Manolis Kiagias, MSc

FreeBSD 12.1-RELEASE

Base System to Desktop Environment

Chania, 2019

For comments / corrections: sonic2000gr@gmail.com

Installing FreeBSD 12.1-RELEASE

Before beginning the installation of an operating system, the BIOS has to be adjusted to boot from the intended installation media. Most of the newer PC hardware have the ability to boot off USB devices such as flash drives as well as from CD/DVD drives.

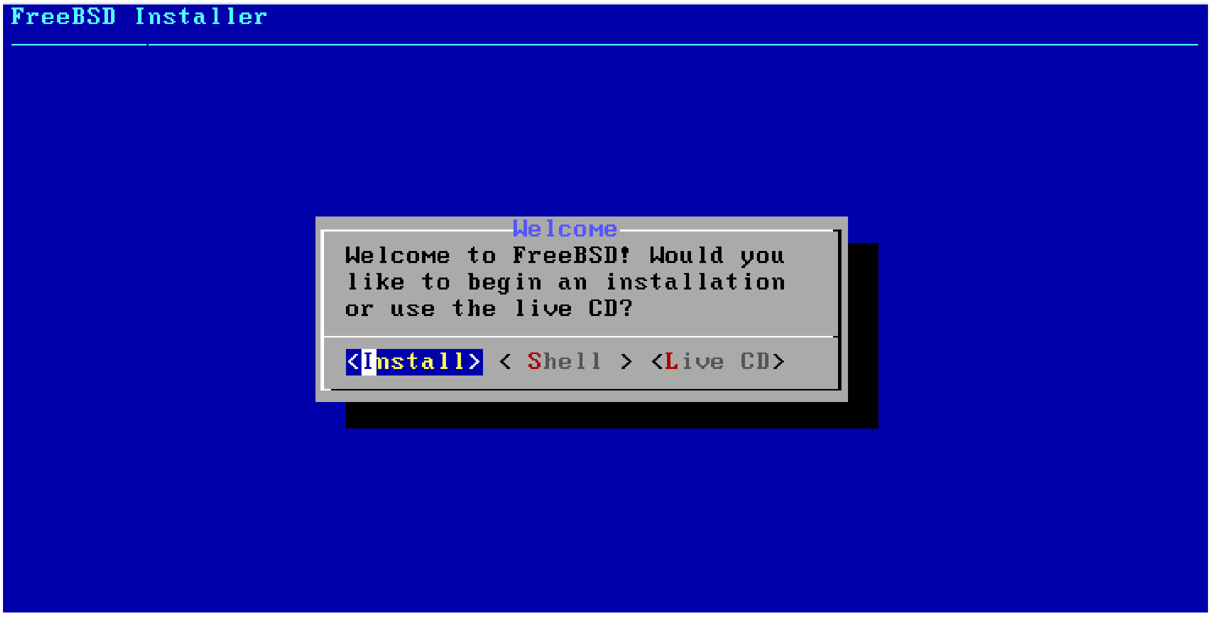
FreeBSD comes in different versions for 32 and 64bit PC architectures as well as many others. Typically, the 32bit version is intended for systems utilizing an older processor (or smaller amounts of RAM, generally less than 4GB) and the 64bit version for more modern processors and capable systems.

While FreeBSD is available both on CD and DVD, the DVD version is not required for this install. The DVD contains ready-made packages that are also available online. For this install we are using the 32bit disc1 ISO image and VirtualBox. If you wish, you may download the complete Appliance of this install here:

