

**PLA DE TREBALL DEL TREBALL FINAL DE CARRERA:**

Desenvolupament des de zero d'un sistema Linux  
per a ser instal·lat als ordinadors de les aules  
ordinàries d'un institut de secundària

Óscar Arranz Lafuente

Segon quadrimestre del curs 2013-2014

Enginyeria tècnica en informàtica de sistemes

# ÍNDEX

---

1 Motivacions.....	3
2 Objectius.....	4
3 Requeriments.....	5
4 Pla de treball.....	6
5 Tasques preliminars a la construcció del LDZ.....	8
5.1 Creació de la partició i obtenció dels paquets necessaris per construir el LDZ.....	8
5.2 Establiment de l'entorn de treball per construir el LDZ.....	9
5.3 Instal·lació del sistema Linux mínim.....	11
6 Construcció del sistema LDZ.....	14
6.1 Preliminars a la instal·lació dels paquets.....	14
6.2 Instal·lació dels paquets.....	16
7 Establiment dels scripts d'arrencada del sistema.....	19
8 Instal·lació del nucli i arrencada del sistema.....	21
9 Implementació de l'escriptori gràfic.....	28
10 Conclusions.....	37
ANNEX I: Fitxers de configuració i paquets de programari del sistema Linux mínim.....	41
ANNEX II: Paquets de programari instal·lats del sistema LDZ base.....	43
ANNEX III: Fitxers de configuració del sistema LDZ base.....	45
ANNEX IV: Paquets de programari emprats en la implementació de l'escriptori gràfic.....	49

## **1 Motivacions**

Dues han estat les motivacions que m'han dut a decidir-me per triar com a projecte de fi de carrera el desenvolupament d'un sistema Linux des de zero per tal de ser implementat en els ordinadors disponibles per part del professorat als instituts de secundària:

La primera d'elles ha sigut el meu interès personal per la manera com els sistemes estan dissenyats internament. En descobrir als estudis sobre sistemes operatius a la UOC la història del desenvolupament de Linux a partir de Minix, així com l'existència del projecte Hurd de Gnu per elaborar enlloc d'un nucli monolític un microkernel, he tingut sempre l'interès per aprofundir en aquest aspecte. A més en descobrir també que en l'àmbit de Linux i el projecte Gnu es podien bastir sistemes operatius personalitzats des de zero, no vaig tenir cap mena de dubte que això darrer era el que volia fer com a projecte final de carrera.

La segona motivació que m'ha endut a la tria d'aquest projecte ha vingut donada per la meva experiència laboral com a professor de secundària. Actualment l'ús d'ordinadors en l'àmbit de l'ensenyament està més que consolidat. És ben normal que a cada aula d'un centre docent es disposi d'un ordinador amb el qual el professor pugui ajudar-se per fer la seva activitat docent. El problema és que aquests ordinadors, que funcionen amb alguna versió de Windows, acostumen a ser força antics i a estar sobrecarregats de programes. Això comporta una gran lentitud en carregar el sistema operatiu i en l'execució dels programes, fet que comporta una considerable pèrdua del temps disponible per les tasques docents. A més com la situació econòmica actual fa molt poc probable una renovació d'aquest material es fa necessari aprofitar els ordinadors disponibles amb les seves limitacions. I és en aquest aspecte en el que considero que un

sistema Linux dissenyat exclusivament a oferir només els serveis que la tasca docent exigeix pot ser molt més eficient que l'ús no diguem ja de Windows, sinó de una distribució convencional de Linux pel que fa l'aprofitament dels recursos limitats d'aquest ordinadors.

## **2 Objectius**

### **Generals**

- Aprendre l'estructura d'un sistema operatiu basat en Linux.
- Crear-ne un sistema basat en Linux des de zero.
- Instal·lar en un ordinador un sistema operatiu basat en Linux des de zero.

### **Concrets**

- Minimitzar els requeriments de memòria i CPU del sistema Linux des de zero.
- Implementar només el programari necessari per oferir les funcionalitats requerides en la tasca docent d'un institut de secundària: accés a Internet, programari ofimàtic i programari audiovisual.



### **3 Requeriments**

#### **Per al desenvolupament del projecte**

- Llibre Linux From the Scratch versió 7.5 de Gerard Beekmans .
- Ordinador amb arquitectura AMD/Intel x86 de 32 ó 64 bits.
- Sistema operatiu Linux instal·lat.

#### **Per a la instal·lació del sistema Linux creat des de zero**

- Ordinador amb arquitectura AMD/Intel x86 de 32 ó 64 bits.

#### **Requeriments per part de l'usuari (professor de secundària)**

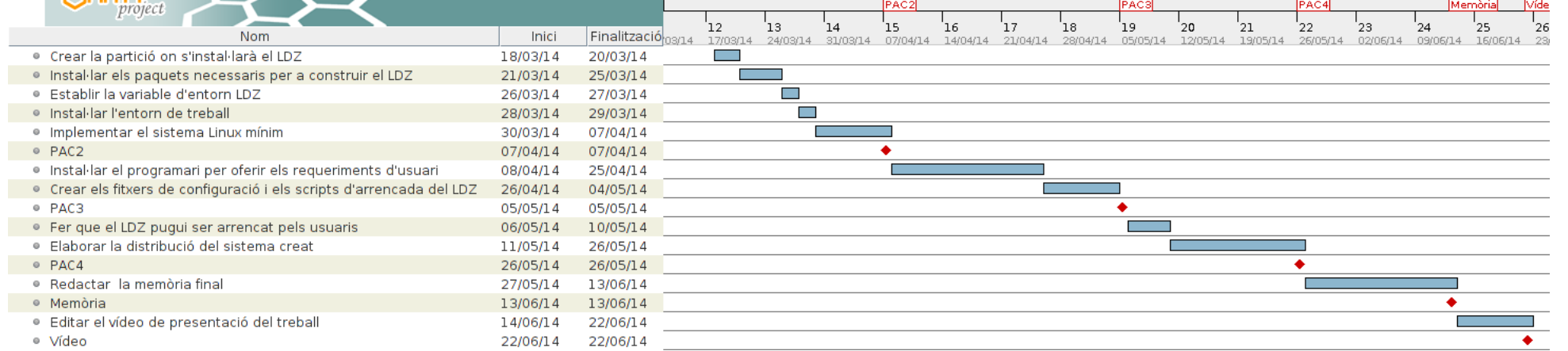
- Disposar dels navegadors Mozilla i Chrome per accedir als continguts d'Internet.
- Accedir a plataformes educatives, ja sigui basades en la tecnologia Moodle o específiques d'alguna empresa proveïdora d'aquest servei.
- Disposar de la suite ofimàtica LibreOffice.
- Disposar del reproductor multimèdia VLC.
- Disposar del programari Smart Notebook per a poder fer activitats amb la pissarra electrònica.
- Accedir als recursos compartits de l'institut: servidors de disc, impressores o la intranet del centre.
- Accedir al sistema operatiu a través del sistema d'usuari i contrasenya.
- Disposar d'un petit espai en el disc de l'ordinador local per poder emmagatzemar fitxers de mida reduïda.

## 4 Pla de treball

L'elaboració del sistema *Linux des de zero* (d'ara endavant LDZ) que es pretén dur a terme com a treball de final de carrera cal dividir-ho en un seguit de tasques successives que permetin l'assoliment dels objectius plantejats en l'apartat dos d'aquest document. Així seguint tant les indicacions del llibre *Linux From the Scratch*, com del pla docent d'aquesta matèria, he establert la següent llista de tasques, que són les que conformen el pla de treball a seguir per part meva. També, per acomplir amb el calendari de lliurament d'activitats de la matèria, he agrupat aquestes tasques en les PACs indicades en el pla docent:

<ul style="list-style-type: none"> <li>• Tasca 1: Crear la partició on s'instal·larà el LDZ.</li> <li>• Tasca 2: Instal·lar els paquets necessaris per a construir el LDZ.</li> <li>• Tasca 3: Establir la variable d'entorn LDZ.</li> <li>• Tasca 4: Instal·lar l'entorn de treball.</li> <li>• Tasca 5: Implementar un sistema Linux mínim per encetar la implementació del LDZ.</li> </ul>	<b>PAC 2</b>
<ul style="list-style-type: none"> <li>• Tasca 6: Instal·lar el programari per oferir els requeriments d'usuari detallats en l'apartat 3.</li> </ul>	<b>PAC 3</b>
<ul style="list-style-type: none"> <li>• Tasca 7: Crear els fitxers de configuració i els scripts d'arrencada del LDZ.</li> <li>• Tasca 8: Fer que el LDZ pugui ser arrencat pels usuaris.</li> </ul>	<b>PAC 4</b>
<ul style="list-style-type: none"> <li>• Tasca 9: Redactar la memòria final</li> <li>• Tasca 10: Editar el vídeo de presentació</li> </ul>	

La temporització de totes aquestes tasques es recull en el següent diagrama de Gantt:



## 5 Tasques preliminars a la construcció del LDZ

### 5.1 Creació de la partició i obtenció dels paquets necessaris per construir el LDZ

El primer requisit que cal satisfer per tal de construir el LDZ es reservar prou espai en el disc, o en un dels discos, de l'ordinador on es fa aquesta tasca per tal d'instal·lar-hi aquest sistema. És a dir cal crear una partició específica per al LDZ. El mínim espai exigible pel sistema descrit a *Linux From Scratch* és de 2,8 GB, tanmateix es recomana un mínim de 10 GB per poder fer ampliable el sistema. Cal tenir en compte que partim de la base que el sistema que es vol construir haurà de ser destinat a ordinadors antics, amb uns quants anys, al disc dels quals s'hauran de crear dues particions, una per instal·lar una distribució Linux convencional i una altra per al LDZ. Donat que una distribució com Ubuntu per exemple només exigeix vora 5 GB per poder ser instal·lat, és necessari comptar amb un disc de 15GB d'espai per tal de poder fer la instal·lació. Espai prou reduït com per que un ordinador, encara que antic, pugui disposar d'ell. En el meu cas, donat que dispo de prou espai en el disc del meu ordinador he destinat 40 GB per ubicar-hi el LDZ en una partició, feta amb el programa Gparted, amb un sistema de fitxers de tipus ext4.

Un cop creada la partició amb el sistema de fitxers, aquesta ha de ser muntada des d'alguna distribució Linux. En el meu cas Ubuntu 12.04 LTS que és la que faig servir habitualment. Llavors simplement amb la comanda `mount` es pot accedir des de l'Ubuntu a la partició on s'anirà construint el LDZ. Primer de tot cal crear un directori `sources` on emmagatzemar tots els paquets i patches de programari necessaris per elaborar el LDZ. Aquest procés es força trivial, ja que des de la pàgina web [www.linuxfromscratch.org](http://www.linuxfromscratch.org) s'ofereix una llista d'aquests paquets per automatitzar la seva baixada amb l'ordre `wget`.

A la següent captura es pot observar tant el muntatge de la partició com els directoris creats en ella:

```
ansgarius@Lenovo: ~  
ansgarius@Lenovo:~$ ls /media/  
lfs  
ansgarius@Lenovo:~$ ls /media/lfs/  
ansgarius@Lenovo:~$ cat scripInici.sh  
#!/bin/bash  
export LFS=/media/lfs  
mount -t ext4 /dev/sda9 $LFS  
ansgarius@Lenovo:~$ sudo ./scripInici.sh  
ansgarius@Lenovo:~$ ls /media/lfs/  
sources tools  
ansgarius@Lenovo:~$
```

## 5.2 Establiment de l'entorn de treball per construir el LDZ

Un cop creada la partició i obtinguts els paquets necessaris, cal establir un entorn de treball adient per la construcció del LDZ. La raó subjacent a la creació d'aquest entorn rau en el fet que la construcció del LDZ s'ha de dur a terme de la manera més aïllada possible de l'entorn propi de la màquina amfitrió. Bona part de les eines emprades en la compilació dels paquets: el compilador, assemblador, o enllaçador, empen variables d'entorn, com la variable `PATH`, per dur a terme la seva funció. Llavors, per tal d'evitar problemes degut a les múltiples configuracions diferents de l'entorn que cada usuari pugui tenir a la seva distribució de Linux, el llibre *Linux From Scratch*, defineix un entorn específic i simple per construir un sistema des de zero. Primer de tot es crea un enllaç a l'arrel / del sistema amfitrió, anomenat `tools`, cap al directori `/tools` de la partició on

es vol ubicar el LDZ, per tal que sempre es busqui la utilitat compilada en aquesta, i no la que pugui oferir també el sistema amfitrió. A continuació cal crear l'usuari `lfs`, que serà amb el qual es durà a terme la construcció del LDZ. El pas més rellevant en l'establiment d'aquest entorn és la definició d'un fitxer de configuració del shell que aquest usuari faci servir després del login en el sistema. Es vol evitar que aquest shell faci servir la configuració per defecte del sistema, especificada a `/etc/profile`, que s'obté un cop fet el login. S'aprofita per això el fet que un cop s'hagi obtingut la configuració es consulta el fitxer de configuració a nivell d'usuari particular `.bash_profile`, fent que aquest obri un terminal de tipus `bash` només amb les variables d'entorn `HOME`, `TERM` i `PS1` definides específicament. En ésser aquest shell de tipus `non-login`, llegeix per defecte el fitxer de configuració `.bashrc` també a nivell de l'usuari específic. És en aquest fitxer en el que es defineixen les altres variables d'entorn que aquest shell farà servir: `LFS`, on s'indica el punt de muntatge de la partició on es construirà el sistema LDZ; `LC_ALL`; `LFS_TGT`, amb el nou nom assignat al 'target triplet', que en el meu cas és `x86_64-lfs-linux-gnu`; i finalment `PATH`, amb l'assignació `/tools/bin:/bin:/usr/bin`, que estableix que sempre es buscarà qualsevol utilitat primer de tot a `/tools`, que com s'ha comentat anteriorment, és un enllaç cap al directori `tools` que es troba a la partició on es construirà el LDZ. El contingut d'aquests dos fitxers de configuració es poden consultar a l'Annex I.

