

#####

#####. #####, ### ##### ## ##—

#####.

- ##### ## ### ##### ##### ##### # #####
(##, ##, ##). ## ##### ## ##### ## make
buildworld.

- ## ##### ##### ## ##### ##### ## ## ##### ## ## ##### ## ##—
#####.
- ## ##### ##### ## ##### ##### #####.
- ## ##### ##### #####, ##### ##### ##### #####.
- ## ##### ##### ##### ##### # ##### ##### ## ##### ##### ##
#####
- ### ## ##### ##### ##### ##### ##### ##### ##### ## ##### ## ##
#####

```
#####  #####  #####  #####  #####  #####  #####. ## #####  ## #####:
#####  ## #####  #####  #####  #####  #####  ##  #####  ##  ## #####
```


#:

- ## ##### #### # ##### ## ##### # ### ##### #####.
- ## ##### #### ##### ##### ##### ## #####.
- ##### ## ## ## ## #####(8) # ## ##### ## ##### ## #####.
- ## ## ##### # ##### ## ##### # ## ## ##### ## ##### ##### ## #####—
#####.
- ##### ##### ## ##### ## ##### ## ##### ## ## ##### ##
##(1)
- #####, ##### ## ##### ## ##### # ## ##### (##### ##
#####) ##### ## #####, ## ##### # #####
#####.

4. ##### #: ##### ##

```
## ## ## ##### # ##### ## ## ##### ## ## fase_1.sh. ##### ## ##### ## ##-  
##### ##### ## ## ##### ## ## ##### ##### ## #####. ## #####-  
##### ##### ## ##### ##### ## ## ##### ##### #####  
##### ## #####. ## ##### # ##### ##:
```

- ```

● ##### ## #####.

##-
#####. ## ##### ##### ##### ## ##### ## ##-
/, /usr # /var ## /tmp ##### ##### /var/tmp.
/home (## ##### ## ## ##-
#####), /home/ncvs (##### ## #####, /usr/ports (## ##### ##
#####), /src (##### ## #####) /share (#####
#####, ## #####)

```

- ```
● #####.
```
- ```

####.# ## ##### ## ##### ## #####/### ##### ##
/etc/passwd. ## ##### ##### ##
(## ### ## #####),#### ## ## #####
#####.
```

?### ## ## ##### ##### ##### ## ## ##### ## ## ##### ##? : ## #####  
# ## ## ##### ## ##### ##### ## ##### ## ##### # ## #####—  
###: ##### ## ##### ## ## ## ##### ## ##### ##### ##### ##  
##### ##### ## ## ##### ##### ## ##### ## ##### #####  
##### ## ## ##### ## ## ##### ## ## ##### ## #####  
#####. ## ## ##### ## ##### ##### ##### ##### ##### #####

```
#####
```

```
(#####) #####

#####. #####
#####5.
```

```
fase_1.sh #####
make installworld installkernel,
```

- #####
- ##### make buildworld
- ##### KERNCONF= nombre\_de\_su\_kernel

```
fase_1.sh #####

/usr/src, ##### mergemaster #####
#####. #####
#####. (#####: #####, #####,
#####). ##### mergemaster #####
original (#####
#####: #####, ##### -STABLE # ##### -CURRENT #####
#####). #####
mergemaster #####
/usr/src # #####.
```

```
fase_1.sh #####
(#####) ##### set -e, #####
#####. #####
fase_1.sh.
```

```
fase_1.sh ##### mergemaster. #####
mergemaster
```

```
*** Comparison complete
```

```
Do you wish to delete what is left of -/var/tmp/temproot.fase1? [no] no
```

```
##
```

```
*** Comparación completada
```

```
?Quiere borrar el contenido de -/var/tmp/temproot.fase1? [no] no
```

```
#####, ##### no # #####
/var/tmp/temproot.fase1
(# #####).
```

```
#####(1) #
#####(1):
```

##### #: ##### ## #####

\*\*\* You chose the automatic install option for files that did not exist on your system. The following were installed for you:  
-rootnuevo/etc/defaults/rc.conf  
-...  
-rootnuevo/COPYRIGHT