http://www.schoolspace.gr/files/FreeBSD-12.1-RELEASE.zip

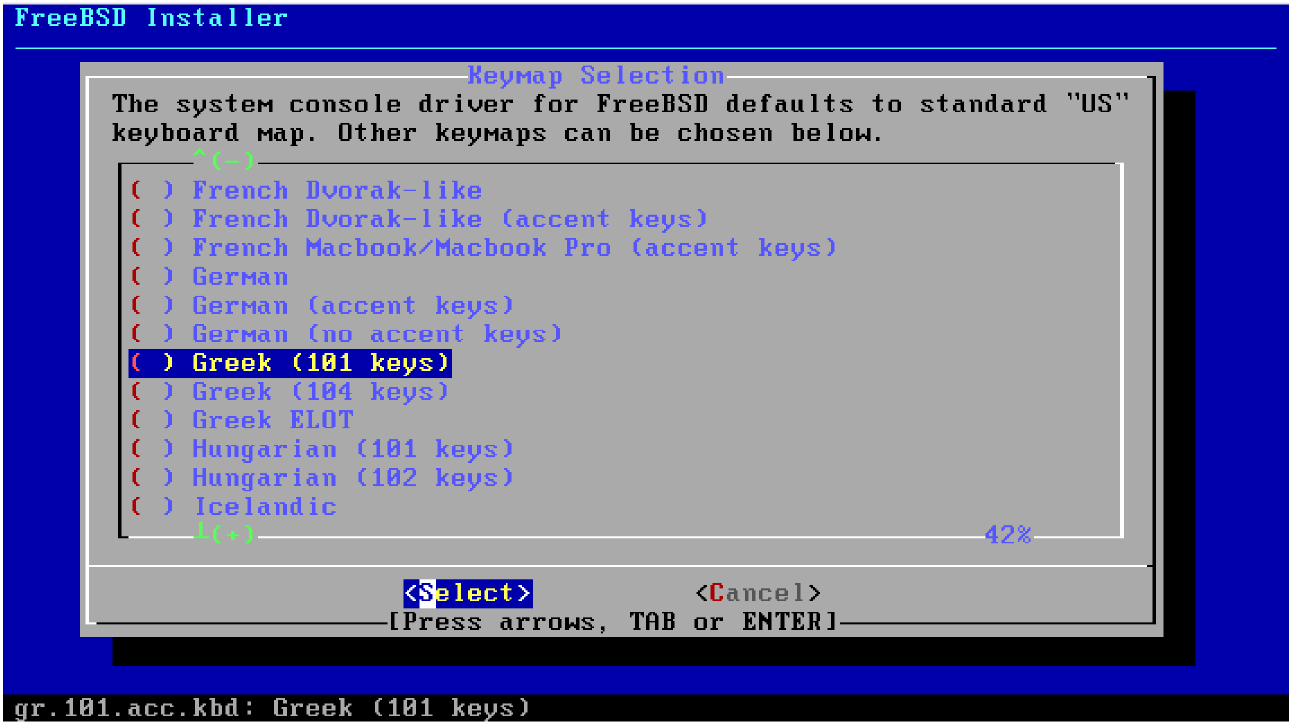
In the Appliance, thhe password for both **user** and **root** accounts is 1234

Booting off the installation media, after a while the following dialog appears:



Welcome to FreeBSD! Would you like to begin an installation or use the live CD?