Així doncs quan des d'un terminal amb l'usuari propi, en el meu cas el que faig servir a l'Ubuntu, es faci el login de l'usuari `lfs` fent `su - lfs`, es carregarà aquest entorn específic automàticament, tal com es pot observar a la següent captura:

```
ansgarius@Lenovo: ~  
ansgarius@Lenovo:~$ su - lfs  
Contrasenya:  
lfs@Lenovo:~$ echo $HOME  
/home/lfs  
lfs@Lenovo:~$ echo $TERM  
xterm  
lfs@Lenovo:~$ echo $PS1  
${debian_chroot:+($debian_chroot)}\u@\h:\w\$  
lfs@Lenovo:~$ echo $SHELL  
/bin/bash  
lfs@Lenovo:~$ echo $LFS  
/media/lfs  
lfs@Lenovo:~$ echo $LC_ALL  
POSIX  
lfs@Lenovo:~$ echo $LFS_TGT  
x86_64-lfs-linux-gnu  
lfs@Lenovo:~$ echo $PATH  
/tools/bin:/bin:/usr/bin  
lfs@Lenovo:~$
```

### 5.3 Instal·lació del sistema Linux mínim

Malgrat que sembli que un cop habilitat l'entorn de treball ja s'està en disposició d'iniciar la construcció del LDZ, això no és encara possible. El motiu, tal com s'ha comentat abans, és la necessitat que les eines per compilar els paquets amb els quals cal construir el sistema, a més d'altres com l'enllaçador dinàmic, tinguin una configuració adient per la construcció del LDZ, que no queda garantida amb la configuració per defecte d'aquestes eines en el sistema amfitrió. Cal per tant primer de tot compilar des de zero les eines que es faran servir per construir el LDZ, passant llavors els paràmetres adients en fer el `./configure` per tal d'evitar que facin servir qualsevol eina o variable del sistema amfitrió que pugui donar problemes. En aquest apartat es tracta doncs de compilar aquestes eines, construint llavors un sistema Linux mínim, paral·lel a l'amfitrió, amb un entorn totalment controlat, per tal de poder crear el LDZ sense problemes inesperats.

Per aquest motiu, la tasca d'aquest apartat consisteix en seguir fil per randa totes les indiccions del capítol 5, *Constructing a Temporary System*, del llibre *Linux From Scratch*. En aquest punt no es tracta encara de fer cap personalització, doncs cal assegurar que l'entorn sigui el més fiable possible.

L'ordre i la manera com es van compilant i instal·lant els diferents paquets es rellevant per tal que aquest sistema mínim permeti la construcció del LDZ sense problemes. La clau està en el paquet `binutils` que conté l'assemblador i l'enllaçador, el qual ha de ser compilat i instal·lat el primer de tot, per tal que a continuació, en compilar i instal·lar el paquet `gcc`, aquest faci servir l'assemblador i l'enllaçador del paquet `binutils` que s'ha compilat i instal·lat anteriorment. Per aquest motiu aquesta compilació i instal·lació dels paquets `binutils` i `gcc` es fa en dos passos, compilant i instal·lant entre mig les capçaleres que conformem l'API de Linux, així com evidentment, les llibreries de C i C++ amb `glibc` i `Libstdc++` respectivament, que empraran aquesta API. La resta de paquets, fins a vint-i-nou, es van compilant i instal·lant per tal que facin servir les eines que prèviament s'hagin compilant i instal·lant. Un cop enllestit el procés de compilació i instal·lació tal com es descriu en aquest capítol de *Linux From Scratch* s'obté un seguit d'eines de desenvolupament entrelligades, *toolchain*, que permetran compilar i instal·lar els paquets que conformaran el LDZ definitiu. I a més a més de manera deslligada de les que ofereix el sistema amfitrió, problemàtiques potencialment, que és la raó per la qual es crea aquest sistema intermedi.

El resultat de tot aquest procés és pot observar a les següents captures:



```
lfs@lenovo:/media/lfs/tools/bin$ ls
[
addr2line      colrm      gawk-4.1.0  ld         mknod      pwd         strings     unlzma      x86_64-lfs-linux-gnu-size
ar             column     gcc         ld-new     mktemp     ranlib      strip       unshare     x86_64-lfs-linux-gnu-strings
as            comm       gcc-ar      ldd        mount      readelf     stty        unix        x86_64-lfs-linux-gnu-strip
awk           cp         gcc-nm      link       mountpoint readlink    sync        uptime      x86_64-unknown-linux-gnu-c++
base64        cpp        gcc-ranlib  linux32    msgfmt     rename      tac         users       x86_64-unknown-linux-gnu-gcc
basename     csplit    gcov        linux64    msgmerge   renice      tac         utmpdump   x86_64-unknown-linux-gnu-gcc-4.8.2
bash         cut        genatc      ln         mtrace     reset       rev         tail        uidgen      x86_64-unknown-linux-gnu-gcc-ar
bashbug      cytune    getconf     locale     mv         rmdir       tar         tailf      vdir        x86_64-unknown-linux-gnu-gcc-nm
bunzip2      date      getent      localedef  namei      rm          taskset    wall        x86_64-unknown-linux-gnu-gcc-ranlib
bzip2        dd         getopt      locate     ncursesw5-config
bzcat        df         gprof       logger     nice        runcon      rmdir      taskset    wc          xargs
bzcmp        ddiff     grep        logname    nl          runtest     tee        who         xz          xzcat
bzegrep      dff3      groups      look       nm          nohup       script     test        whoami     xzcmp
bzfgrep      dir       gunzip      ls         nproc      sediff      text2any   x86_64    xzdec     xzdiff
bzgrep       dircolors gzeke      lsblk      nroff      sediff      text2dvi  x86_64-lfs-linux-gnu-addr2line
bzip2       dirmname  gzip       lscpu      nsenter    sed         text2pdf  x86_64-lfs-linux-gnu-ar
bzp2recover dmesg     head       lsof       numfmt     seq         texindex  x86_64-lfs-linux-gnu-as
bzip2       du         hexdump    lzcat      objcopy    setarch     tic        timeout    x86_64-lfs-linux-gnu-c++filt
bzless      echo      hostid     lzfgrep    oldfind    setsid      toe        touch      x86_64-lfs-linux-gnu-g++
bzmore     egrep     hostname   lzgrep     patch      setterm     tput      true       x86_64-lfs-linux-gnu-gcc
c++filt     eject     i386       lzfgrep    paste      sha1sum     tr         truncate  x86_64-lfs-linux-gnu-gcc-ar
cal          elfedit   iconv      lzgrep     patch      sha224sum   tr         true       x86_64-lfs-linux-gnu-gcc-4.8.2
captoinfo   env       id         lzless     pathchk    sha256sum   tr         true       x86_64-lfs-linux-gnu-gcc-nm
cat          expand     igawk      lzmainfo   p          sha384sum   perl       tset       x86_64-lfs-linux-gnu-gcc-ranlib
catchsegv   expect    infocmp    lzmadec    pdftex12dvi
cc           expr      infotocap  lzmainfo   perl       sha512sum   shred      tsort     x86_64-lfs-linux-gnu-gcov
chcon       factor    install-info lzmore     pg         shuf        size       tzselect  x86_64-lfs-linux-gnu-gprof
checkmk     fallocate install-info lzmore     pg         shuf        size       tzselect  x86_64-lfs-linux-gnu-gprof
chgrp       false     ionice     m4         pldd       sleep       ul         umount    x86_64-lfs-linux-gnu-ld
chmod       fgrep     ipcman     make       pod2text   sort        uncompress x86_64-lfs-linux-gnu-ld.bfd
chown       file      ipcperm   makedb     pod2text   sort        uncompress x86_64-lfs-linux-gnu-ld.bfd
chroot      find      ipcs      maketnfo   pod2text   sort        uncompress x86_64-lfs-linux-gnu-ld.bfd
chrt        findmnt  isosize   mcookie    pr          printenv   sprof     unexpand  x86_64-lfs-linux-gnu-objdump
cksum       flock    join      mds5sum    printenv   sprof     unexpand  x86_64-lfs-linux-gnu-objdump
clear       fmt       kill      msg        prlimit    stat        uniq       x86_64-lfs-linux-gnu-ranlib
cmp         fold     last      mkdir      ptx        stdbuf     unlnk     x86_64-lfs-linux-gnu-readelf
col         g++      lastb     mkfifo     ptx        stdbuf     unlnk     x86_64-lfs-linux-gnu-readelf
lfs@lenovo:/media/lfs/tools/bin$
```

A la captura anterior es poden observar tots els programes instal·lats a /media/lfs/tools/bin i a la següent les llibreries a /media/lfs/tools/bin:

```
lfs@lenovo:/media/lfs/tools/lib$ ls
Mcr1.o      libatomic.so      libdl.so.2      libmenuv.a      libnss_db-2.19.so  libquadmath.la  libthread_db-1.0.so
Scr11.o     libatomic.so.1    libformw.a      libmenuv.so     libnss_db.so       libquadmath.so  libthread_db.so
audit       libatomic.so.1.0.0 libformw.so     libmenuv.so.5   libnss_db.so.2     libquadmath.so.0  libthread_db.so.1
crt1.o     libbfd.a          libformw.so.5   libmenuv.so.5.9 libnss_dns-2.19.so libquadmath.so.0.0.0 libtsan.a
crti.o     libbfd.la        libformw.so.5.9 libmount.a      libnss_dns.so      libresolv-2.19.so libtsan.la
crtn.o     libblkid.a        libg.a          libmount.la     libnss_dns.so.2    libresolv.a      libtsan.so
expect5.45 libblkid.la       libgcc_s.so     libmount.so     libnss_files-2.19.so libresolv.so.0   libtsan.so.0
gawk       libblkid.so       libgcc_s.so.1   libmount.so.1   libnss_files.so    libresolv.so.2   libtsan.so.0.0.0
gcc        libblkid.so.1     libiberty.a     libmount.so.1.1.0 libnss_files.so.2  librpcsvc.a      libutil-2.19.so
gconv     libblkid.so.1.1.0 libieee.a       libmudflap.a    libnss_hesiod-2.19.so librt-2.19.so    libutil.a
gcrt1.o   libbz2.a          libitm.a        libmudflap.la   libnss_hesiod.so   librt.a          libutil.so
lcl4.0.0  libc-2.19.so     libitm.la       libmudflap.so   libnss_hesiod.so.2 librt.so         libutil.so.1
ld-2.19.so libc.a           libitm.so       libmudflap.so.0 libnss_hesiod.so.2 librt.so.1      libuuid.a
ld-linux-x86-64.so.2 libc.so         libitm.so.1     libmudflap.so.0.0.0 libnss_nts-2.19.so librt.so.1      libuuid.la
libBrokenLocale-2.19.so libc.so.6      libitm.so.1.0.0 libmudflapth.a   libnss_nts.so.2   libssp.la       libuuid.so
libBrokenLocale.a libc_nonshared.a libitm.spec     libmudflapth.la libnss_nisplus-2.19.so libssp.so       libuuid.so.1
libBrokenLocale.so libcheck.a     liblzma.a       libmudflapth.so libnss_nisplus.so.2 libssp.so.0     libuuid.so.1.3.0
libBrokenLocale.so.1 libcheck.la    liblzma.la      libmudflapth.so.0 libnss_nisplus.so.2 libssp.so.0.0.0 perl5
libSegFault.so libcheck.so    liblzma.so      libmudflapth.so.0.0.0 libopcodes.a      libssp_nonshared.a pkgconf
libanl-2.19.so libcheck.so.0 liblzma.so.5    libncursesw.a   libopcodes.la     libssp_nonshared.la sqlite3.8.0
libanl.a libcheck.so.0.0.0 liblzma.so.5.0.5 libncursesw.so  libpanelw.a       libstdc++.a     tc18
libanl.so.1 libcidn-2.19.so libm-2.19.so    libncursesw.so.5 libpanelw.so      libstdc++.a-gdb.py libstdc++.la    tc18.6
libasan.a libcidn.so.1    libm.a          libncursesw.so.5.9 libpanelw.so.5.9 libstdc++.so     tcLocConfig.sh
libasan.la libcrypt-2.19.so libm.so.6       libncursesw.so.5.9 libpcprofile.so  libstdc++.so      tdb1.0.0
libasan.so libcrypt.a     libmagic.la     libnsl-2.19.so  libpthread-2.19.so libstdc++.so.6  tdbcmysql1.0.0 tdbcodbc1.0.0
libasan.so.0 libcrypt.so     libmagic.so     libnsl.a        libpthread.so      libstdc++.so.6.0.18-gdb.py tdbcpstgres1.0.0
libasan.so.0.0 libcrypt.so.1  libmagic.so.1   libnsl.so.1     libpthread.so.0    libsupc++.a     terminfo
libasan_preinit.o libdl-2.19.so  libmagic.so.1.0.0 libnss_compat-2.19.so libpthread.so.0  libsupc++.la    thread2.7.0
libatomic.a libdl.a        libmagic.a      libnss_compat.so libpthread_nonshared.a libtcl8.6.so    tinfo
libatomic.la libdl.so       libmagic.a      libnss_compat.so.2 libquadmath.a     libtclstub8.6.a
lfs@lenovo:/media/lfs/tools/lib$
```

La llista completa i ordre d'instal·lació dels paquets que conformen aquest sistema Linux mínim es poden consultar a l'Annex I.

## 6 Construcció del sistema LDZ

La finalitat de l'apartat anterior consistia en la creació d'un sistema compost d'un seguit d'eines que fessin possible la creació d'un sistema hoste (guest) de la manera més independent que fos possible de l'amfitrió (host). Conseqüentment, en aquest apartat hi correspon la implementació de l'esmentat sistema hoste. Tanmateix abans de la instal·lació dels paquets adients per a la seva constitució cal dur a terme tot un seguit d'accions preliminars que la facin possible. Són les que es detallen al següent subapartat.