(END)

## #####

\*\*\* Ha elegido la opción de instalar automáticamente los ficheros que no existen en su sistema. Han sido instalados los siguientes:  
-rootnuevo/etc/defaults/rc.conf  
-...  
-rootnuevo/COPYRIGHT

##### # ##### ## #####. ##### ## ## ##### login.conf:

\*\*\* You installed a login.conf file, so make sure that you run  
-'/usr/bin/cap\_mkdb -/newroot/etc/login.conf'  
to rebuild your login.conf database

Would you like to run it now? y or n [n]

## #####

\*\*\* Ha instalado un fichero login.conf así que asegúrese de ejecutar -'/usr/bin/cap\_mkdb -/rootnuevo/etc/login.conf' para reconstruir la base de datos de login.conf

-?Quiere ejecutarlo ahora mismo? (s)i o (n)o [n]

## ##### ## ##### ##### ## ##### ##\_####(1) ## #####  
## #####.

#### ## ## fase\_1.sh ##### ## ## ##### ##### ## ## ##-  
##### ## ##### ## ## #####.

#### ## ## fase\_1.sh ## ## ##, ## ## ##### ## #####, ## ##-  
##### ## ##### 1, 2, 5 # 6.



#####

### #####, ##### ## ##### ## ## ##  
### #####(8). ## ## ## ## ##  
##### ## ## ## ##  
##### ## ## ## ##

#####

```

##, ## ##### /dev/da3s1a, /dev/vinum/var_a # /dev/vinum/usr_a.

#####
```

```
#!/bin/sh

fase_1.sh -- FreeBSD From Scratch, Primera Fase: Instalación del Sistema.
Uso: ./fase_1.sh

$FreeBSD: head/es_ES.ISO8859-1/articles/fbsd-from-scratch/fase_1.sh 38826 2012-05-17 -
19:12:14Z hrs $

set --x --e
PATH=/bin:/usr/bin:/sbin:/usr/sbin

Requisitos:

a) Haber completado sin errores -"make buildworld" y -"make buildkernel"
b) Particiones sin usar (al menos una para el sistema de ficheros raíz, -
probablemente más para los nuevos -/usr y -/var, a gusto de cada uno.)

El punto montaje de la raíz bajo la que va usted a crear el sistema nuevo.
Sólo va a usarse como punto de montaje; que no se usará espacio en él -
puesto que todos los ficheros serán depositados en el o los sistemas -
de ficheros que están efectivamente montados.
DESTDIR=/rootnuevo
SRC=/usr/src # Aquí está su árbol de fuentes.

----- #
Primer Paso: Creación de un árbol de directorios vacío bajo $DESTDIR. -
----- #

step_one () {
 # El nuevo raíz del sistema de ficheros. Obligatorio.
 # Cambie los nombres de dispositivo (DEV_*) para hacerlos acordes con -
 # sus necesidades o el -"script" le
 # estallará en la cara.
 DEV_ROOT=/dev/da3s1a
 mkdir --p ${DESTDIR}
 newfs ${DEV_ROOT}
 tune2fs --n enable ${DEV_ROOT}
 mount --o noatime ${DEV_ROOT} ${DESTDIR}

 # Sistemas de ficheros extra y sus correspondientes puntos de montaje. -
 # Opcional.
 DEV_VAR=/dev/vinum/var_a
 newfs ${DEV_VAR}
 tune2fs --n enable ${DEV_VAR}
 mkdir --m 755 ${DESTDIR}/var
```

```

mount --o noatime ${DEV_VAR} ${DESTDIR}/var

DEV_USR=/dev/vinum/usr_a
newfs ${DEV_USR}
tunefs --n enable ${DEV_USR}
mkdir --m 755 ${DESTDIR}/usr
mount --o noatime ${DEV_USR} ${DESTDIR}/usr

mkdir --m 755 --p ${DESTDIR}/usr/ports
mount -/dev/vinum/ports ${DESTDIR}/usr/ports

Aquí crearemos los demás directorios. Obligatorio.
cd ${SRC}/etc; make distrib-dirs DESTDIR=${DESTDIR}
Personalmente me gusta enlazar tmp a var/tmp. Opcional.
cd ${DESTDIR}; rmdir tmp; ln --s var/tmp
}

Segundo Paso: Poblamos el árbol de directorios -/etc que está vacío aún y -
ubicamos unos cuantos ficheros en -/.

step_two () {
Añada o borre de ésta lista según su criterio. La mayoría son obligatorios.
for f in \
 -/.profile \
 -/etc/group \
 -/etc/hosts \
 -/etc/inetd.conf \
 -/etc/ipfw.conf \
 -/etc/make.conf \
 -/etc/master.passwd \
 -/etc/nsswitch.conf \
 -/etc/ntp.conf \
 -/etc/printcap \
 -/etc/profile \
 -/etc/rc.conf \
 -/etc/resolv.conf \
 -/etc/start_if.xl0 \
 -/etc/ttys \
 -/etc/ppp/* \
 -/etc/mail/aliases \
 -/etc/mail/aliases.db \
 -/etc/mail/hal9000.mc \
 -/etc/mail/service.switch \
 -/etc/ssh/*key* \
 -/etc/ssh/*_config \
 -/etc/X11/XF86Config-4 \
 -/boot/splash.bmp \
 -/boot/loader.conf \
 -/boot/device.hints -; do
 cp --p ${f} ${DESTDIR}${f}
done