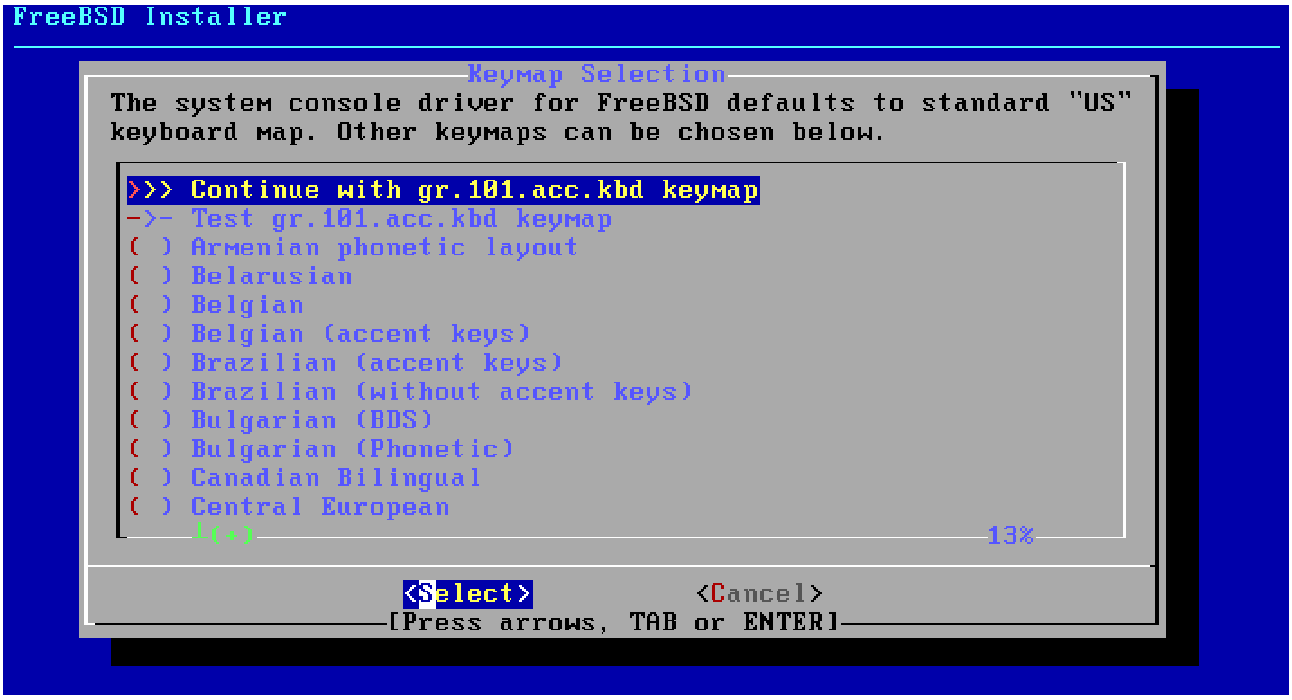
Just press Enter on the default Install option.



```
FreeBSD Installer

Keymap Selection
The system console driver for FreeBSD defaults to standard "US"
keyboard map. Other keymaps can be chosen below.
^(-)
( ) French Dvorak-like
( ) French Dvorak-like (accent keys)
( ) French Macbook/Macbook Pro (accent keys)
( ) German
( ) German (accent keys)
( ) German (no accent keys)
( ) Greek (101 keys)
( ) Greek (104 keys)
( ) Greek ELOT
( ) Hungarian (101 keys)
( ) Hungarian (102 keys)
( ) Icelandic
v(+)                                                      42%

        <Select>            <Cancel>
[Press arrows, TAB or ENTER]

gr.101.acc.kbd: Greek (101 keys)
```