### 6.1 Preliminars a la instal·lació dels paquets

El sistema Linux que s'especifica al llibre *Linux from scratch*, del qual es vol derivar el LDZ en aquest treball, respecta l'estàndard FHS (*Filesystem Hierarchy Standar, Estàndard de jerarquia del sistema de fitxers*) propi de bona part de les distribucions basades en Linux. Per tant el primer que cal fer és crear-ne a l'arrel de la partició on es vol bastir el sistema LDZ els directoris que conformen aquest estàndard. A la següent imatge es poden observar els directoris creats just per sota del directori arrel:

```
root:/# ls /
bin  dev  home  lib64  mnt  proc  run   sources  sys  tools  var
boot etc  lib   media  opt  root  sbin  srv      tmp  usr
root:/#
```

Tanmateix la instal·lació del sistema hoste al directori arrel de la partició on es vol construir aquest planteja un seriós problema pel que fa a la necessària independència entre aquest i el sistema amfitrió: els fitxers de dispositiu i els sistemes de fitxers virtuals del nucli. Per un costat tenim que els paquets de programari que constituirà el sistema final hauran de referenciar un seguit de fitxers de dispositiu, els que es troben al directori `/dev`. Tanmateix, en aquest punt del desenvolupament del sistema aquest

directori encara no n'inclou cap. La solució que es dóna a *Linux from Scratch* per sortir-se'n d'aquest problema és doble. Primer de tot crear-ne dos fitxers de dispositiu, console i null, amb l'eina mknod especificant els nombres minor i major de la següent manera:

```
mknod -m 600 $LFS/dev/console c 5 1
```

```
mknod -m 666 $LFS/dev/null c 1 3
```

El següent que cal fer és muntar el directori /dev del sistema amfitrió al directori /dev del sistema hoste que estem bastint amb la instrucció mount. D'altra banda la solució plantejada per als sistemes de fitxers virtuals del nucli és similar: muntar-los directament amb la instrucció mount. Per aconseguir aquesta fita, ja que cada vegada que s'apagui el sistema cal tornar a repetir aquest muntatge, empro el següent script que evita haver d'escriure les instruccions de muntatge cada vegada que es vol accedir al sistema hoste:

```
ansgarius@Lenovo:~$ more ./scripInici.sh
#!/bin/bash
export LFS=/media/lfs
mount -t ext4 /dev/sda9 $LFS

mount -v --bind /dev $LFS/dev

mount -vt devpts devpts $LFS/dev/pts -o gid=5,mode=620
mount -vt proc proc $LFS/proc
mount -vt sysfs sysfs $LFS/sys
mount -vt tmpfs tmpfs $LFS/run

if [ -h $LFS/dev/shm ]; then
mkdir -pv $LFS/$(readlink $LFS/dev/shm)
fi

chroot "$LFS" /tools/bin/env -i HOME=/root TERM="$TERM" PS1='\u:\w\$ ' PATH=/bin:/usr/bin:/sbin:/usr/sbin:/tools/bin /tools/bin/bash --login +h

ansgarius@Lenovo:~$ █
```

A l'script anterior és pot observar com la darrera instrucció d'aquest és chroot. És a dir que es canvia el directori arrel de l'interpret de comandes, deixant de ser aquest el del

sistema amfitrió per passar a ser el de l'hoste (la variable `$LFS`, apunta a la partició on es basteix el sistema). Aquest canvi de directori arrel és clau per a la construcció del sistema LDZ final. La instal·lació de tots els programes de programari que el conformen s'ha de fer d'aquesta manera, per tal que un cop instal·lades aquestes eines sempre apuntin al directori arrel de la partició, esdevenint així aquest el directori arrel del sistema, que es el que es vol aconseguir. Cal fer notar que en fer aquest canvi de directori arrel, l'interpret de comandes que s'executa pren un entorn específic determinat amb les variables `HOME`, `TERM`, `PS1` i `PATH`.

La darrera tasca preliminar significativa que cal dur a terme és la creació dels fitxers `/etc/passwd` i `/etc/group`, per tal de crear els usuaris i grups, respectivament necessaris per construir correctament el sistema. Aquests fitxers de configuració es poden consultar a l'annex III.

## 6.2 Instal·lació dels paquets

El sistema LDZ que es pretén implementar parteix dels sistema LFS (*Linux From Scratch*) que es detalla al llibre *Linux from Scratch – version7.5*. Per tant cal primer de tot bastir exactament el sistema tal com es detalla en aquest llibre, per a continuació instal·lar els paquets específics del LDZ. Per aquest motiu en aquest punt aquests no se n'instal·laran, doncs això constituirà la tasca final, tal com es recomana dins del projecte BLFS (*Beyond Linux From Scratch*), que es el que farem servir per donar la forma final al nostre LDZ.

Per tant la instal·lació d'aquests paquets, en l'ordre en què estan llistats a l'annex II, es fa seguint fil per randa de manera mecànica les instruccions donades a l'apartat sis del

## llibre *Linux from Scratch*.

El tret més rellevant que cal destacar en aquest procés d'instal·lació rau, igual que en la construcció del sistema temporal, en la interrelació en la instal·lació d'aquells paquets de programari que inclouen eines de compilació, l'anomenat *toolchain* en anglès. Aquests paquets ara seran compilats, enllaçats i instal·lats per les eines del sistema temporal, que es troben a `/tools`, i ja no amb les del sistema amfitrió. Es pot observar com d'aquesta manera s'estableix la independència respecte aquell sistema. Per aquest motiu l'establiment d'aquesta *toolchain* és més fàcil, i no cal instal·lar determinats paquets, com el del compilador, el `gcc`, en dues fases. Ara només cal instal·lar-los un a un en l'ordre adient. En aquesta ocasió, el que primer s'ha d'instal·lar són les capçaleres de l'API de Linux per tal que el paquet `glibc`, que conté les llibreries de C, pugui fer referència a aquesta API quan s'instal·li a continuació. En aquest punt cal fer uns ajustaments per tal que l'enllaçador dinàmic que faci servir el compilador sigui el creat en l'apartat anterior, durant la instal·lació del sistema temporal. La fita següent és la instal·lació del paquet `binutils`, que conté tant l'assemblador com l'enllaçador que farà servir el compilador. El punt final del *toolchain* el marca doncs el paquet `gcc`, que farà servir les llibreries de C instal·lades amb `glibc`, amb l'API de Linux inclosa, i l'enllaçador dinàmic adient; així com l'assemblador i enllaçador acabats d'instal·lar amb el paquet `binutils`. Totes aquestes eines, així entrelligades, permetran en el moment adient la compilació del nucli de manera aïllada respecte el sistema amfitrió.

Així doncs, un cop instal·lats tots aquests paquets de programari, ja tenim bastit el sistema base que conformarà el LDZ, nomé hi manca el nucli, i els paquets que personalitzaran el sistema.

A la següent captura es pot observar a tall d'exemple les eines instal·lades als directoris /bin, /sbin i /usr/sbin (aquest darrer només parcialment degut al gran nombre d'aplicacions que el componen):

```

root:/# ls /bin
[
  bzip2  chown  dd      egrep  findmnt gzip  killall ls      lzma  mount  passwd  ps      sed  su      traceroute  unlzma  xzcat
bash    bzip2  chown  df      false  fuser  head   kmod  lsblk  mknod  mv      mountpoint  pidof  pwd  sh      sync  true  umount  unxz  zcat
bashbug cat    cp      dmesg  fgrep  grep  hostname ln     lsmod  mknod  mv      nice    ping  rm      sleep  tar     test  umount  wdctl  xz
bunzip2 chgrp  date   echo    flnd   gunzip kill  login  lzcat  more   mv      nice    ping  rmdir  stty   tar     test  uname  xz

root:/# ls /sbin
agetty  chcpu  e2label  fsck.ext4  grub-install  halt  killall5  mkfs.bfs  modinfo  resize2fs  runlevel  swapon  wipefs
badblocks  ctrlaltdel  e2undo  fsck.ext4dev  grub-mkconfig  hwclock  klogd  mkfs.cramfs  modprobe  rmmode  sfdisk  switch_root
blkdiscard  ctstat  fdisk  fsck.mntnx  grub-mknetdir  ifcfg  ldconfig  mkfs.ext2  nologin  route  shutdown  sysctl
blkid  debugfs  flndfs  fsfreeze  grub-ofpathname  ifconfig  lnstat  mkfs.ext3  nstat  routel  sln  syslogd
blockdev  depmod  fsck  fstab-decode  grub-probe  ifstat  logsave  mkfs.ext4  pivot_root  rtacct  ss  tc
bootlogd  dumpe2fs  fsck.cramfs  fstropt  grub-reboot  ifstat  losetup  mkfs.ext4dev  poweroff  rtmon  sulogin  telnet
bridge  e2fsck  fsck.ext2  genl  grub-set-default  init  mke2fs  mkfs.mntnx  raw  rtptr  swaplabel  tune2fs
cfdisk  e2image  fsck.ext3  grub-bios-setup  grub-sparc64-setup  ip  mkfs  mkswap  reboot  rtstat  swapoff  udevadm

root:/# ls /usr/bin/
32p  column  gcc-ranlib  info  lzmadec  pcprofiledump  ranlib  splain  vim
aclocal  comm  gcov  infocmp  lzmainfo  pdfmcom  rcp  split  vimdiff
aclocal-1.14  compile_et  gdbm_dump  infokey  lzmore  pdfroff  readelf  sprof  vintutor
addftinfo  config_data  gdbm_load  infotocap  m4  pdftexi2dvi  readlink  stat  vmstat
addr2line  corelist  gdbmtool  install  make  peekfd  realpath  stdbuf  w
afmtodit  cpan  gdiffmk  install-info  makeadb  perl  recode-sr-latin  strings  wall
apropos  cpan2dist  gencat  instmodsh  makeinfo  perl5.18.2  refer  strip  watch
ar  cpanp  geqn  ionice  man  perlbug  rename  sum  wc
as  cpanp-run-perl  getconf  ipcmk  mandb  perlloc  renice  tabs  whatis
autoconf  cpp  getent  ipcrm  manpath  perltp  reset  tac  whereis
autoheader  csplit  getkeycodes  ipcs  mapscrn  perlthanks  rev  tail  who
    
```

## **7 Establiment dels scripts d'arrencada del sistema**

Un cop instal·lats tots els paquets de programari que compondran la base del sistema LDZ cal crear els fitxers de configuració adients que permetin l'arrencada del sistema. Bona part d'aquests fitxers de configuració s'instal·len amb un paquet que els conté: `LFS-Bootscripts-20130821`. Aquest paquet està creat pel desenvolupadors de *Linux from scratch*, evitant així el fet d'haver de crear un a un tots els fitxers de configuració. Tanmateix n'hi ha altres, que depenen de les característiques pròpies de l'ordinador on s'instal·larà el sistema que sí que han de ser creats manualment.

Entre aquests darrers es troben els pertinents per configurar la xarxa. Entre aquests els de la configuració de les diferents interfícies de xarxa que conté l'ordinador, on s'especifiquen les adreces IP, la màscara de subxarxa, la porta de sortida, etc. Dos són els creats, a `/etc/sysconfig` un per cada interfície de xarxa, la de cable, `eth0` i la sense fils, `wlan0`. Duen els noms `ifconfig.eth0` i `ifconfig.wlan0` respectivament. El seu contingut es pot consultar a l'annex III.

Una altra acció que cal realitzar és posar un nom al sistema hoste, en aquest cas LDZ, al fitxer `/etc/sysconfig/network` simplement amb la instrucció: `echo "HOSTNAME=LDZ" > /etc/sysconfig/network`.

La resta de fitxers de configuració que resten per crear poden ser modificats a partir de la seva forma per defecte conforme les preferències de l'usuari. Tanmateix per a la construcció del LDZ farem servir la configuració per defecte. El primer d'aquests fitxers que cal crear expressament és `/etc/inittab`, que és el que fa servir el procés inicial `init` quan s'arrenca el sistema. Amb aquest script s'indica la iniciació dels scripts que es

troben a `/etc/rc.d`, que han esta instal·lats anteriorment amb el paquet `LFS-Bootscripts-2013082`. Un altre és el que correspon al rellotge del sistema a `/etc/sysconfig/clock`. El darrer que obligatòriament cal crear és `/etc/inputrc` que especifica el tractament del mapatge del teclat en el cas de circumstàncies específiques. Tots aquests fitxers de configuració es poden consultar a l'annex III.



## **8 Instal·lació del nucli i arrencada del sistema**

L'establiment dels scripts de configuració del sistema duts a terme a l'apartat anterior era el pas immediatament previ a la compilació i instal·lació del nucli o kernel del sistema operatiu. Doncs és precisament el nucli qui es configurarà conforme els paràmetres especificats en aquests scripts per tal d'arrencar i funcionar de la manera allà especificada.

Tanmateix abans de procedir a la implementació del nucli es requereix encara l'elaboració d'un fitxer on es recullin els sistemes de fitxers als quals aquest tindrà accés. Aquest fitxer, que s'ubica com la resta de fitxers de configuració al directori `/etc`, rep el nom de `fstab`, file systems table, taula de sistemes de fitxers. En ella s'especifiquen els mateixos sistemes de fitxers, el físic a la partició de disc on s'està instal·lant el LDZ, així com els virtuals `proc`, `sysfs`, `devpts`, `tmpfs` i `devtmpfs` en els mateixos punts de muntatge `/proc`, `/sys`, `/dev/pts`, `/run` i `/dev` respectivament, que es muntaven com a pas previ a la instal·lació de tots els paquets de programari descrits en el punt sis. Si llavors aquests sistemes de fitxers i punts de muntatge eren necessaris per a la compilació i instal·lació adient dels paquets, així com per a l'us del `chroot` que permetia executar i emprar aquestes eines com a sistema hoste en el sistema amfitrió, ara ho seran per a la seva execució com a sistema amfitrió pròpiament, exclusivament per part del nucli que s'instal·larà i es carregarà directament pel gestor d'arrencada. Només hi ha una petita variació, i és que en aquesta taula de sistemes de fitxers cal especificar la partició d'intercanvi. Aquesta no calia especificar-se durant la implementació del sistema des del sistema amfitrió, ja que aquest, amb el seu nucli corresponent era qui accedia directament a aquesta partició. Tanmateix com l'execució d'aquest sistema ja no com a hoste, sinó de manera nativa pròpiament, cal que sigui el nucli que es compilarà a tal