```



#####

```
Borre el temproot que haya creado mergemasger. Si lo ha creado.
TEMPROOT=/var/tmp/temproot.fase1
if test -d ${TEMPROOT}; then
 chflags --R 0 ${TEMPROOT}
 rm -rf ${TEMPROOT}
fi
mergemaster --i --m ${SRC}/etc --t ${TEMPROOT} --D ${DESTDIR}
cap_mkdb ${DESTDIR}/etc/login.conf
pwd_mkdb --d ${DESTDIR}/etc --p ${DESTDIR}/etc/master.passwd

Mergemaster no crea ficheros vacíos por ejemplo en /var/log. Lo haremos -
aquí pero sin sobrescribir (y destruir) ficheros copiados en el bucle
de más arriba.
cd ${TEMPROOT}
find - . --type f -| sed -s,^/,,' -|
while read f; do
 if test -r ${DESTDIR}/${f}; then
 echo -"${DESTDIR}/${f} ya existe; no copiado"
 else
 echo -"Creando ${DESTDIR}/${f} vacío"
 cp --p ${f} ${DESTDIR}/${f}
 fi
done
chflags --R 0 ${TEMPROOT}
rm -rf ${TEMPROOT}
}

Tercer Paso: Instalando el mundo (install world).

step_three () {
 cd ${SRC}
 make installworld DESTDIR=${DESTDIR}
}

Cuarto Paso: Instalación del kernel y los módulos.

step_four () {
 cd ${SRC}
 # loader.conf y device.hints son necesarios para installkernel.
 # Si en el segundo paso no los ha copiado hágalo tal y como se muestra en -
 # las dos líneas siguientes.
 # cp sys/boot/forth/loader.conf ${DESTDIR}/boot/defaults
 # cp sys/i386/conf/GENERIC.hints ${DESTDIR}/boot/device.hints
 make installkernel DESTDIR=${DESTDIR} KERNCONF=NOMBRE_DE_SU_KERNEL
}

Quinto Paso: Instalación y modificación de algunos ficheros clave.