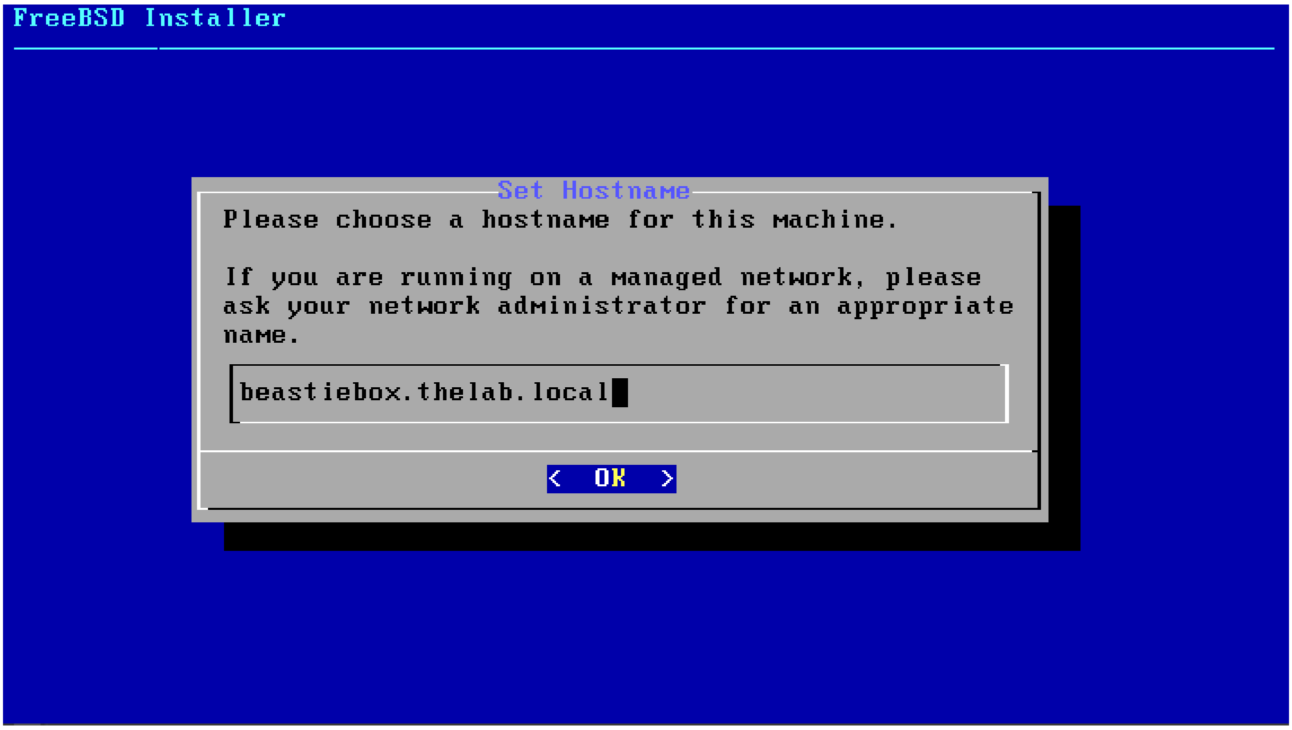
Select the desired keyboard layout. This is not a critical setting for a desktop system as it will not affect your GUI settings. If a layout does not seem to match your keyboard/language, just select the US layout.

During the installation process, use the arrow keys to move between options. Use the SPACE bar to mark any checkboxes. Move between different dialog parts (such as the main dialog and the OK / Cancel buttons) using the TAB button.



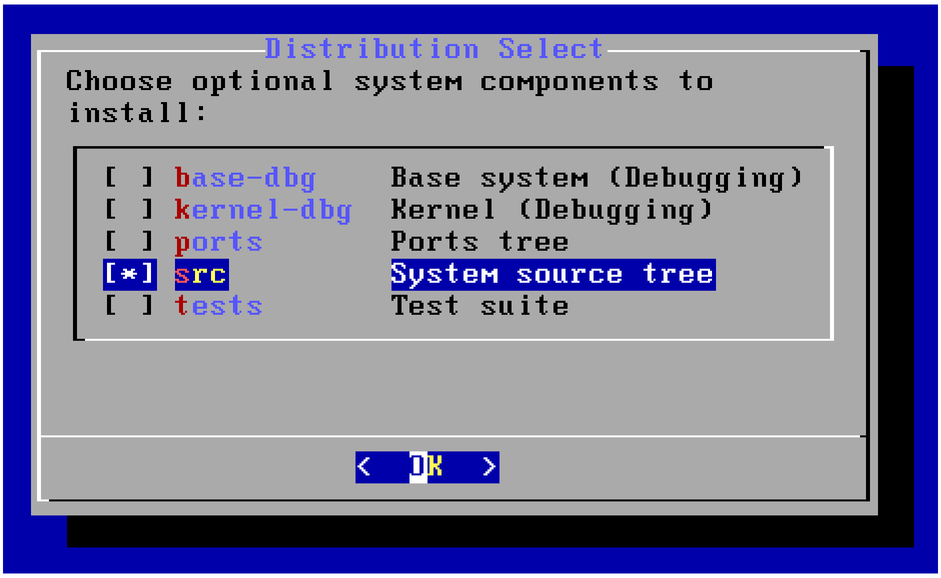
```
FreeBSD Installer

Keymap Selection
The system console driver for FreeBSD defaults to standard "US"
keyboard map. Other keymaps can be chosen below.

>>> Continue with gr.101.acc.kbd keymap
->- Test gr.101.acc.kbd keymap
( ) Armenian phonetic layout
( ) Belarusian
( ) Belgian
( ) Belgian (accent keys)
( ) Brazilian (accent keys)
( ) Brazilian (without accent keys)
( ) Bulgarian (BDS)
( ) Bulgarian (Phonetic)
( ) Canadian Bilingual
( ) Central European
v(+)                                                      13%

        <Select>            <Cancel>
[Press arrows, TAB or ENTER]
```