efecte qui accedeixi directament a la partició d'aquest espai d'intercanvi, motiu pel que és necessari la seva especificació a la taula de sistemes de fitxers. A la següent captura es visualitza el fitxer `/etc/fstab` del sistema LDZ:

```
root:/# more /etc/fstab
# Begin /etc/fstab

# file system mount-point type      options                dump    fsck
#                                     order

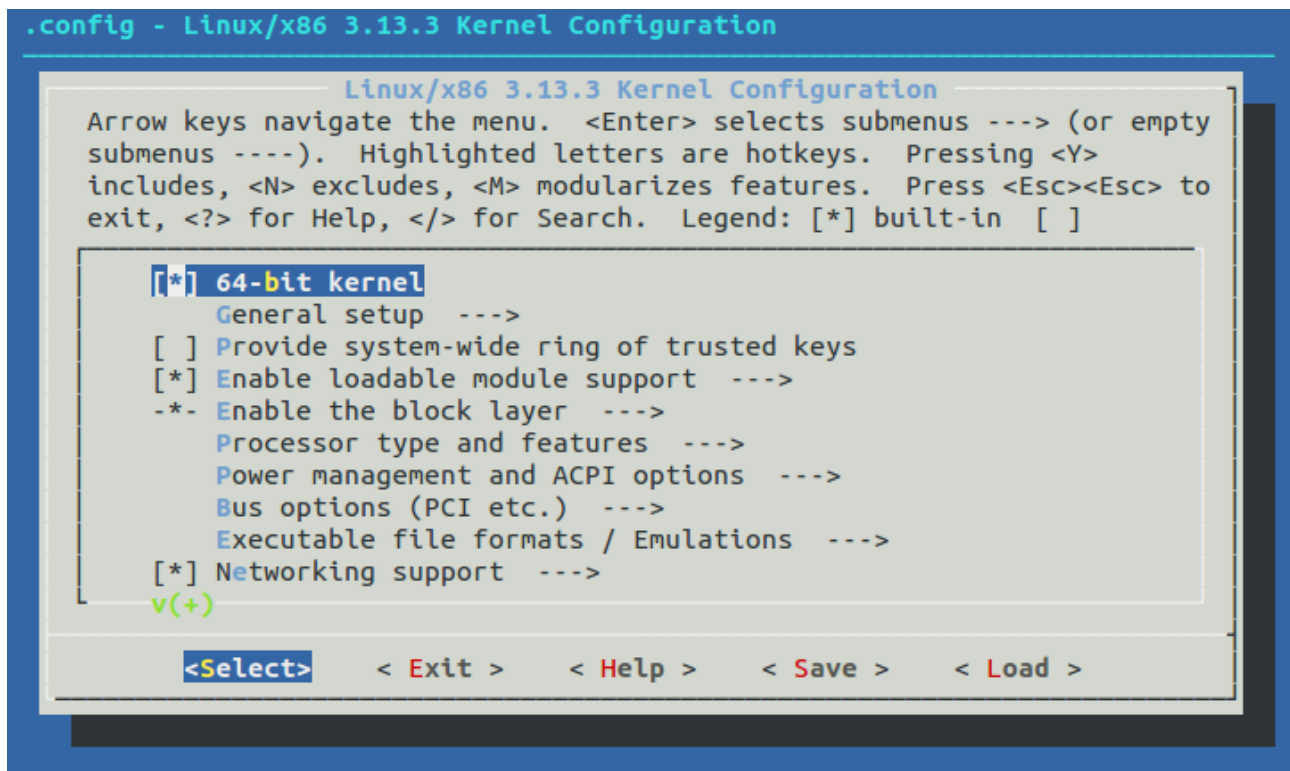
/dev/sda9    /                ext4                  defaults            1       1
/dev/sda5    swap             swap                  pri=1               0       0
proc         /proc            proc                  nosuid,noexec,nodev 0       0
sysfs        /sys             sysfs                 nosuid,noexec,nodev 0       0
devpts       /dev/pts         devpts                gid=5,mode=620      0       0
tmpfs        /run             tmpfs                 defaults             0       0
devtmpfs     /dev             devtmpfs              mode=0755,nosuid    0       0

# End /etc/fstab
root:/#
```

El pas següent consisteix en la compilació i instal·lació del paquet `linux-3.13.3.tar.xz`, el qual conté nucli del sistema LDZ que estem construint. Aquesta és la culminació del procés encetat fins ara. Cal recordar que per a poder dur a terme aquesta compilació primer s'ha hagut d'implementar un sistema mínim compilat amb les eines del sistema amfitrió, i per tant encara dependents d'aquest. És per aquest motiu que el nucli no podia compilar-se directament per les eines d'aquell sistema, doncs aquest nucli ha de ser totalment independent, lliure de dependències per poder carregar-se directament a la màquina com a sistema amfitrió ell mateix. Calia configurar, compilar i instal·lar el compilador que compilés el nucli amb aquest sistema intermedi, descrit a l'apartat 5, en un entorn senzill i definit amb unes variables d'entorn

controlades, cosa que el sistema amfitrió no pot oferir. Per tant amb aquestes eines s'ha construït el sistema, amb les eines i programes, que efectivament conformaran el sistema definitiu, entre elles el compilador. Aquest degudament configurat, amb l'encadenament adient amb les llibreries de C de `glibc`, que farà servir, amb un enllaçador dinàmic propi d'aquest sistema i que no empri el del sistema amfitrió, així com amb l'assemblador del paquet `binutils` tal com s'ha explicat en els dos apartats cinc i sis. I sobretot, ara cobra sentit, l'exportació que s'ha fet, també en aquells dos punts, de les capçaleres de l'API del nucli, l'única acció directament relacionada amb el paquet `linux-3.13.3.tar.xz` que el conté duta a terme fins ara. D'aquesta manera el compilador `gcc` està configurat de tal manera que coneix aquesta API i per tant està optimitzat per poder compilar aquest nucli de manera eficient i independent del sistema amfitrió fins el punt que pugui ser arrencat per l'ordinador com a sistema operatiu nadiu.

Abans de compilar el nucli però cal procedir a la seva configuració. En aquest cas, el paquet que el conté no disposa d'un script `configure` que automatitza aquesta tasca com a la resta de paquets. És un xic diferent, doncs cal emprar una opció que és `menuconfig` a l'ordre `make`. Això fa que aparegui un menú de configuració que permet especificar un munt d'opcions i paràmetres per compilar el nucli. A la següent imatge es pot observar aquest menú:



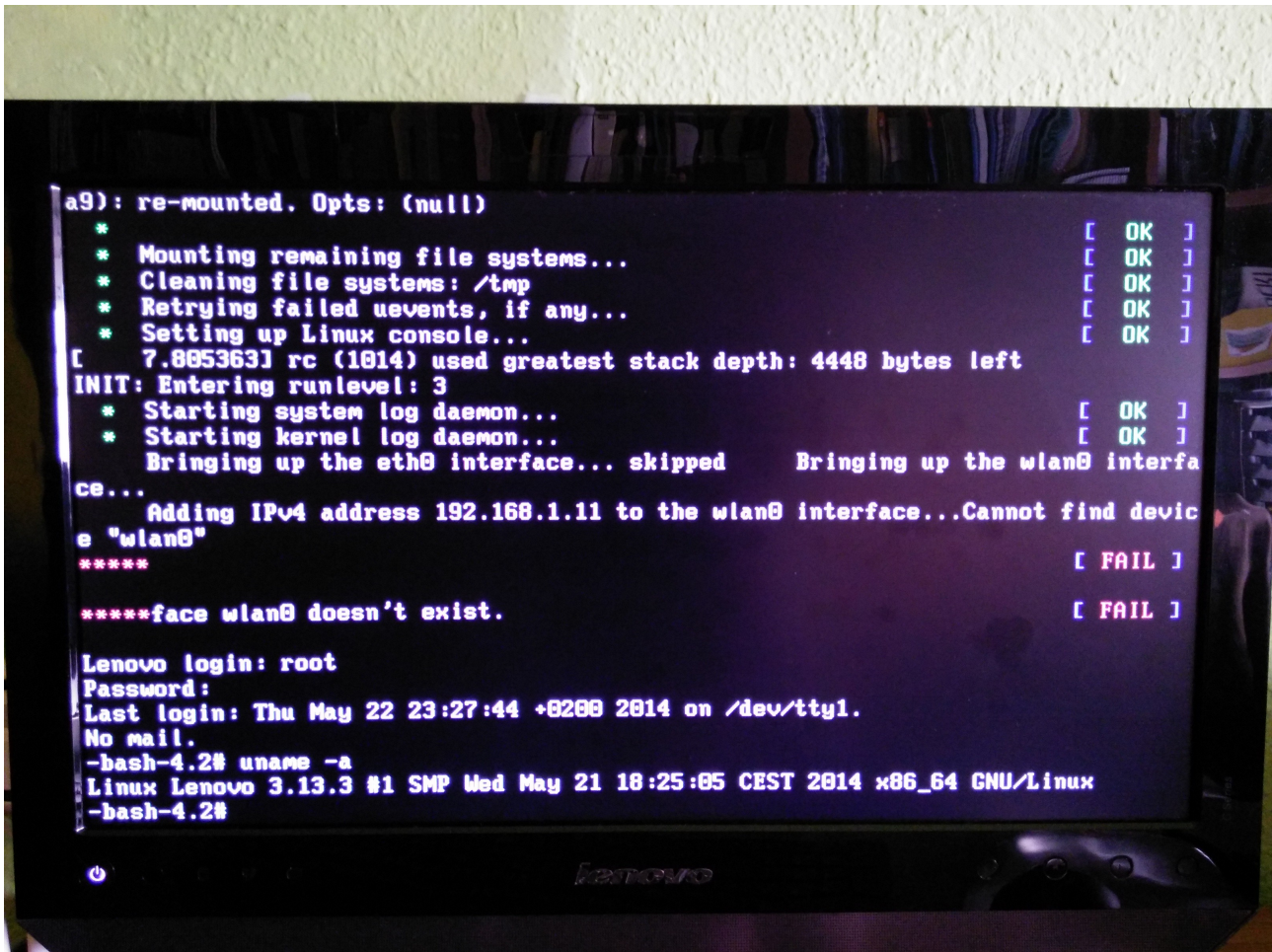
De totes maneres aquest menú ja conté una configuració del nucli per defecte que és la que he fet servir per compilar-lo, ja que la majoria d'opcions indicades sobrepassen amb escreix els meus coneixements.

La compilació en ella mateixa no ofereix cap mena de complicació, és idèntica a la de qualsevol altre paquet. El resultat, de manera anàloga al binari executable que en resulta da la compilació de la resta d'eines, és un arxiu comprimit de nom bzImage que conté el nucli executable que s'acaba de compilar. Aquest arxiu cal copiar-lo dins del directori /boot del sistema que serà on el gestor d'arrencada anirà a trobar-lo per engegar el sistema operatiu que es faci amb el control de l'ordinador. La següent captura conté els contingut del directori /boot:

```
root:~# ls /boot/
config-3.13.3  grub  System.map-3.13.3  vmlinuz-3.13.3-lfs-7.5
root:~#
```

A la captura anterior es pot constatar com el nom del fitxer comprimit del nucli s'ha reanomenat `vmlinuz-3.13.3-lfs-7.5`, ja que `vmlinuz` és un nom convencional per aquest fitxer. També s'aprofita per indicar a la resta del nom la versió del nucli i el de la distribució, que en aquest cas és la del *Linux From Scratch 7.5* que hem bastit.

Arribats a aquest punt disposem a més de les eines bàsiques i fonamentals d'un sistema Linux que ja eren usables des de l'entorn del `chroot` al sistema amfitrió, d'un nucli executable a la partició on s'han instal·lat aquestes eines. Per tant el sistema ja està construït. Només cal configurar el gestor d'arrencada del sistema ubicat al MBR del disc dur on es troba la partició del LDZ per permetre que carregui el nucli que acabem de compilar i fer servir a partir d'ara de manera directa, sense cap amfitrió el LDZ. Al llibre *Linux From Scratch* s'indica que per fer aquesta tasca s'instal·li al MBR el gestor `grub` que s'ha compilat i instal·lat en l'apartat 6 i es generi un fitxer de configuració amb el nom `grub.cfg` a `/boot/grub/` per tal d'aconseguir aquest objectiu. El resultat que s'obté és certament la càrrega i execució del nucli que s'ha compilat tal com es pot observar a la següent imatge:

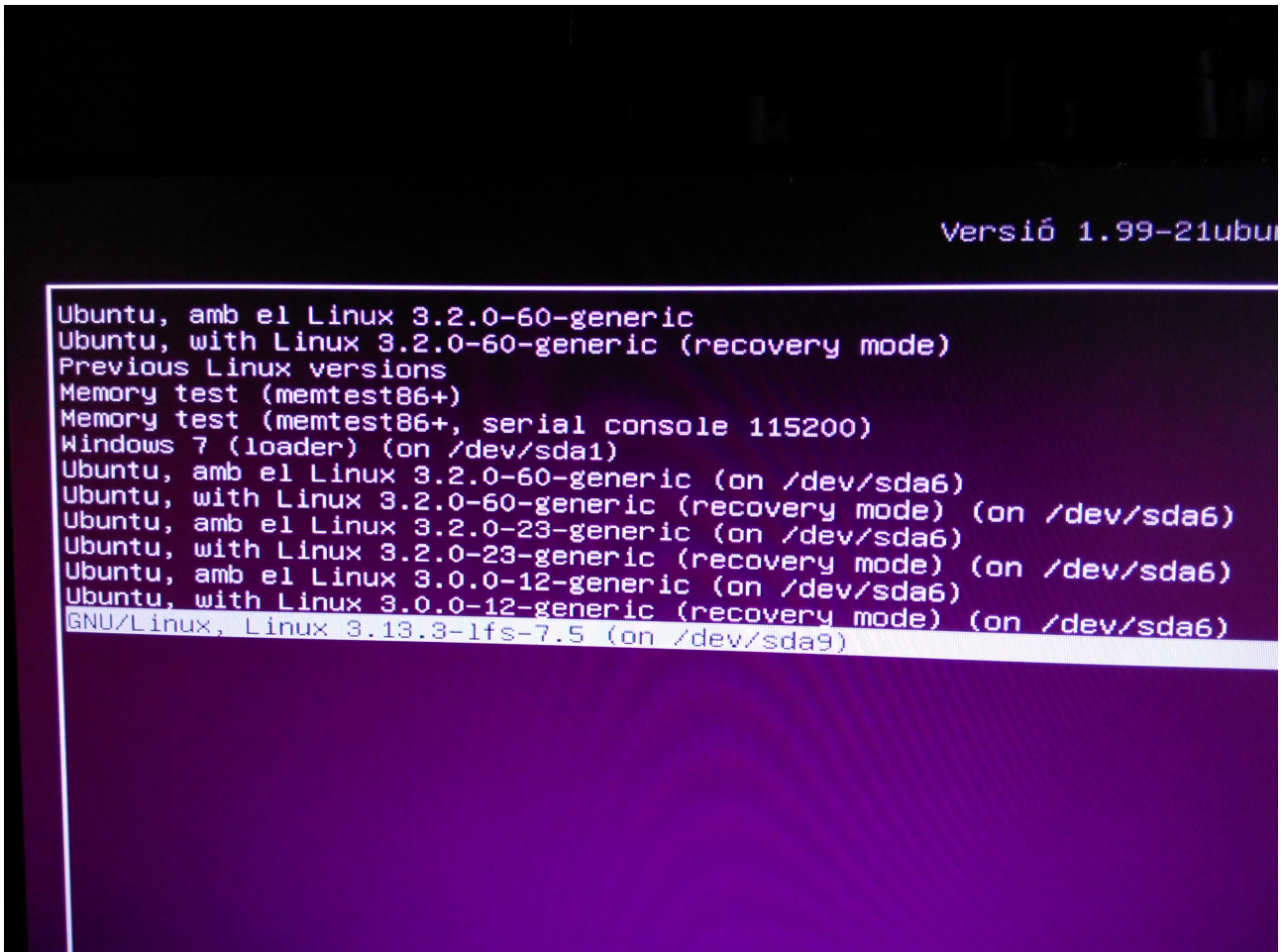


Cal notar a la imatge anterior com durant la càrrega del sistema es dona un missatge d'error. El problema rau en el driver de la targeta wireless de tipus usb que faig servir per connectar-me a l'encaminador adsl amb el que tinc accés a Internet. El nucli no en disposa d'un adient, no pot comunicar-se amb aquest dispositiu i per això avisa que no el pot trobar, malgrat que en el fitxer de configuració `ifconfig.wlan0` ubicat a `/etc/sysconfig/` s'hagi indicat la seva existència.