```

```

step_five () {
Creamos /etc/fstab; obligatorio. Modifíquelo para que coincida con sus -
dispositivos.
cat <<EOF >${DESTDIR}/etc/fstab
Device Mountpoint FSType Options Dump Pass#
/dev/da3s1b none swap sw 0 0
/dev/da4s2b none swap sw 0 0
/dev/da3s1a -/ ufs rw 1 1
/dev/da1s2a -/src ufs rw 0 2
/dev/da2s2f -/share ufs rw 0 2
/dev/vinum/var_a -/var ufs rw 0 2
/dev/vinum/usr_a -/usr ufs rw 0 2
/dev/vinum/home -/home ufs rw 0 2
/dev/vinum/ncvs -/home/ncvs ufs rw,noatime 0 2
/dev/vinum/ports -/usr/ports ufs rw,noatime 0 2
#
/dev/cd0 -/dvd cd9660 ro,noauto 0 0
/dev/cd1 -/cdrom cd9660 ro,noauto 0 0
proc -/proc procfs rw 0 0
EOF
Más directorios; opcional.
mkdir --m 755 --p ${DESTDIR}/src; chown root:wheel ${DESTDIR}/src
mkdir --m 755 --p ${DESTDIR}/share; chown root:wheel ${DESTDIR}/share
mkdir --m 755 --p ${DESTDIR}/dvd; chown root:wheel ${DESTDIR}/dvd
mkdir --m 755 --p ${DESTDIR}/home; chown root:wheel ${DESTDIR}/home
mkdir --m 755 --p ${DESTDIR}/usr/ports; chown root:wheel ${DESTDIR}/usr/ports
Configuración de la zona horaria; no es obligatorio pero casi.
cp ${DESTDIR}/usr/share/zoneinfo/Antarctica/South_Pole ${DESTDIR}/etc/localtime
if test -r /etc/wall_cmos_clock; then
 cp --p /etc/wall_cmos_clock ${DESTDIR}/etc/wall_cmos_clock
fi
}

Sexto Paso: Lo que considero importante tener cuando accedo a un sistema
nuevo por primera vez.
NOTA: No instale demasiados binarios en éste paso. Con el sistema viejo -
en funcionamiento y los nuevos binarios y ficheros de cabecera instalados -
es casi seguro tener problemas de bootstrap. Los -"ports" deberían compilarse -
después de haber arrancado el nuevo sistema.

step_six () {
chroot ${DESTDIR} sh --c -"cd /usr/ports/shells/zsh; make clean install clean"
chroot ${DESTDIR} sh --c -"cd /etc/mail; make install" # configuración -
de sendmail

Si no enlazamos simbólicamente compat los ficheros de linux_base -
irán a parar al sistema de ficheros raíz.
cd ${DESTDIR}; mkdir --m 755 usr/compat
chown root:wheel usr/compat; ln --s usr/compat

```

#####

```
mkdir --m 755 usr/compat/linux
mkdir --m 755 boot/grub

Creación de los directorios -"spool" para las impresoras que hay en -
mi -/etc/printcap
cd ${DESTDIR}/var/spool/output/lpd; mkdir --p as od ev te lp da
touch ${DESTDIR}/var/log/lpd-errs

Más ficheros que quiero heredar del sistema antiguo.
for f in \
 -/var/cron/tabs/root \
 -/var/mail/* \
 -/boot/grub/*; do
 cp --p ${f} ${DESTDIR}/${f}
done

Si no tiene /home en una partición compartida es un buen momento para -
copiarlo al sitio correcto.
mkdir --p ${DESTDIR}/home
cd /home; tar cf -- -. -| (cd ${DESTDIR}/home; tar xpvf --)

Como novedad en FreeBSD 5.x perl está en /usr/local/bin pero la -
mayoría de -"scripts" esperan encontrarlo en /usr/bin/perl y así lo -
reflejan en su primera línea; use un enlace simbólico para que funcionen.
cd ${DESTDIR}/usr/bin; ln --s ../local/bin/perl
cd ${DESTDIR}/usr; rmdir src; ln --s ../src/current src
}

do_steps () {
 step_one
 step_two
 step_three
 step_four
 step_five
 step_six
}

do_steps 2>&1 -| tee fase_1.log

EOF $RCSfile: fase_1.sh,v $ vim: tabstop=2:expandtab:
```