After selecting the keyboard layout, use the arrow keys to select “Continue wth <your selection>” and press ENTER on this dialog to confirm it.



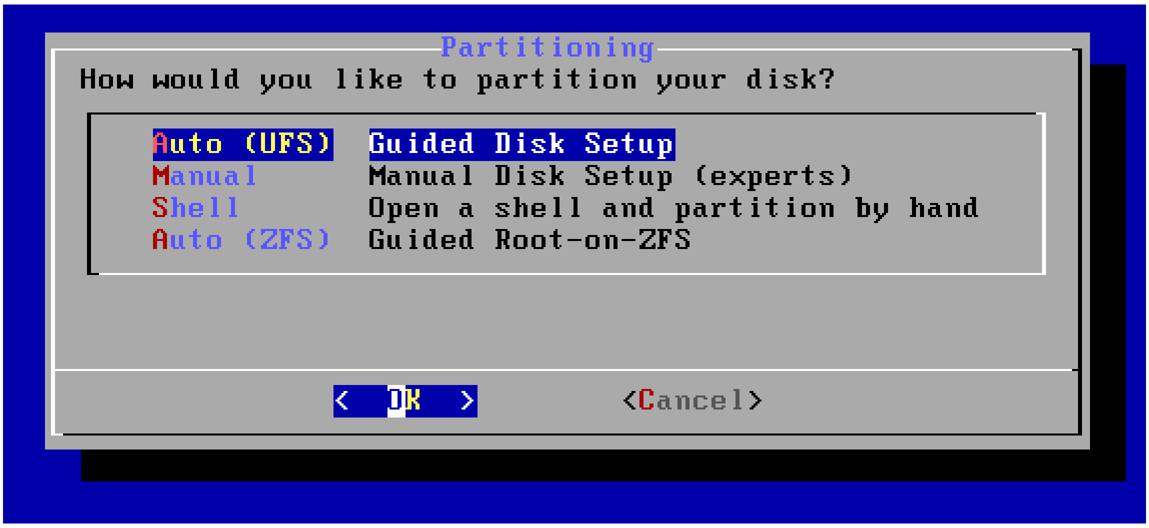
```
Please choose a hostname for this machine.

If you are running on a managed network, please
ask your network administrator for an appropriate
name.

beastiebox.thelab.local
```

Every system must have a hostname. Typically, a system also belongs to a domain. As our sample system is not directly connected to the Internet (it is most probably served by a home DSL router) any domain name may be chosen. In this example, the domain name is **thelab.local** and the hostname is **beastiebox**. Original hostnames for a system are many times based on plants, planets, constellations and animals. Something as dull as system01 may also be used!