Tanmateix el gestor grub d'arrencada que s'instal·la d'aquesta manera només reconeix de tots els instal·lats a les diferents particions del disc dur de l'ordinador el sistema operatiu LDZ. És poc respectuós amb els altres sistemes. Per tal de fer possible l'ús de tots els que en tinc instal·lats el que he fet és carregar un sistema Ubuntu amb l'ajuda



d'un disc d'arrencada i instal·lar el grub al MBR des d'aquest Ubuntu, així com fer servir l'opció update per tal que em reconegui i permeti la càrrega de tots els sistemes operatius, inclòs el LDZ tal com s'observa a la següent imatge:



## 9 Implementació de l'escriptori gràfic

Com l'objectiu d'aquest treball és la construcció des de zero d'un sistema operatiu gnu-linux que pugui ser emprat en un aula de secundària, disposar d'un sistema operatiu només en mode text, interaccionant amb l'ordinador a través d'un intèrpret de comandes és del tot insatisfactori. Cal implementar una interfície gràfica intuïtiva que qualsevol usuari habituat a fer servir un ordinador pugui fer servir sense que calgui tenir cap coneixement tècnic. Per tant es fa necessari la compilació i instal·lació d'un escriptori gràfic. En l'àmbit de gnu-linux els dos escriptoris gràfics més emprats i coneguts són KDE i Gnome, per aquest motiu he deixat de banda considerar altres opcions i escollir-ne un d'aquests.

D'altra banda l'equip que ha desenvolupat i manté el projecte *Linux From Scratch*, que he seguit fins ara, l'han ampliat per donar suport a tot un seguit d'utilitats i programes que sobre el *Linux From Scratch* bàsic, permeten personalitzar i adaptar el sistema a les necessitats personals de cadascú. Aquesta continuació del projecte inicial rep el nom de BLFS (*Beyond Linux From Scratch*), accessible a l'adreça web <http://www.linuxfromscratch.org/blfs/view/svn/>. Seguint les pautes allà indicades he triat instal·lar el KDE com a entorn gràfic i no el KDE degut a que el primer no acaba de funcionar prou bé sobre el sistema bàsic del LFS. Aquesta decisió és el primer pas per personalitzar el LFS ja implementat per convertir-lo en el *Linux des de Zero*, LDZ, que és l'objectiu d'aquest treball.

A diferència de la implementació del LDZ on es seguia un document linealment, on els paquets amb el codi font eren uns concrets, que fins i tots els podies descarregar tots de cop, a BLFS desapareix aquesta linealitat. Per exemple en el cas del KDE s'especifiquen el nombre de paquets que el componen, 20 en total, amb les indicacions, força ben

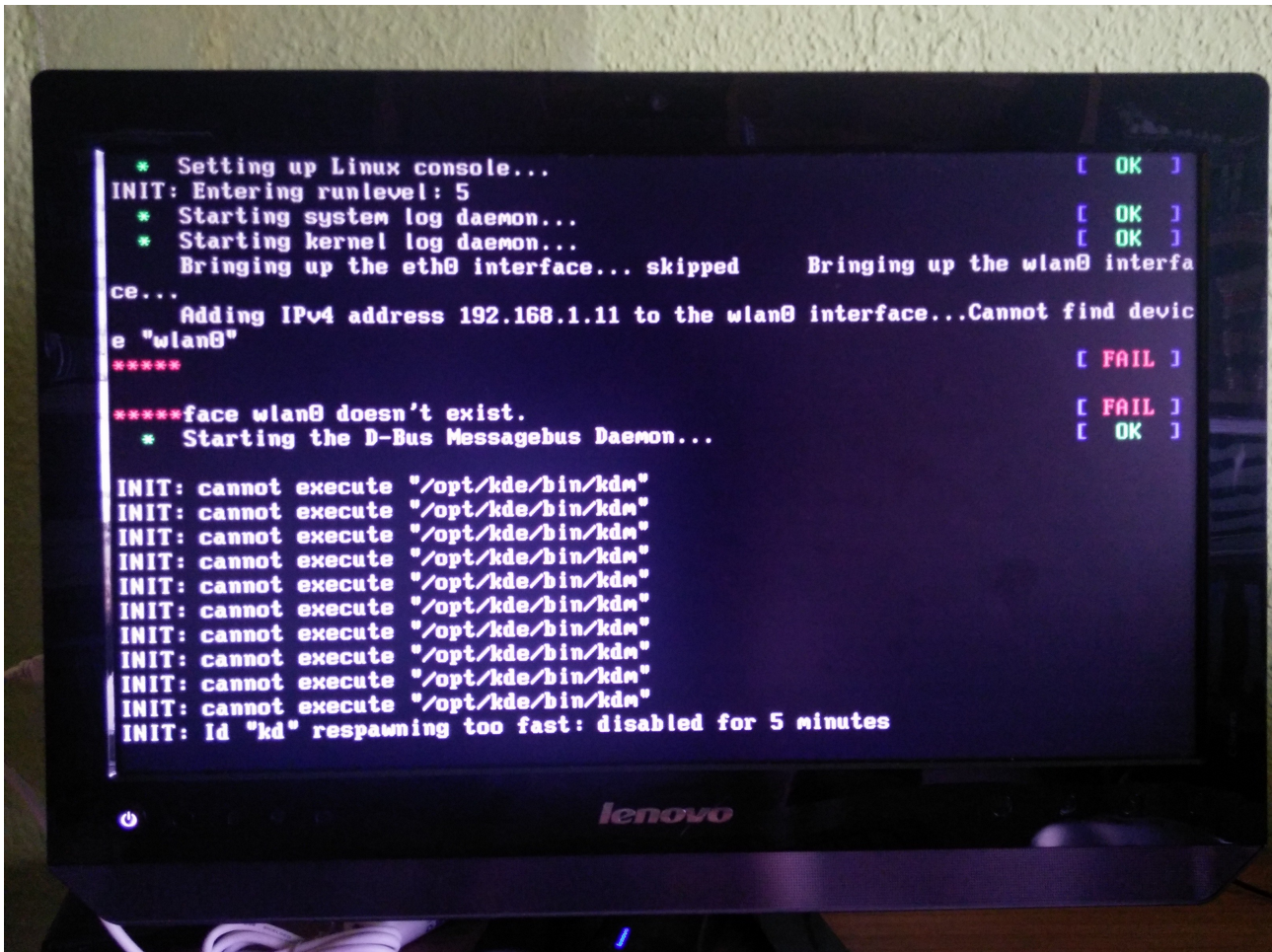


pautades per poder-los instal·lar. Tanmateix en aquest punt m'he trobat amb un contratemps, que a la llarga ha estat capital pel desenvolupament i resultat final d'aquest treball. En fer el plantejament inicial del projecte, concretament el pla de treball, havia suposat que l'entorn gràfic havia de consistir en un paquet principal, prou gros, això sí, i amb un nombre reduït de dependències. Reconec que l'ús habitual que faig d'Ubuntu i del programa gestor de paquets *Synàptic* em va induir a aquest error. Normalment quan s'instal·la un programa d'aquesta manera el nombre de paquets totals és relativament reduït. Per això m'esperava que en el desenvolupament del LDZ la situació seria similar, un nombre reduït de paquets, que això sí caldria compilar abans d'instal·lar-los. Només així es pot explicar que en el pla de treball ni tan se vols vaig incloure una entrada específica per a la seva instal·lació, creient que els podria instal·lar conjuntament amb els que componen el LFS pròpiament. Tanmateix seguint els fils de dependències dels 20 paquets que componen pròpiament el KDE, em vaig trobar amb vora 200 paquets necessaris per implementar l'escriptori gràfic. La totalitat dels paquets instal·lats, amb les seves dependències es recullen a l'Annex IV. Aquest fet no feia només trontollar el meu projecte, sinó que el feia inabastable. Amb prou feines podria acabar la instal·lació de tots aquests paquets, i això gràcies, a que en previsió de possibles contratemps, havia deixat al pla de treball el període entre la data de lliurament de la PAC 4 i la memòria final sense cap feina. Comentant degudament aquesta situació amb el meu consultor, ell em va recomanar que em centrés en instal·lar l'escriptori gràfic, deixant de banda les altres tasques, en especial la creació d'un live-cd del LDZ resultant.

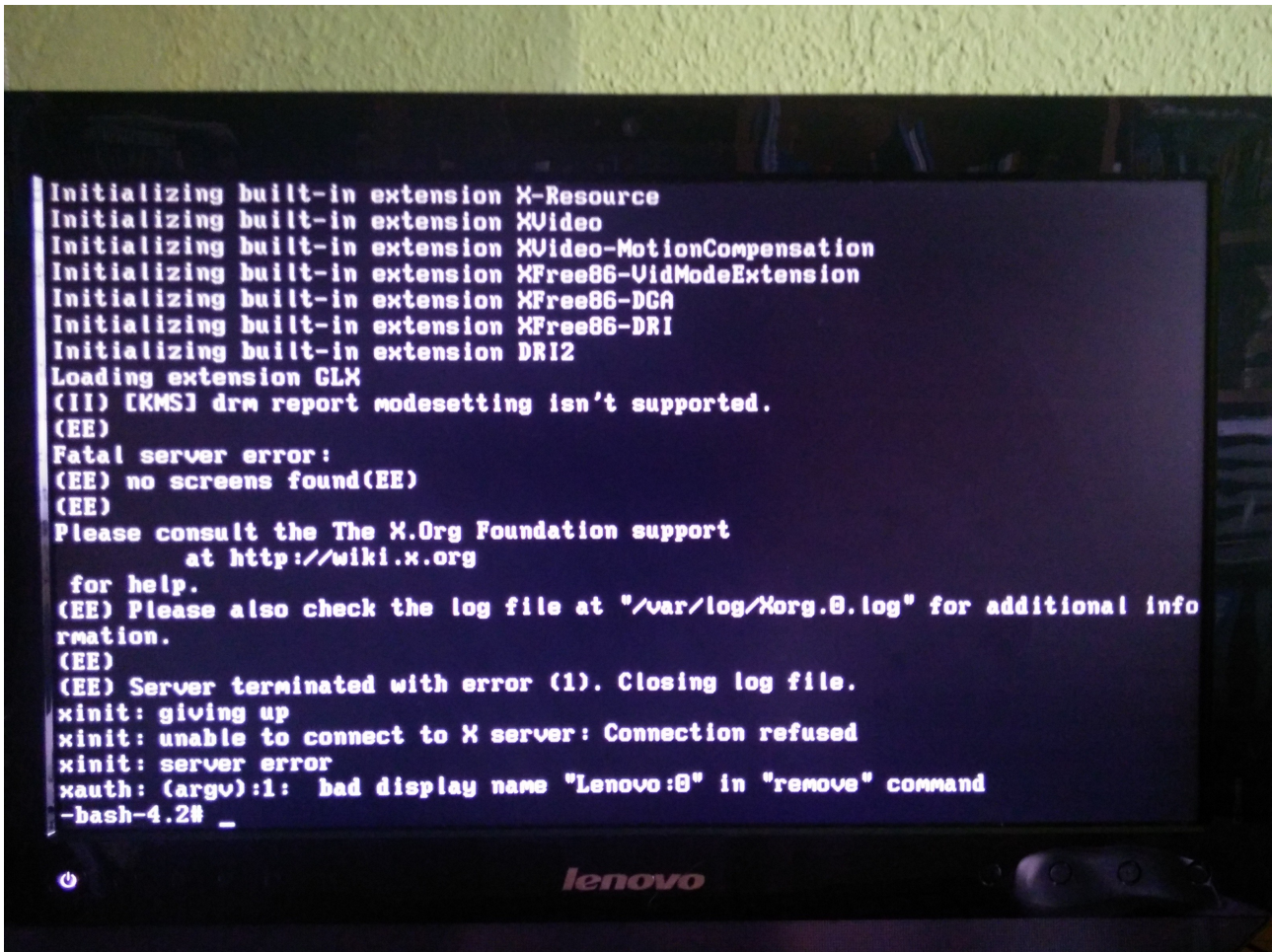
Així doncs, un cop decidit a limitar-me a implementar l'escriptori gràfic només calia partir dels 20 paquets que composaven el KDE, establir les seves dependències i anar instal·lant un a un els paquets, seguint les indicacions que per a cada un d'ells es recollia al projecte BLFS. Aquesta és una tasca mecànica i sistemàtica que no té cap aspecte

rellevant a destacar tret d'un de sol, que em va permetre entendre un aspecte dels escriptoris gràfics que ja m'havia trobat amb l'ús de les distribucions de Linux: el servidor de les X. De fet creia que formava part pròpiament de l'escriptori gràfic ja fos el KDE o el Gnome. Tanmateix amb la implementació paquet a paquet quedava clar que el KDE no s'instal·la directament, sinó que primer cal bastir-ne un entorn gràfic més bàsic i primari, l'X Window System Environment, l'entorn de sistema de finestres X. El qual funciona a partir d'un model client servidor. El servidor d'aquest, el que ja coneixia com el servidor de les X, és l'encarregat de rebre les peticions a través del teclat o el ratolí dels programes clients per tal que aquest els hi doni sortida en forma de finestres gràfiques. Aquesta és la base dels sistemes de finestres dels sistemes Unix, del qual LFS és un exemple. Per tant una de les dependències demanades per alguns paquets de KDE, no corresponien a un paquet pròpiament, sinó al sistema de les X en la seva totalitat, el servidor i les eines necessàries pel seu correcte funcionament, ja que el KDE com a escriptori gràfic fa ús de finestres que són servides per aquest sistema. Per aquest motiu a l'Annex IV es troben dividits els paquets instal·lats en dos apartats, aquells que formen part de l'X Window System Environment i aquells que formen pròpiament el KDE.