##### [fase\\_1.sh](#).

## #####

- #####
- #####
- #####
- ##### /etc/ttys # inetd.

```
(### ##### #-#####)#####.#####
fase_2.sh#####
pkg_add.#####
#####.#####
#####.#####
```



```
fi
set --- $args
DRYRUN=
for i; do
 case -"$i" in
 -n) DRYRUN=yes;;
 --) break;;
 *) usage;;
 esac
done

cat << EOF -|
lang perl5
security sudo
x11-servers XFree86-4-Server
x11 wrapper
x11 XFree86-4-libraries
x11 XFree86-4-clients
x11-fonts XFree86-4-font75dpi
x11-fonts XFree86-4-font100dpi
x11-fonts XFree86-4-fontScalable
x11-fonts urwfonts
x11-fonts webfonts
x11-toolkits open-motif
x11 rxvt
x11-wm ctwm
security openssh-askpass
astro xplanet
astro setiathome make BATCH=yes install
astro xephem
editors vim
print ghostscript-gnu make A4=yes BATCH=yes install
print a2ps-a4
print psutils-a4
print gv
print acroread5
print transfig
archivers zip
archivers unzip
java linux-sun-jdk13 yes -| make install
java jdk13
www apache2
www weblint
www amaya
www mozilla make WITHOUT_MAILNEWS=yes WITHOUT_CHATZILLA=yes install
www netscape48-navigator
www checkbot
www privoxy
graphics xfig
graphics xv
graphics fxtv
lang expect
news tin
```

#####

```
net freebsd-uucp
net cvsup-without-gui
net pathchar make NO_CHECKSUM=yes install
ftp wget
ftp ncftp3
textproc ispell
german ispell-neu
german ispell-alt
textproc docproj make JADETEX=yes HAVE_MOTIF=yes install
sysutils samefile
sysutils pstree
sysutils mkisofs
sysutils cdrtools
sysutils grub
devel ddd
devel ctags
devel ElectricFence
mail procmail make BATCH=yes install
mail metamail
mail mutt
mail spamoracle
emulators mtools
sysutils portupgrade
news inn-stable CONFIGURE_ARGS="--enable-uucp-rnews ---enable-setgid-inews" make install
misc figlet-fonts
textproc gmat
EOF
while read CATEGORY NAME CMD; do
 case -"${CATEGORY}" in
 \#*) continue;;
 *) continue;;
 esac
 DIR="${PORTS}/${CATEGORY}/${NAME}"
 if -! test --d -"${DIR}"; then
 echo -"${DIR} no existe --- ignorado"
 continue
 fi
 cd ${DIR}
 PKGNAME=`make --V PKGNAME`
 if test --d -"${DBDIR}/${PKGNAME}"; then
 echo -"${CATEGORY}/${NAME} ya instalado como ${PKGNAME}"
 continue
 fi
 LOG="${LOGDIR}/${CATEGORY}+${NAME}"
 echo -"==> Instalando ${CATEGORY}/${NAME}; registrando instalación en ${LOG}"
 test --n -"${CMD}" -| CMD="make install"
 if test --n -"${DRYRUN}"; then
 echo -"${CMD}"
 continue
 fi
 date -"+++++++ %v %T ++++++" > ${LOG}
 echo -"CMD: ${CMD}" >> ${LOG}
 (
```

##### ####

```
make clean
eval -"${CMD}"
make clean # Descoméntelo si no le sobra espacio bajo ${PORTS}.
) 2>&1 -| tee --a ${LOG}
done

Instalamos StarOffice como -"package", previamente creado en el sistema -
antiguo mediante -"make package" porque el -"port" usa una instalación -
interactiva en X11.
#pkg_add ${PORTS}/editors/staroffice52/staroffice-*.tbz

EOF $RCSfile: fase_2.sh,v $ vim: tabstop=4:
```