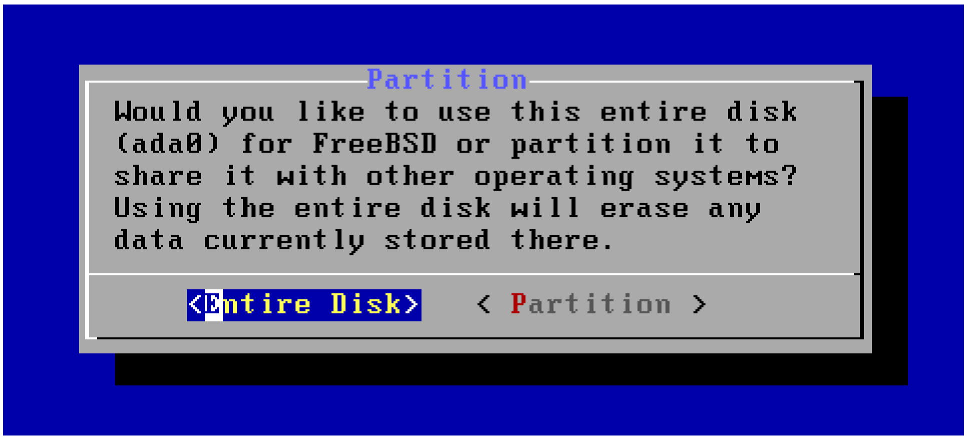
```
Choose optional system components to
install:

 [ ] base-dbg     Base system (Debugging)
 [ ] kernel-dbg   Kernel (Debugging)
 [ ] ports        Ports tree
 [*] src          System source tree
 [ ] tests        Test suite
```

Select the additional or optional components to install. There is no need to select the Ports tree as the version on the CD is outdated and the ports can be readily installed off the Internet. The system source code is useful as a reference and if you plan to build your own kernel. A few programs from the Ports Collection may also need the system source code in order to build.



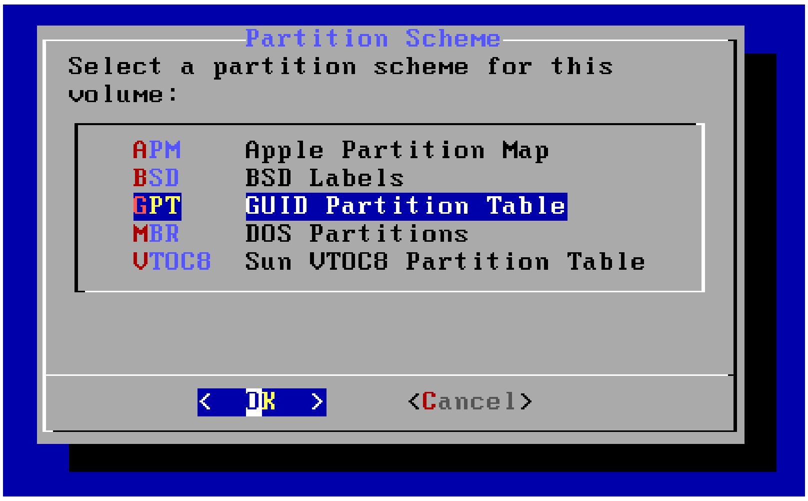
```
Partitioning
How would you like to partition your disk?

   Auto (UFS)   Guided Disk Setup
   Manual       Manual Disk Setup (experts)
   Shell        Open a shell and partition by hand
   Auto (ZFS)   Guided Root-on-ZFS
```

There are many different ways to partition the system's disk. The easiest method is to use the default **Guided** option. FreeBSD can be installed on the traditional UFS filesystem or the newer ZFS one. For an older system using the 32 bit installation media, the best option would be UFS. ZFS is recommended for modern systems with adequate amounts of RAM (4GB or more). For our VirtualBox installation, we selected UFS for simplicity.