Finalment, un cop compilats i instal·lats els aproximadament 200 paquets que conformen tant el sistema de les X com el KDE, en arrencar el sistema, ja es podria carregar l'entorn gràfic, passant de ser pròpiament el sistema una concreció del *Linux From Scratch* a una del *Linux des de Zero* que era la fita final d'aquest treball. Tanmateix en arrencar el sistema no s'obria l'entorn gràfic, sinó que el resultat obtingut era el que s'observa a la següent imatge:



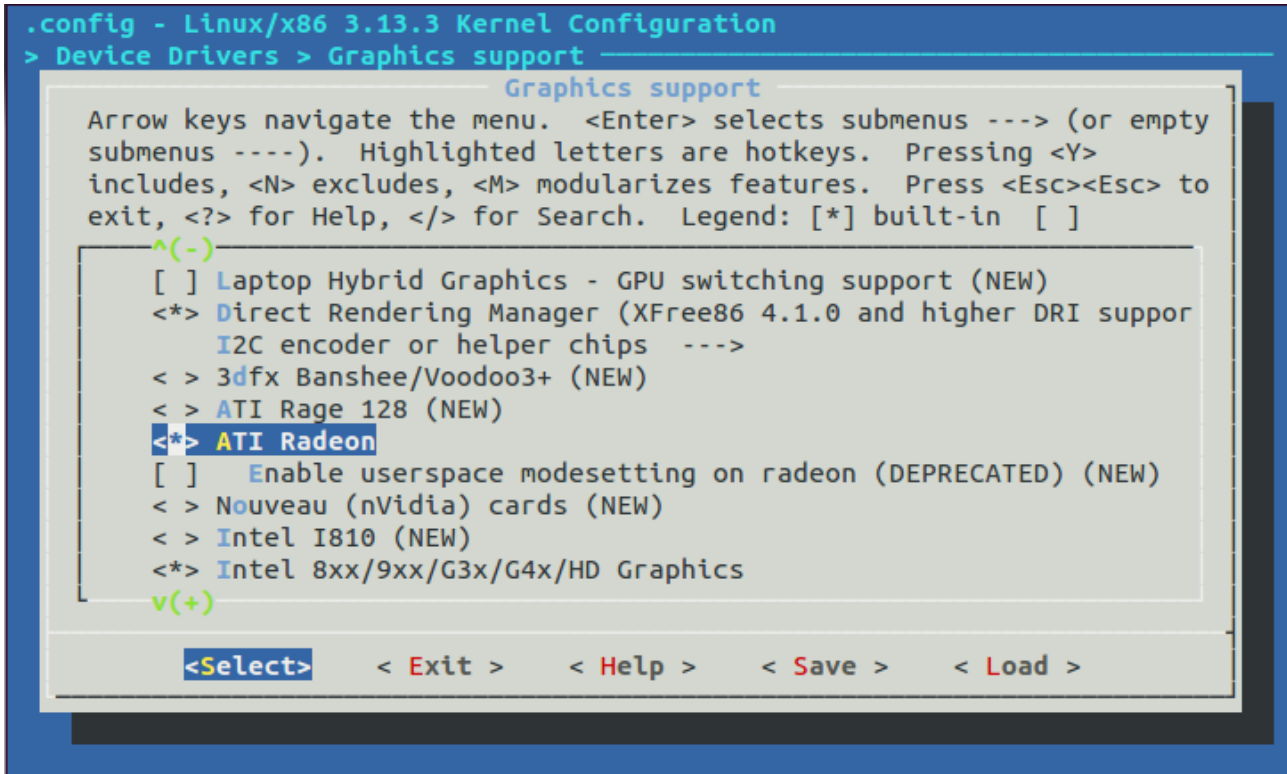
És evident que hi havia un problema en la càrrega de l'escriptori gràfic. Aquest resultat s'obtenia amb un mode d'execució, `runlevel 5`, el que correspon al mode multiusuari gràfic, degudament indicat al fitxer de configuració `/etc/inittab`. Provant que passava modificant el paràmetre corresponent al mode d'execució de cinc a tres per iniciar el sistema en mode normal, és a dir amb un terminal de text, el resultat obtingut era el següent:



La informació donada pel sistema indicava en una línia: [KMS] drm report modesetting isn't supported. Una mica de recerca per Internet donava com a resultat que KMS significa Kernel mode-setting, que per un costat indicava que el problema estava relacionat amb el nucli, i a més que tenia a veure amb aspectes gràfics com la resolució de la pantalla, el color, etc. Quelcom força relacionat amb les exigències d'un escriptori gràfic. Fent una mica més de recerca, començant per la mateixa pàgina de BLFS, tot apuntava que la opció KMS, és una de les opcions disponibles en el moment de configurar el nucli. Era evident que la configuració per defecte que havia fet servir per construir el LFS era incompatible amb la càrrega d'un escriptori gràfic. S'havia de tornar a configurar el nucli, compilar-lo i instal·lar-lo de nou. Afortunadament només calia conèixer la targeta gràfica del meu ordinador, que és una ATI Radeon, i habilitar dins del

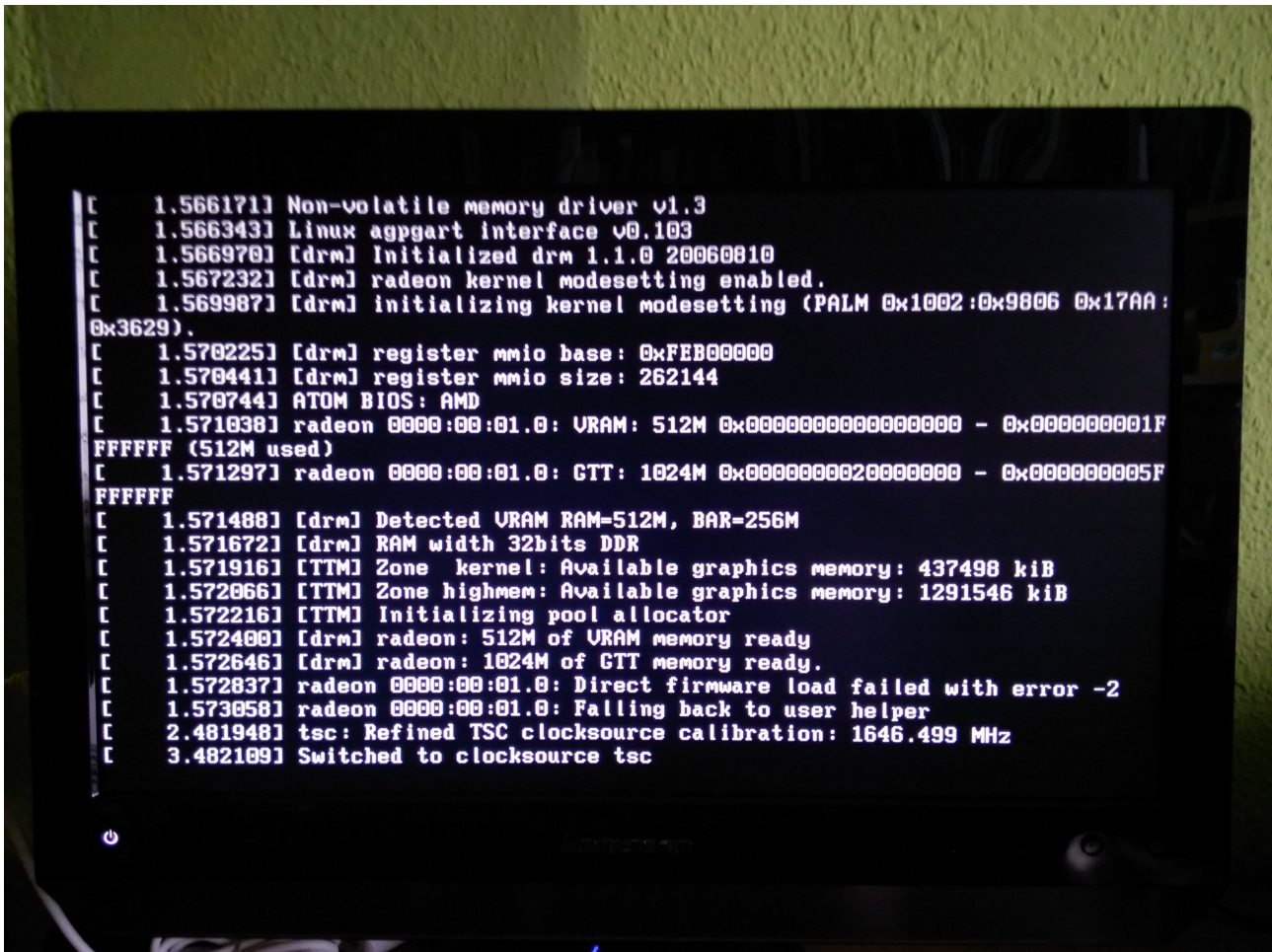


menú de configuració del nucli l'opció que es troba a Device Drivers -> Graphics support -> ATI Radeon, com es pot veure a la següent captura:



```
.config - Linux/x86 3.13.3 Kernel Configuration
> Device Drivers > Graphics support
    Graphics support
    Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty
    submenus ----).  Highlighted letters are hotkeys.  Pressing <Y>
    includes, <N> excludes, <M> modularizes features.  Press <Esc><Esc> to
    exit, <?> for Help, </> for Search.  Legend: [*] built-in [ ]
    ^(-)
    [ ] Laptop Hybrid Graphics - GPU switching support (NEW)
    <*> Direct Rendering Manager (XFree86 4.1.0 and higher DRI support
    I2C encoder or helper chips --->
    < > 3dfx Banshee/Voodoo3+ (NEW)
    < > ATI Rage 128 (NEW)
    <*> ATI Radeon
    [ ] Enable userspace modesetting on radeon (DEPRECATED) (NEW)
    < > Nouveau (nVidia) cards (NEW)
    < > Intel I810 (NEW)
    <*> Intel 8xx/9xx/G3x/G4x/HD Graphics
    v(+)
    <Select>  < Exit >  < Help >  < Save >  < Load >
```

Un cop compilat i instal·lat el nucli al directori /boot, ja s'estava en disposició de carregar l'escriptori gràfic un cop s'iniciés el sistema. Efectivament a la següent pantalla, que es la que apareix en un primer moment es pot observar que el KMS, Kernel mode-setting, ja està habilitat:



A continuació apareix una pantalla on es pot apreciar que la resolució és una altra i que per tant el canvi fet al nucli és efectiu:

```

[ 122.714815] netconsole: network logging started
[ 122.717543] acpi-cpufreq: overriding BIOS provided _PSS data
[ 122.720423] ALSA device list:
[ 122.723048]  #0: HDA ATI SB at 0xfef40000 irq 16
[ 122.776749] usb 1-4: New USB device found, idVendor=1658, idProduct=0740
[ 122.775321] usb 1-4: New USB device strings: Mfr=1, Product=2, SerialNumber=5
[ 122.776240] usb 1-4: Product: Fly Passport 0740
[ 122.776840] usb 1-4: Manufacturer: Fly Passport 0740
[ 122.781432] usb 1-4: SerialNumber: Western Digital
[ 122.788328] usb-storage 1-4:1.0: USB Mass Storage device detected
[ 122.791127] scsi4 : usb-storage 1-4:1.0
[ 122.998356] usb 2-5: new high-speed USB device number 3 using ehci-pci
[ 123.012432] md: Waiting for all devices to be available before autodetect
[ 123.015215] md: If you don't use raid, use raid=noautodetect
[ 123.018458] md: Autodetecting RAID arrays.
[ 123.021160] md: Scanned 0 and added 0 devices.
[ 123.023744] md: autorun ...
[ 123.026290] md: ... autorun DONE.
[ 123.029302] EXT4-fs (sda9): couldn't mount as ext3 due to feature incompatibilities
[ 123.032591] EXT4-fs (sda9): couldn't mount as ext2 due to feature incompatibilities
[ 123.063912] EXT4-fs (sda9): mounted filesystem with ordered data mode. Opts: (null)
[ 123.066570] VFS: Mounted root (ext4 filesystem) readonly on device 8:9.
[ 123.109609] devtmpfs: mounted
[ 123.112811] Freeing unused kernel memory: 620K (c1ac7000 - c1bd2000)
[ 123.115813] Write protecting the kernel text: 7920K
[ 123.118560] Write protecting the kernel read-only data: 2624k
[ 123.198541] request_module: runaway loop modprobe binfmt-464c
[ 123.199565] usb 2-5: New USB device found, idVendor=0a08, idProduct=c012
[ 123.199560] usb 2-5: New USB device strings: Mfr=1, Product=2, SerialNumber=0
[ 123.199571] usb 2-5: Product: Lenovo USB2.0 UVC Camera
[ 123.199574] usb 2-5: Manufacturer: Umicore Co., Ltd
[ 123.213042] kworker/u:1 (1005) used greatest stack depth: 6500 bytes left
[ 123.213042] kworker/u:1 (1003) used greatest stack depth: 6400 bytes left
[ 123.219424] Starting init: /sbin/init exists but couldn't execute it (error -8)
[ 123.287499] request_module: runaway loop modprobe binfmt-464c
[ 123.282117] Starting init: /bin/sh exists but couldn't execute it (error -8)
[ 123.294724] Kernel panic - not syncing: No working init found. Try passing init= option to kernel. See Linux Documentation/init.txt for guidance.
[ 123.297424] CPU: 1 PID: 1 Comm: swapper/0 Not tainted 2.13.3 #1
[ 123.300409] Hardware name: Lenovo Lenovo C225/Intra C20, BIOS E2N117MS 07/06/2011
[ 123.302399] 00000000 00000000 f5c7b770 c17a3c03 807e4042 f5c7b730 c17a3c2e c1971a10
[ 123.302405] c1730660 c196548e 007a4042 00000000 c1af3a00 f5c7b7ac c17a3c64 c1960070
[ 123.300451] 00000000 f5c03f04 f5c7a000 c17a7277 c17a3a70 00000000 00000000 00000000
[ 123.311290] Call trace:
[ 123.314073] [] dump_stack+0x11/0x12
[ 123.313070] hub 4-0:1.0: debounce: port 3: total 100ms stable 100ms status 0x01
[ 123.315063] [] panic+0x7d/0x160
[ 123.322824] [] kernel_init+0x14/0x100
[ 123.323070] [] ret_from_kernel_thread+0x1b/0x20
[ 123.328223] [] ? rest_init+0x43/0x60
[ 123.331079] drn_hns_helper: panic occurred, switching back to text console

```

Tanmateix apareix el següent error:

```

Starting init: /sbin/init exists but couldn't execute it (error -8)
...
Starting init: /sbin/sh exists but couldn't execute it (error -8)
Kernel panic - not syncing: No working init found...

```

Un error relacionat amb el nucli. De fet apareix independentment del mode d'execució, runlevel, amb el qual s'iniciï el sistema, ja sigui en mode gràfic, cinc, o en mode text, tres. Per tant dedueixo que és independent de l'escriptori gràfic. De fet no puc ni constatat si aquest ha estat correctament instal·lat o no, ja que el sistema falla abans.

