y eliminamos el fichero jdk_6_li.bin ya que no lo necesitaremos.

```
root@aso-client:/opt# ls
jdk1.6.0  jdk_6_li.bin
root@aso-client:/opt# mv jdk1.6.0 java1.6
root@aso-client:/opt# rm jdk_6_li.bin
root@aso-client:/opt# _
```

4.3.2 Mireu el contingut i localitzeu on es troba l'executable java que executa la m·quina virtual. Ara moveu tot el directori (no només el bin) en què s'ha descomprimit al directori de destí (/opt/java1.6).

Verifiquem que està correctament instal·lada:

/opt/java1.6/bin/java -version

4.3.3 Ara repetiu aquest passos per les JDK 1.4 i 1.5 en instal·leu-les a /opt/java1.4 i /opt/java1.5 respectivament.

Si intentem esbrinar quina és la versió per defecte fent:

java -version

Quin error surt?

```
root@aso-client:/opt# java -version
-bash: java: command not found
```

A què es degut?

Falta crear un softlink de /opt/java.16/bin/java en el directorio /usr/bin

4.3.4 Per solucionar aquest problema el més senzill es fer un *softlink* des d'un dels directoris que es troben al PATH cap al binari que volem que sigui accessible. Feu un *softlink* de /usr/bin/java a /opt/java1.6/bin/java.

Amb quina comanda ho aconseguiriu?

`ln -s /opt/java1.6/bin/java /usr/bin/java`

```
root@aso-client:/opt/java1.6/bin# ln -s /opt/java1.6/bin/java /usr/bin/java
root@aso-client:/opt/java1.6/bin# java -version
java version "1.6.0"
Java(TM) SE Runtime Environment (build 1.6.0-b105)
Java HotSpot(TM) Client VM (build 1.6.0-b105, mixed mode, sharing)
root@aso-client:/opt/java1.6/bin# _
```

4.3.5 A més volem que cada versió sigui accessible directament amb *javaversió* (p.ex. *java1.6*).

Quines comandes feu servir per aconseguir-ho?

```
root@aso-client:/opt# ln -s /opt/java1.6/bin/java /usr/bin/java1.6
root@aso-client:/opt# ln -s /opt/java1.5/bin/java /usr/bin/java1.5
root@aso-client:/opt# java1.6 -version
java version "1.6.0"
Java(TM) SE Runtime Environment (build 1.6.0-b105)
Java HotSpot(TM) Client VM (build 1.6.0-b105, mixed mode, sharing)
root@aso-client:/opt# java1.5 -version
java version "1.5.0"
Java(TM) 2 Runtime Environment, Standard Edition (build 1.5.0-b64)
Java HotSpot(TM) Client VM (build 1.5.0-b64, mixed mode, sharing)
root@aso-client:/opt# _
```

5 Instal·lació a partir del codi font

De vegades haurem d'instal·lar una aplicació directament des del codi font, bé perquè no existeix el paquet als nostres repositoris o bé perquè volem adaptar-lo d'alguna manera al nostre sistema.

5.1 Volem instal·lar un petit shell restringit que farem servir en altres pràctiques. Baixeu el fitxer [asosh-0.1.tar.gz](http://ahto.epsevg.upc.es/~home/public/adso/asosh-0.1.tar.gz) del directori [ahto.epsevg.upc.es : /home/public/adso](http://ahto.epsevg.upc.es/~home/public/adso).

5.2 Un lloc habitual per posar el codi font és a </usr/src>. Descomprimiu el codi font amb la comanda `tar` en aquest directori.

Quina comanda heu fet servir?

```
root@aso-client:/usr/src# tar xzf asosh_0_.tgz
root@aso-client:/usr/src# ls
asosh-0.1  asosh_0_.tgz
root@aso-client:/usr/src# _
```

5.3 Executeu adequadament l'script `configure` perquè s'instal·li a </usr/local>.

Mireu el contingut del directori amb el codi font. Habitualment trobareu un script anomenat **configure** que permet configurar parts del procés de compilació i instal·lació (activar/desactivar parts del codi, decidir el directori d'instal·lació, etc.). La informació específica sobre aquest script sol estar als fitxers *README* i *INSTALL*.

Per defecte s'instal·larà a </usr/local>.

Quins paràmetres heu fet servir?

`./configure`

Fixeu-vos que dóna un error en detectar si les llibreries es troben instal·lades.

Quin error dóna?

```
configure: error: *** readline headers not found. ***
root@aso-client:/usr/src/asosh-0.1# _
```

A què es degut?

Falta la libreria readline.