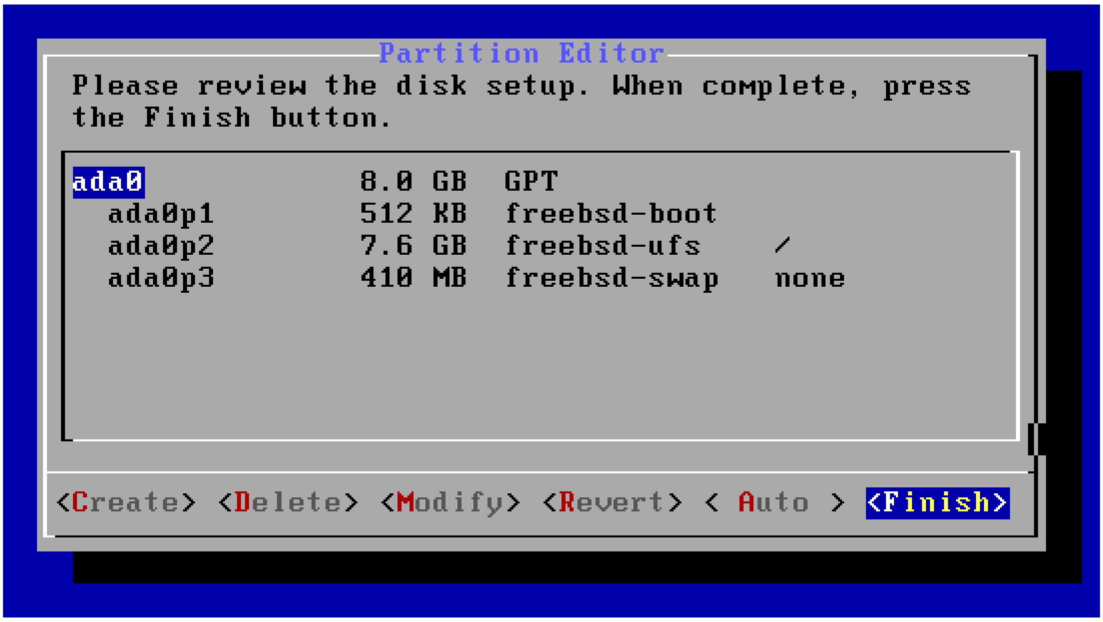
```
Partition
Would you like to use this entire disk
(ada0) for FreeBSD or partition it to
share it with other operating systems?
Using the entire disk will erase any
data currently stored there.
```

Assuming a completely empty disk, select the **Entire Disk** option. Note that this will erase anything on the disk and create appropriate BSD partitions! Multi boot systems may be created using the **Partition** option, however this is outside the scope of this concise guide.



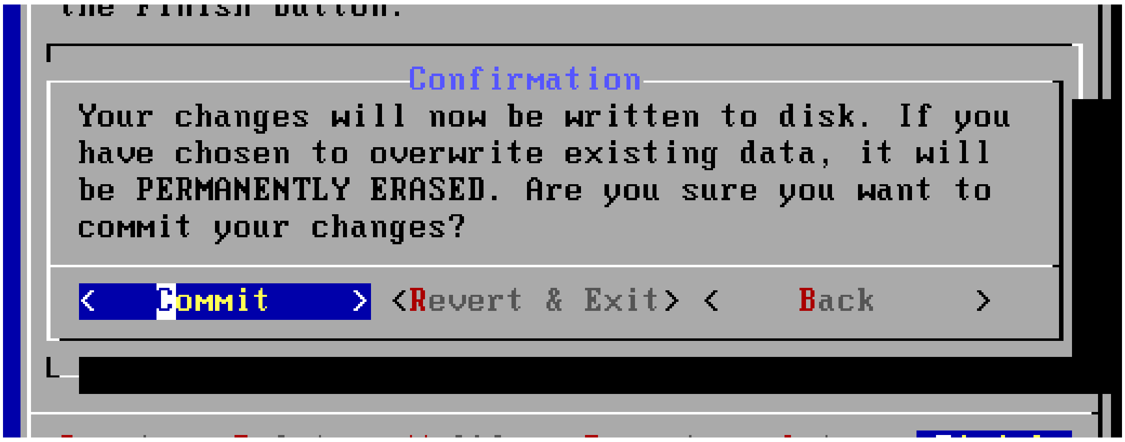
```
Partition Scheme
Select a partition scheme for this
volume:

   APM     Apple Partition Map
   BSD     BSD Labels
   GPT     GUID Partition Table
   MBR     DOS Partitions
   VTOC8   Sun VTOC8 Partition Table
```

The newer partitioning scheme is GPT and allows more than four primary partitions on a disk. Most system have no trouble booting from GPT partitions, so this option is recommended. The older MBR style partitions may also be used.