##### fase\_2.sh.

## 6. #####

```
#####
#####, ##### ## ##### ##### ## ## ## #####. ## ## ##-
##, ##### ## ##### ##### ## ## #####-
###. ##### ##### ## ## ##### fase_2.sh ##### ## ## ## ##-
##
##
#####.
```

```
Makefile ##### #-
#####:
```

```
make -f fase_3.mk -
nombre_del_port
```

```
fase_2.sh ##### ## ## ##### ## ## ##### ## ## fase_3.mk
##, ##### ## ##### ## ##
#####.
```

```
fase_3.mk -- FreeBSD From Scratch, Tercera Fase: Ajustes Personalizados -
de la Configuración de -
los Ports.
Uso: make -f fase_3.mk all (configurar todos los ports)
or make -f fase_3.mk target (configurar sólo target)
#
Es una buena idea asegurarse de que ningún -"target" pueda hacerse más de -
una vez sin efectos no deseados. -
#
$FreeBSD: head/es_ES.ISO8859-1/articles/fbsd-from-scratch/fase_3.mk 39632 2012-10-01 -
11:56:00Z gabor $

.POSIX:

message:
```



#####

```
@echo -"Por favor, use uno de los siguientes \"targets\":"
@echo -"config_apache"
@echo -"config_inn"
@echo -"config_javaplugin"
@echo -"config_privoxy"
@echo -"config_setiathome"
@echo -"config_sgml"
@echo -"config_sudo"
@echo -"config_TeX"
@echo -"config_tin"
@echo -"config_uucp"
@echo -"all --- todos los anteriores"
```

```
all: config_apache \
config_inn \
config_javaplugin \
config_privoxy \
config_setiathome \
config_sgml \
config_sudo \
config_TeX \
config_tin \
config_uucp
```

config\_apache:

# 1. Modificación httpd.conf.

```
perl -pi \
-e -s/#ServerName new.host.name/ServerName hal9000.s.shuttle.de/; \
-e -s/^ServerAdmin.*/ServerAdmin schweikh\@schweikhardt.net/; \
-e -s,/usr/local/www/cgi-bin/,/home/opt/www/cgi-bin/;; \
/usr/local/etc/apache2/httpd.conf
```

# 2. Restituír los enlaces simbólicos a los sitios web.

```
cd -/usr/local/www/data; \
ln --fs -/home/schweikh/prj/homepage schweikhardt.net; \
ln --fs -/home/opt/www/test -.
```

config\_inn:

```
pw usermod --n news --d -/usr/local/news --s -/bin/sh
Facilitar al sistema de news su configuración inicial.
```

```
cd -/home/root/setup; \
```

```
install --C --o news --g news --m 664 active newsgroups -/usr/local/news/db
```

# El innd.sh que viene con el -"port" falla: busca -

# history.pag, fichero que no existe. -

```
cd -/home/root/setup; \
```

```
install --C --o root --g wheel --m 555 innd.sh -/usr/local/etc/rc.d
```