Com l'heu solucionat? (pista: recordeu que els headers solen estar en un paquet a part)
Instalando el paquete libreadline-dev con la herramienta apt

```
root@aso-client:/usr/src/asosh-0.1# apt-get install libreadline-dev
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  file libmagic-mgc libmagic1 lynx-common mime-support
Use 'apt-get autoremove' to remove them.
The following extra packages will be installed:
  libreadline7 libtinfo-dev ncurses-bin
Suggested packages:
  readline-doc
The following NEW packages will be installed:
  libreadline-dev libreadline7 libtinfo-dev
The following packages will be upgraded:
  ncurses-bin
1 upgraded, 3 newly installed, 0 to remove and 126 not upgraded.
Need to get 751 kB of archives.
After this operation, 1366 kB of additional disk space will be used.
Do you want to continue [Y/n]? Y_
```

```
checking readline/readline.h usability... yes
checking readline/readline.h presence... yes
checking for readline/readline.h... yes
checking readline/history.h usability... yes
checking readline/history.h presence... yes
checking for readline/history.h... yes
configure: creating ./config.status
config.status: creating Makefile
config.status: creating src/Makefile
config.status: creating config.h
config.status: config.h is unchanged
config.status: executing depfiles commands
root@aso-client:/usr/src/asosh-0.1# _
```

5.4 Un cop fet el configure passem a compilar el codi font (comproveu que no surten errors en compilar):

make

En general per fer aquests dos primers passos no fan falta permisos d'administrador així que és recomanable fer-ho amb un usuari que no sigui root. L'últim pas, en canvi, que consisteix en col·locar els binaris i altres fitxers (de configuració, de dades, llibreries, etc.) al lloc que volem del sistema de fitxers, habitualment necessita permisos de root.

make install

Verifiqueu que s'ha instal·lat tot correctament tot executant la comanda **asosh**.


```

root@aso-client:/usr/src/asosh-0.1# make
make all-recursive
make[1]: Entering directory `/usr/src/asosh-0.1'
Making all in src
make[2]: Entering directory `/usr/src/asosh-0.1/src'
if gcc -DHAVE_CONFIG_H -I. -I. -I.. -g -O2 -MT asosh.o -MD -MP -MF ".deps/asosh.Tpo" -c -o asosh.o asosh.c; \
    then mv -f ".deps/asosh.Tpo" ".deps/asosh.Po"; else rm -f ".deps/asosh.Tpo"; exit 1; fi
/bin/bash ../libtool --tag=CC --mode=link gcc -g -O2 -o asosh -lreadline asosh.o
mkdir .libs
gcc -g -O2 -o asosh asosh.o -lreadline
make[2]: Leaving directory `/usr/src/asosh-0.1/src'
make[2]: Entering directory `/usr/src/asosh-0.1'
make[2]: Leaving directory `/usr/src/asosh-0.1'
make[1]: Leaving directory `/usr/src/asosh-0.1'
root@aso-client:/usr/src/asosh-0.1# _

```

Make install

```

asosh.o
mkdir .libs
gcc -g -O2 -o asosh asosh.o -lreadline
make[2]: Leaving directory `/usr/src/asosh-0.1/src'
make[2]: Entering directory `/usr/src/asosh-0.1'
make[2]: Leaving directory `/usr/src/asosh-0.1'
make[1]: Leaving directory `/usr/src/asosh-0.1'
root@aso-client:/usr/src/asosh-0.1# make install
Making install in src
make[1]: Entering directory `/usr/src/asosh-0.1/src'
make[2]: Entering directory `/usr/src/asosh-0.1/src'
test -z "/usr/local/bin" || mkdir -p -- "/usr/local/bin"
/bin/bash ../libtool --mode=install /usr/bin/install -c 'asosh' '/usr/local/bin/asosh'
/usr/bin/install -c asosh /usr/local/bin/asosh
make[2]: Nothing to be done for `install-data-am'.
make[2]: Leaving directory `/usr/src/asosh-0.1/src'
make[1]: Leaving directory `/usr/src/asosh-0.1/src'
make[1]: Entering directory `/usr/src/asosh-0.1'
make[2]: Entering directory `/usr/src/asosh-0.1'
make[2]: Nothing to be done for `install-exec-am'.
make[2]: Nothing to be done for `install-data-am'.
make[2]: Leaving directory `/usr/src/asosh-0.1'
make[1]: Leaving directory `/usr/src/asosh-0.1'
root@aso-client:/usr/src/asosh-0.1# _

```

Ejecución asosh

```

root@aso-client:/usr/src/asosh-0.1# asosh
Type 'help' to view available commands:
> _

```

Durant el procés de compilació s'hauran generat diversos fitxers de caràcter temporal (p.ex. els fitxers objecte). Així que un cop finalitzada la instal·lació és una bona opció esborrar aquests fitxers. El Makefile ja ve preparat perquè això es pugui fer fàcilment.

Quina comanda heu fet servir per esborrar els fitxers temporals?

Make clean

```
root@aso-client:/usr/src/asosh-0.1# make clean
Making clean in src
make[1]: Entering directory `/usr/src/asosh-0.1/src'
rm -f asosh asosh
rm -rf .libs _libs
rm -f *.o
rm -f *.lo
make[1]: Leaving directory `/usr/src/asosh-0.1/src'
Making clean in .
make[1]: Entering directory `/usr/src/asosh-0.1'
rm -rf .libs _libs
rm -f *.lo
make[1]: Leaving directory `/usr/src/asosh-0.1'
root@aso-client:/usr/src/asosh-0.1# ls src/
Makefile Makefile.am Makefile.in asosh.c
root@aso-client:/usr/src/asosh-0.1# _
```

A més a més, habitualment el Makefile també vindrà preparat per poder desfer totes les passes que va fer en el procés d'instal·lació.

Amb quin argument s'ha d'invocar al Makefile perquè ho faci?

Make uninstall