Buscant per Internet, em trobo no gaire informació sobre aquest problema, força tècnica, molt per sobre de les meves capacitats i possibilitats. El màxim d'aigua clara que en puc treure és que el problema pot estar relacionat amb una incompatibilitat entre la

versió del nucli compilat quan s'executa en un ordinador de 64 bits. En cas de ser així, sembla que aquesta es dóna quan s'habilita el suport gràfic almenys per les targetes ATI Radeon, ja que amb la compilació del nucli amb les opcions per defecte no es dóna aquest problema. Comentant aquest fet amb el meu consultor m'adverteix que el problema no és trivial, que caldria investigar-ho més a fons, però que ja no hi havia prou temps per redreçar-lo. Haig de matissar que amb el temps destinat a compilar i instal·lar l'enorme nombre de paquets de l'escriptori només en vaig disposar de dos dies per enllestir el projecte abans del seu lliurament, pel qual em veig obligat a deixar l'estat d'aquest treball en aquest punt.



## 10 Conclusions

Malgrat la profunda decepció resultant del fracàs final per no haver pogut carregar l'escriptori gràfic en el sistema *Linux des de Zero* desenvolupat, l'extracció de conclusions del projecte en el seu conjunt passa en un primer moment per valorar els objectius marcats a l'inici del treball i deixar una mica apartada la sensació de frustració final.

A l'apartat dos d'aquesta memòria es recullen els objectius d'aquest treball final de carrera. Procedeixo a valorar un a un la seva consecució:

### Objectius Generals

1. Aprendre l'estructura d'un sistema operatiu basat en Linux.

Considero que aquest objectiu ha estat assolit. El grau d'aprenentatge adquirit per la meua part queda reflectit en les explicacions desenvolupades al llarg d'aquesta memòria, doncs en ella es va explicant l'estructura del sistema LDZ que he volgut desenvolupar pas a pas.

2. Crear-ne un sistema basat en Linux des de zero.

Aquest objectiu general no ha estat assolit. Entenc per crear un sistema basat en Linux des de zero un que sigui únic i que es pugui fer servir, en aquest cas el que he anomenat en aquesta memòria LDZ, *Linux Des de Zero*, que prenent com a base el LFS, *Linux From Scratch*, fos un sistema operatiu que es pogués usar en els ordinadors d'una aula d'un institut de secundària. I no ha estat possible perquè no he pogut arrencar l'escriptori

gràfic, i evidentment sense aquest element el LDZ no és usable, no ha arribat a arrencar i per tant no es pot afirmar que hagi estat creat.

### 3. Instal·lar en un ordinador un sistema operatiu basat en Linux des de zero.

Encara que en un primer moment pugui semblar incongruent amb el que he afirmat a l'apartat anterior, crec que aquest objectiu sí que ha estat assolit en bona mesura. He instal·lat la majoria dels paquets que configurarien el sistema operatiu plantejat a l'inici, els que componen pròpiament el *Linux From Scratch* i els de l'escriptori gràfic. Només m'han faltat, per falta de temps, alguns com el LibreOffice, o el Firefox. Tanmateix el problema que m'ha impedit arrencar el LDZ no ha estat per no haver instal·lat els paquets necessaris. És un problema del nucli, probablement per incompatibilitats amb el meu processador de 64 bits. De fet no puc comprovar si la instal·lació de l'escriptori gràfic ha estat satisfactòria o no, però completa sí que ha estat.

## Objectius Concrets

### 1. Minimitzar els requeriments de memòria i CPU del sistema Linux des de zero.

Al menys des del punt de vista del disseny aquest objectiu sí que ha estat assolit. El sistema operatiu que s'ha volgut bastir, en fer ho paquet a paquet, s'ha construït amb un nombre relativament petit de programes. Malgrat els recursos que un escriptori gràfic exigeix, tant en termes de CPU com de memòria, aquests seran els mínims necessaris, o molt propers a aquest mínim, ja que el nombre de programes executant-se concurrentment es limitaran als propis del sistema, i als pocs que l'usuari faci servir.

2. Implementar només el programari necessari per oferir les funcionalitats requerides en la tasca docent d'un institut de secundària: accés a Internet, programari ofimàtic i programari audiovisual.

Aquest darrer objectiu ha estat només parcialment assolit, teòricament, doncs en no poder iniciar-se el LDZ no es pot contrastar empíricament. Tanmateix el programari audiovisual ha estat implementat. Aquest consistia en el programa VLC, que com es pot comprova a l'annex IV, l'incorpora el KDE, l'escriptori gràfic. Pel que fa l'accés a Internet, l'escriptori gràfic inclou per defecte un navegador, el Konqueror, per tant l'accés a Internet a través del navegador és possible. Finalment el programari ofimàtic, en aquest cas el LibreOffice no ha estat instal·lat per manca de temps, cosa que fa que aquest objectiu només s'assoleixi parcialment.

Fetes aquestes consideracions concloc la present memòria amb la valoració final. Després de tot el temps dedicat al desenvolupament d'aquest treball, he arribat a la conclusió, que era massa ambiciós per a les meves possibilitats. Haig de reconèixer, que amb els coneixements adquirits durant la seva realització, ara mateix no el plantejaria de cap de les maneres com ho vaig fer a l'inici. Ha estat d'una gran ingenuïtat per la meva banda creure que implementar un entorn gràfic, així com programes com reproductors de vídeo, ofimàtics, etc, era una tasca assequible. No m'hagués imaginat a l'inici ni de bon tros la càrrega de feina, de temps de compilació, que aquests programes exigeixen. A més a més en l'intent de la seva implementació m'he trobat amb un problema a nivell de nucli que supera en molt les meves capacitats. Hagués estat molt millor haver-me centrat només en un sistema en mode text amb l'intèrpret de comandes. I a partir d'aquí implementar funcionalitats que requerissin un nombre molt menor de paquets, per

exemple un servidor web, un tallafocs, etc.

D'altra banda també haig de reconèixer que l'objectiu final, desenvolupar un sistema operatiu per ser emprat a les aules d'un institut de secundària, no és gaire viable, ni gaire recomanable fent-lo a partir d'un desenvolupament des de zero. El temps que cal dedicar per implementar-ne un de sol és excessiu, a més dels problemes, que com he pogut comprovar, poden sorgir i que al final no facin possible el seu ús, per no parlar dels problemes que pot suposar el seu manteniment. Per tant si es considera un nombre rellevant d'ordinadors on implementar un sistema d'aquesta manera seria totalment absurd dedicar-ne tant de temps i esforços a aquesta tasca quan es poden fer servir distribucions molt més fiables i fàcils d'instal·lar. El guany en temps de CPU o ús de memòria que un sistema creat des de zero pot suposar no compensa ni de bon tros els seus inconvenients.

Tanmateix no vull acabar amb una valoració negativa, doncs l'experiència obtinguda en el desenvolupament d'aquest projecte és totalment positiva. Crec que desenvolupar un sistema seguint els criteris de *Linux From Scratch* és abans de tot una tasca d'aprenentatge, per familiaritzar-se ja en un cert grau de profunditat amb el món gnu-linux, no buscar-ne cap aplicació pràctica, a no ser que es tinguin uns coneixements molt avançats. I en aquest aspecte la realització d'aquest treball ha estat força estimulante i satisfactòria. Partint del sistema base de *Linux From Scratch*, i seguint les indicacions del projecte *Beyond Linux From Scratch* he descobert tot un territori a explorar.

## **ANNEX I: Fitxers de configuració i paquets de programari del sistema Linux mínim**

- **Fitxer `.bash_profile`**

```
exec env -i HOME=$HOME TERM=$TERM PS1='\u:\w\$ ' /bin/bash
```

- **Fitxer `.bashrc`**

```
set +h
umask 022
LFS=/mnt/lfs
LC_ALL=POSIX
LFS_TGT=$(uname -m)-lfs-linux-gnu
PATH=/tools/bin:/bin:/usr/bin
export LFS LC_ALL LFS_TGT PATH
```

- **Paquets de programari**

```
Binutils-2.24 (Primer pas de la instal·lació)
GCC-4.8.2 (Primer pas de la instal·lació)
Linux-3.13.3 (Només les capçaleres de l'API)
Glibc-2.19
Libstdc++-4.8.2
Binutils-2.24 (Segon pas de la instal·lació)
GCC-4.8.2 (Segon pas de la instal·lació)
Tcl-8.6.1
Expect-5.45
DejaGNU-1.5.1
Check-0.9.12
Ncurses-5.9
Bash-4.2
Bzip2-1.0.6
Coreutils-8.22
Diffutils-3.3
File-5.17
Findutils-4.4.2
Gawk-4.1.0
Gettext-0.18.3.2
Grep-2.16
Gzip-1.6
M4-1.4.17
Make-4.0
```

Patch-2.7.1  
Perl-5.18.2  
Sed-4.2.2  
Tar-1.27.1  
Texinfo-5.2  
Util-linux-2.24.1  
Xz-5.0.5

**ANNEX II: Paquets de programari instal·lats del sistema LDZ base**

Linux-3.13.3 (Només les capçaleres de l'API)  
Man-pages-3.59  
Glibc-2.19  
Zlib-1.2.8  
File-5.17  
Binutils-2.24  
GMP-5.1.3  
MPFR-3.1.2  
MPC-1.0.2  
GCC-4.8.2  
Sed-4.2.2  
Bzip2-1.0.6  
Pkg-config-0.28  
Ncurses-5.9  
Shadow-4.1.5.1  
Psmisc-22.20  
Procps-ng-3.3.9  
E2fsprogs-1.42.9  
Coreutils-8.22  
Iana-Etc-2.30  
M4-1.4.17  
Flex-2.5.38  
Bison-3.0.2  
Grep-2.16  
Readline-6.2  
Bash-4.2  
Bc-1.06.95  
Libtool-2.4.2  
GDBM-1.11  
Inetutils-1.9.2  
Perl-5.18.2  
Autoconf-2.69  
Automake-1.14.1  
Diffutils-3.3  
Gawk-4.1.0  
Findutils-4.4.2  
Gettext-0.18.3.2  
Groff-1.22.2  
Xz-5.0.5  
GRUB-2.00  
Less-458  
Gzip-1.6  
IPRoute2-3.12.0  
Kbd-2.0.1  
Kmod-16  
Libpipeline-1.2.6  
Make-4.0  
Patch-2.7.1

Sysklogd-1.5  
Sysvinit-2.88dsf  
Tar-1.27.1  
Texinfo-5.2  
Udev-208 (Procedent del paquet systemd-208)  
Util-linux-2.24.1  
Man-DB-2.6.6  
Vim-7.4



## **ANNEX III: Fitxers de configuració del sistema LDZ base**

- **Fitxer /etc/passwd**

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/dev/null:/bin/false
nobody:x:99:99:Unprivileged User:/dev/null:/bin/false
```

- **Fitxer /etc/group**

```
root:x:0:
bin:x:1:
sys:x:2:
kmem:x:3:
tape:x:4:
tty:x:5:
daemon:x:6:
floppy:x:7:
disk:x:8:
lp:x:9:
dialout:x:10:
audio:x:11:
video:x:12:
utmp:x:13:
usb:x:14:
cdrom:x:15:
mail:x:34:
nogroup:x:99:
```

- **Fitxer /etc/sysconfig/ifconfig.eth0**

```
ONBOOT=
IFACE=eth0
SERVICE=ipv4-static
IP=192.168.1.10
GATEWAY=192.168.1.1
PREFIX=24
BROADCAST=192.168.1.255
```

- **Fitxer /etc/sysconfig/ifconfig.wlan0**

```
ONBOOT=yes
IFACE=wlan0
SERVICE=ipv4-static
IP=192.168.1.11
GATEWAY=192.168.1.1
PREFIX=24
BROADCAST=192.168.1.255
```

- **Fitxer /etc/inittab del sistema *Linux From Scratch***

```
# Begin /etc/inittab
id:3:initdefault:
si::sysinit:/etc/rc.d/init.d/rc S
l0:0:wait:/etc/rc.d/init.d/rc 0
l1:S1:wait:/etc/rc.d/init.d/rc 1
l2:2:wait:/etc/rc.d/init.d/rc 2
l3:3:wait:/etc/rc.d/init.d/rc 3
l4:4:wait:/etc/rc.d/init.d/rc 4
l5:5:wait:/etc/rc.d/init.d/rc 5
l6:6:wait:/etc/rc.d/init.d/rc 6
ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -r now
su:S016:once:/sbin/sulogin
1:2345:respawn:/sbin/agetty
2:2345:respawn:/sbin/agetty
3:2345:respawn:/sbin/agetty
4:2345:respawn:/sbin/agetty
5:2345:respawn:/sbin/agetty
6:2345:respawn:/sbin/agetty
--noclear tty1 9600
tty2 9600
tty3 9600
tty4 9600
tty5 9600
tty6 9600
# End /etc/inittab
```

- **Fitxer /etc/inittab del sistema Linux Des de Zero**

```
# Begin /etc/inittab

id:5:initdefault:

si::sysinit:/etc/rc.d/init.d/rc S

10:0:wait:/etc/rc.d/init.d/rc 0
11:S1:wait:/etc/rc.d/init.d/rc 1
12:2:wait:/etc/rc.d/init.d/rc 2
13:3:wait:/etc/rc.d/init.d/rc 3
14:4:wait:/etc/rc.d/init.d/rc 4
15:5:wait:/etc/rc.d/init.d/rc 5
16:6:wait:/etc/rc.d/init.d/rc 6

ca:12345:ctrlaltdel:/sbin/shutdown -t1 -a -r now

su:S016:once:/sbin/sulogin

1:2345:respawn:/sbin/agetty --noclear tty1 9600
2:2345:respawn:/sbin/agetty tty2 9600
3:2345:respawn:/sbin/agetty tty3 9600
4:2345:respawn:/sbin/agetty tty4 9600
5:2345:respawn:/sbin/agetty tty5 9600
6:2345:respawn:/sbin/agetty tty6 9600

# End /etc/inittab
kd:5:respawn:/opt/kde/bin/kdm
```

- **Fitxer /etc/sysconfig/clock**

```
# Begin /etc/sysconfig/clock
UTC=1
# Set this to any options you might need to give to hwclock,
# such as machine hardware clock type for Alphas.
CLOCKPARAMS=
# End /etc/sysconfig/clock
```

- **Fixer /etc/inputrc**