# Configuración del método de almacenamiento.

```
cd -/home/root/setup; \
printf -"%s\n%s\n%s\n%s\n" \
"method tradspool {" \
" newsgroups: *" \
" class: 0" \
"}" \
>storage.conf;
```

```

install --C --o news --g news --m 664 storage.conf -/usr/local/news/etc
Configuración de newsfeeds.
printf -"%s\n%s\n" \
"ME:~:" \
"shuttle/news2.shuttle.de:!junk,!control:B32768/512,Tf,Wfb:" \
>/usr/local/news/etc/newsfeeds
Configuración de inn.conf
perl --pi \
-e 's/^(organization:s*).*/$1 An Open Pod Bay Door;' \
-e 's/^(pathhost:s*).*/$1 hal9000.schweikhardt.net;' \
-e 's/^(server:).*/$1 localhost;' \
-e 's/^(domain:).*/$1 schweikhardt.net;' \
-e 's/^(fromhost:).*/$1 schweikhardt.net;' \
-e 's,^(moderatoremailer:).*,$1 \%s\@moderators.isc.org,;' \
-e 's,/usr/local/news/spool,/share/news/spool,;' \
/usr/local/news/etc/inn.conf

config_javaplugin:
cd -/usr/local/lib/netscape-linux/plugins; \
if -! test --h javaplugin.so; then \
ln --s -./../linux-sun-jdk1.3.1/jre/plugin/i386/ns4/javaplugin.so; \
fi; \
ls --l javaplugin.so

config_privoxy:
install --C --o root --g wheel --m 644 config -/usr/local/etc/privoxy

config_setiathome:
perl --pi \
-e 's,^.*seti_wrkdir.*#,seti_wrkdir=/home/nobody/setiathome #,;' \
/usr/local/etc/rc.setiathome.conf

config_sgml:
cp --p -/usr/local/share/gmat/sgml/ISO_8879-1986/entities/* \
-/usr/local/share/xml/docbook/4.1

config_sudo:
if -! grep --q schweikh -/usr/local/etc/sudoers; then \
echo 'schweikh ALL = (ALL) NOPASSWD: ALL' >> -/usr/local/etc/sudoers; \
fi

config_TeX:
En el fichero textproc/docproj se especifica: para generar el FreeBSD -
Handbook con JadeTex cambie los siguientes parámetros a los valores que se -
indican a continuación:
#
perl --pi \
-e 's/^\% original texmf.cnf/\% texmf.cnf;' \
-e 's/^(hash_extra\s*=).*/$1 60000/;' \
-e 's/^(pool_size\s*=).*/$1 1000000/;' \
-e 's/^(max_strings\s*=).*/$1 70000/;' \
-e 's/^(save_size\s*=).*/$1 10000/;' \
/usr/local/share/texmf/web2c/texmf.cnf

```

```
#####
```

```
config_tin:
Dirigimos tin a nuestros archivos.
printf -"%s\n%s\n%s\n" \
"activefile=/usr/local/news/db/active" \
"newsgroupsfile=/usr/local/news/db/newsgroups" \
"spooldir=/share/news/spool/articles" \
>/usr/local/etc/tin.defaults

config_uucp:
UUCP cuenta con encontrar -/usr/bin/rnews.
cd -/usr/bin; ln --fs ../local/news/bin/rnews -.
Configuración de UUCP.
echo nodename js2015 > -/usr/local/etc/uucp/config
echo shuttle js2015 `cat uucp` > -/usr/local/etc/uucp/call
printf 'port tcp\nctype tcp\n' > -/usr/local/etc/uucp/port
printf -"%s\n%s\n%s\n%s\n%s\n%s\n%s\n%s\n" \
"call-login *" \
"call-password *" \
"time any" \
"system shuttle" \
"address mail.s.shuttle.de" \
"commands rmail rnews" \
"port tcp" \
>/usr/local/etc/uucp/sys
cd -/usr/local/etc/uucp; chown uucp:uucp *; chmod o-rwx *
Activar uucico tras el arranque del sistema.
mkdir --p -/usr/local/etc/rc.d; cp uucp.sh -/usr/local/etc/rc.d

EOF $RCSfile: fase_3.mk,v $ vim: tabstop=4:
```

```
fase_3.mk.
```

## 7. #####

```
#####
make BATCH=YES install. ##
yes ##### ##
#####
(### ##### make install; ###
#####/#####13 ## fase_2.sh).
```

```
#####
#11 #####. ## #####
#####, ## ##
#####. ##
#####. ##### ## staroffice ##
```

```
cd -/usr/ports/editors/staroffice52
```

#####

```
make package
==> Building package for staroffice-5.2_1
Creating package -/usr/ports/editors/staroffice52/staroffice-5.2_1.tbz
Registering depends:.
Creating bzip'd tar ball in -/usr/ports/editors/staroffice52/staroffice-5.2_1.tbz'
```

```
#####:
```

```
pkg_add -/usr/ports/editors/staroffice52/staroffice-5.2_1.tbz
```

[illegible]

```
5-CURRENT
5-CURRENT, ##, ##### 5-CURRENT ##### 4-
STABLE # #####, #####
#####
##. ##### 4-STABLE
(#####).
#####:
```

- ## ## ## ## ## ##### ## ##### ## ##### (devfs) #####  
#####  
#####(8) ## ## ## ##,  
####.