```
# Begin /etc/inputrc
# Modified by Chris Lynn <roryo@roryo.dynup.net>
# Allow the command prompt to wrap to the next line
set horizontal-scroll-mode Off
# Enable 8bit input
set meta-flag On
set input-meta On
# Turns off 8th bit stripping
set convert-meta Off
# Keep the 8th bit for display
set output-meta On
# none, visible or audible
set bell-style none
# All of the following map the escape sequence of the value
# contained in the 1st argument to the readline specific functions
"\eOd": backward-word
"\eOc": forward-word
# for linux console
"\e[1~": beginning-of-line
"\e[4~": end-of-line
"\e[5~": beginning-of-history
"\e[6~": end-of-history
"\e[3~": delete-char
"\e[2~": quoted-insert
# for xterm
"\eOH": beginning-of-line
"\eOF": end-of-line
# for Konsole
"\e[H": beginning-of-line
"\e[F": end-of-line
# End /etc/inputrc
```

## **ANNEX IV: Paquets de programari emprats en la implementació de l'escriptori gràfic**

Es mostren els paquets principals i les seves dependències. Els paquets marcats amb un asterisc corresponen a paquets que ja s'han detallat anteriorment amb les possibles dependències que puguin tenir.

### **X Window System Environment**

- util-macros-1.18.0
  - (Xorg build environment)
- Xorg Protocol Headers
  - util-macros-1.18.0\*
  - Sudo-1.8.9p5
  - Wget-1.15
    - OpenSSL-1.0.1f
    - GnuTLS-3.2.11
    - Nettle-2.7.1
- libXau-1.0.8
  - Xorg Protocol Headers\*
- libXdmcp-1.1.1
  - Xorg Protocol Headers\*
- xcb-proto-1.10
  - Python-2.7.6
    - expat-2.1.0
    - libffi-3.0.13
  - (Xorg build environment)\*
- libxcb-1.10
  - libXau-1.0.8
  - xcb-proto-1.10
  - libXdmcp-1.1.1
- Xorg Libraries
  - Fontconfig-2.11.0
    - FreeType-2.5.2
    - Which-2.20
    - libpng-1.6.9

- expat-2.1.0\*
- libxml2-2.9.1
  - Python-2.7.6\*
- libxcb-1.10\*
  
- xcb-util-0.3.9
  - libxcb-1.10\*
  
- xcb-util-image-0.3.9
  - xcb-util-0.3.9\*
  
- xcb-util-keysyms-0.3.9
  - libxcb-1.10\*
  
- xcb-util-renderutil-0.3.8
  - libxcb-1.10\*
  
- xcb-util-wm-0.4.0
  - libxcb-1.10\*
  
- MesaLib-10.0.3
  - libxml2-2.9.1\*
  - Xorg Libraries\*
  - libdrm-2.4.52
    - Xorg Libraries\*
  - expat-2.1.0\*
  - elfutils-0.158
  - libvdpau-0.7
    - Xorg Libraries\*
  - LLVM-3.4
  
- xbitmaps-1.1.1
  - util-macros-1.18.0\*
  
- Xorg Applications
  - libpng-1.6.9\*
  - MesaLib-10.0.3\*
  - xbitmaps-1.1.1\*
  - xcb-util-0.3.9\*
  - Linux-PAM-1.1.8
  
- xcursor-themes-1.0.4
  - Xorg Applications\*
  
- Xorg Fonts
  - xcursor-themes-1.0.4\*

- XKeyboardConfig-2.11
  - Intltool-0.50.2
    - XML::Parser-2.41
      - expat-2.1.0\*
  - Xorg Libraries\*
- Xorg-Server-1.15.0
  - OpenSSL-1.0.1f\*
  - Nettle-2.7.1\*
  - libgcrypt-1.6.1
    - libgpg-error-1.12
  - Pixman-0.32.4
  - Xorg Fonts\*
  - xkeyboard-config-2.11\*
- Xorg Drivers
  - Glamor EGL-0.6.0
  - Xorg Evdev Driver-2.8.2
    - Xorg-Server-1.15.0\*
    - mtdev-1.1.4
  - Xorg ATI Driver-7.3.0
    - Xorg-Server-1.15.0\*
    - Glamor EGL-0.6.0\*
- twm-1.0.8
  - Xorg-Server-1.15.0\*
- xterm-301
  - Xorg Applications\*
- xclock-1.0.7
  - Xorg Libraries\*
- xinit-1.3.3
  - twm-1.0.8\*
  - xclock-1.0.7\*
  - xterm-301\*

## KDE

- Automoc4-0.9.88
  - CMake-2.8.12.2
    - cURL-7.35.0
    - libarchive-3.1.2
    - expat-2.1.0\*
  - Qt-4.8.5
    - Xorg Libraries\*
    - alsa-lib-1.0.27.2
    - MesaLib-10.0.3\*
    - Certificate Authority Certificates
    - D-Bus-1.8.0
      - expat-2.1.0\*
      - Xorg Libraries\*
    - GLib-2.38.2
      - libffi-3.0.13\*
      - Python-2.7.6\*
      - PCRE-8.34
    - desktop-file-utils-0.22
      - GLib-2.38.2\*
    - ICU-52.1 (unicode support)
    - libjpeg-turbo-1.3.0
      - NASM-2.11
    - libmng-2.0.2
      - libjpeg-turbo-1.3.0\*
      - Little CMS-2.5
    - libpng-1.6.9\*
    - LibTIFF-4.0.3
    - OpenSSL-1.0.1f\*
    - SQLite-3.8.3.1
- Phonon-4.7.1
  - automoc4-0.9.88
  - GLib-2.38.2\*
- Phonon-backend-gstreamer-4.7.1
  - phonon-4.7.1\*
  - GStreamer-0.10.36
    - GLib-2.38.2\*
    - libxml2-2.9.1\*
  - gst-plugins-base-0.10.36
    - GStreamer-0.10.36\*
    - Pango-1.36.2
      - Cairo-1.12.16



- libpng-1.6.9\*
- GLib-2.38.2\*
- Pixman-0.32.4\*
- Fontconfig-2.11.0\*
- Xorg Libraries\*
- Harfbuzz-0.9.26
  - GLib-2.38.2\*
  - ICU-52.1\*
  - FreeType-2.5.2\*
- Xorg Libraries\*
- alsa-lib-1.0.27.2\*
- libogg-1.3.1\*
- libtheora-1.1.1
  - libogg-1.3.1\*
  - libvorbis-1.3.4
    - libogg-1.3.1\*
- libvorbis-1.3.4\*
- udev-extras
  - GLib-2.38.2\*
  - Gperf-3.0.4
  - gobject-introspection-1.38.0
- Xorg Libraries\*
- gst-plugins-good-0.10.31
  - gst-plugins-base-0.10.36\*
  - Cairo-1.12.16\*
  - FLAC-1.3.0
  - libjpeg-turbo-1.3.0\*
  - libpng-1.6.9\*
  - Xorg Libraries\*
- gst-plugins-bad-0.10.23
  - gst-plugins-base-0.10.36\*
  - FAAC-1.28
  - libpng-1.6.9\*
  - libvpx-1.3.0
    - yasm-1.2.0
    - Which-2.20\*
  - OpenSSL-1.0.1f\*
  - Xvid-1.3.2
- gst-plugins-ugly-0.10.19
  - gst-plugins-base-0.10.36\*
  - LAME-3.99.5
  - libdvdnav-4.2.1
    - libdvdread-4.2.1
  - libdvdread-4.2.1\*

- Phonon-backend-vlc-0.7.1
  - phonon-4.7.1\*
  - VLC-2.1.3
    - Lua-5.2.3
    - libmad-0.15.1b
    - FFmpeg-2.1.3
      - yasm-1.2.0\*
      - libass-0.10.2
        - FreeType-2.5.2\*
        - FriBidi-0.19.6
      - fdk-aac-0.1.3
      - LAME-3.99.5\*
      - Opus-1.1
      - libtheora-1.1.1\*
      - libvorbis-1.3.4\*
      - libvpx-1.3.0\*
      - x264-20140115-2245
        - yasm-1.2.0\*
      - X Window System\*
      - alsa-lib-1.0.27.2\*
      - SDL-1.2.15
      - libva-1.2.1
        - MesaLib-10.0.3\*
      - libvdpau-0.7\*
    - liba52-0.7.4
    - X Window System\*
    - alsa-lib-1.0.27.2\*
    - libgcrypt-1.6.1\*
- Akonadi-1.11.0
  - shared-mime-info-1.2
    - GLib-2.38.2\*
    - Intltool-0.50.2\*
    - libxml2-2.9.1\*
  - Boost-1.55.0
  - Soprano-2.9.4
    - CMake-2.8.12.2\*
    - Qt-4.8.5\*
  - Redland-1.0.17
    - Rasqal-0.9.31
      - Raptor-2.0.13
        - cURL-7.35.0\*
        - libxslt-1.1.28
          - libxml2-2.9.1\*
  - CLucene-2.3.3.4

- CMake-2.8.12.2\*
- Boost-1.55.0\*
- D-Bus-1.8.0\*
- libiodbc-3.52.8
  - GTK+-2.24.22
    - ATK-2.10.0
      - GLib-2.38.2\*
    - gdk-pixbuf-2.30.4
      - GLib-2.38.2\*
      - libjpeg-turbo-1.3.0\*
      - libpng-1.6.9\*
      - LibTIFF-4.0.3\*
      - Xorg Libraries\*
    - Pango-1.36.2\*
    - hicolor-icon-theme-0.12
- Virtuoso-6.1.8
  - libiodbc-3.52.8\*
  - libxml2-2.9.1\*
  - OpenSSL-1.0.1f\*
  - OpenLDAP-2.4.39
    - Berkeley DB-6.0.20
    - Cyrus SASL-2.1.26
      - OpenSSL-1.0.1f\*
      - Berkeley DB-6.0.20\*
    - OpenSSL-1.0.1f\*
- SQLite-3.8.3.1\*
  
- Attica-0.4.2
  - CMake-2.8.12.2\*
  - Qt-4.8.5\*
  
- QImageblitz-0.0.6
  - Qt-4.8.5\*
  - CMake-2.8.12.2\*
  
- Shared-desktop-ontologies-0.11.0
  - CMake-2.8.12.2\*
  
- Polkit-Qt-0.103.0
  - automoc4-0.9.88\*
  - Polkit-0.112
    - GLib-2.38.2\*
    - Intltool-0.50.2\*
    - JS-17.0.0
      - libffi-3.0.13\*

- NSPR-4.10.3
- Python-2.7.6\*
- Zip-3.0
  
- Oxygen-icons-4.12.2
  - CMake-2.8.12.2\*
  
- Kdelibs-4.12.2
  - phonon-4.7.1\*
  - Attica-0.4.2\*
  - Soprano-2.9.4\*
  - strigi-0.7.8
    - CMake-2.8.12.2\*
    - expat-2.1.0\* o libxml2-2.9.1\*
    - D-Bus-1.8.0\*
    - Qt-4.8.5\*
  - qca-2.0.3
    - Qt-4.8.5\*
    - Which-2.20\*
  - libdbusmenu-qt-0.9.2
    - Qt-4.8.5\*
  - docbook-xml-4.5
    - libxml2-2.9.1\*
    - UnZip-6.0
  - docbook-xsl-1.78.1
    - libxml2-2.9.1\*
  - shared-desktop-ontologies-0.11.0
    - CMake-2.8.12.2\*
  - shared-mime-info-1.2\*
  - polkit-qt-0.103.0\*
  - libpng-1.6.9\*
  - libjpeg-turbo-1.3.0\*
  - giflib-5.0.5
  - UPower-0.9.23
    - dbus-glib-0.102
      - D-Bus-1.8.0\*
      - expat-2.1.0\*
      - GLib-2.38.2\*
    - Intltool-0.50.2\*
    - libusb-1.0.18
    - Polkit-0.112\*
    - udev-extras\*
    - pm-utils-1.4.1
  - UDisks-1.0.4
    - dbus-glib-0.102\*

- libatasmart-0.19
  - LVM2-2.02.105
  - parted-3.1
    - LVM2-2.02.105\*
  - Polkit-0.112\*
  - sg3\_utils-1.37
  - udev-extras\*
- 
- Polkit-kde-agent-0.99.0
    - polkit-qt-0.103.0\*
    - kdelibs-4.12.2\*
- 
- Nepomuk-core-4.12.2
    - kdelibs-4.12.2\*
    - Poppler-0.24.5
      - Fontconfig-2.11.0\*
      - Cairo-1.12.16\*
      - libjpeg-turbo-1.3.0\*
      - libpng-1.6.9\*
    - taglib-1.9.1
      - CMake-2.8.12.2\*
    - Exiv2-0.24
      - expat-2.1.0\*
    - FFmpeg-2.1.3\*
- 
- Nepomuk-widgets-4.12.2
    - nepomuk-core-4.12.2\*
      - kdelibs-4.12.2\*
      - Poppler-0.24.5\*
      - taglib-1.9.1\*
      - Exiv2-0.24\*
      - FFmpeg-2.1.3\*
- 
- Kdepimlibs-4.12.2
    - nepomuk-core-4.12.2\*
    - libxslt-1.1.28\*
    - GPGME-1.4.3
      - Libassuan-2.1.1
        - libgpg-error-1.12\*
    - libical-1.0
      - CMake-2.8.12.2\*
    - Akonadi-1.11.0\*
    - Cyrus SASL-2.1.26\*
      - OpenSSL-1.0.1f\*
    - Berkeley DB-6.0.20\*

- Boost-1.55.0\*
- QJson-0.8.1
  - Qt-4.8.5\*
  - CMake-2.8.12.2\*
- OpenLDAP-2.4.39\*
  
- Kactivities-4.12.2
  - kdelibs-4.12.2\*
  - nepomuk-core-4.12.2\*
  
- Kde-runtime-4.12.2
  - kdelibs-4.12.2\*
  - libtirpc-0.2.4
  - kactivities-4.12.2\*
  - kdeplibs-4.12.2\*
  - alsa-lib-1.0.27.2\*
  - libjpeg-turbo-1.3.0\*
  - Exiv2-0.24\*
  
- Kde-baseapps-4.12.2
  - kdelibs-4.12.2\*
  - kactivities-4.12.2\*
  - nepomuk-widgets-4.12.2\*
  
- Kde-base-artwork-4.12.2
  - kdelibs-4.12.2\*
  
- Kde-workspace-4.11.6
  - kactivities-4.12.2\*
  - qimageblitz-0.0.6\*
  - xcb-util-image-0.3.9\*
  - xcb-util-renderutil-0.3.8\*
  - xcb-util-keysyms-0.3.9\*
  - xcb-util-wm-0.4.0\*
  - kdeplibs-4.12.2\*
  - nepomuk-core-4.12.2\*
  - Boost-1.55.0\*
  - FreeType-2.5.2\*
  - pciutils-3.2.1
  - ConsoleKit-0.4.6
    - acl-2.2.52
      - attr-2.4.47
    - dbus-glib-0.102\*
    - Xorg Libraries\*
    - Linux-PAM-1.1.8\*
    - Polkit-0.112\*