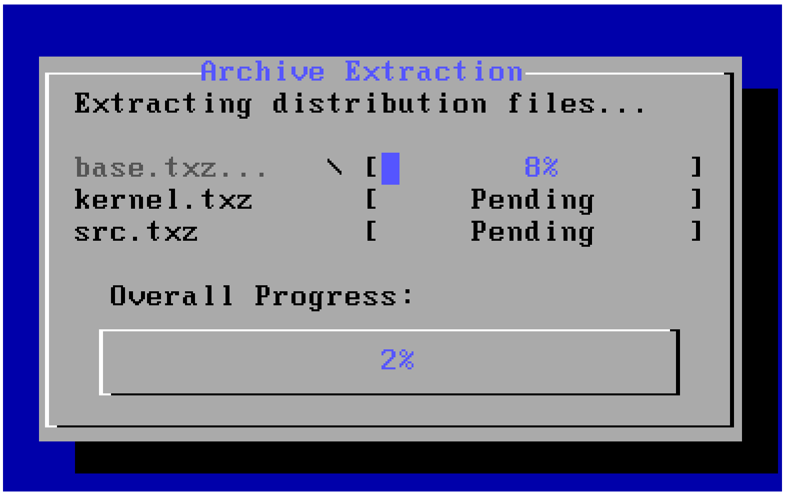
```
Please review the disk setup. When complete, press
the Finish button.

ada0              8.0 GB   GPT
  ada0p1          512 KB   freebsd-boot
  ada0p2          7.6 GB   freebsd-ufs     /
  ada0p3          410 MB   freebsd-swap    none
```

Depending on the disk and RAM size of your system, the installation program will suggest different partition sizes. The default settings may usually be accepted. Using the options in this dialog, the sizes may be customized or additional partitions may be created. Sometimes it is useful to shrink the size of the root partition (marked as “/”) and create a separate partition for **/home.** For this sample installation, simply select **Finish**.



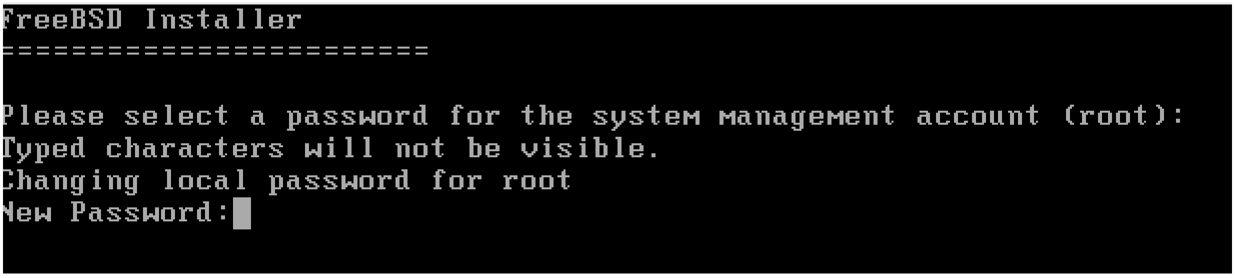
```
the Finish button.

Your changes will now be written to disk. If you
have chosen to overwrite existing data, it will
be PERMANENTLY ERASED. Are you sure you want to
commit your changes?
```

This is the final warning: After selecting **Commit**, changes will be written to disk, partitions will be created and formatted and files will be copied. Any existing data on the disk will be lost after pressing **Commit**.



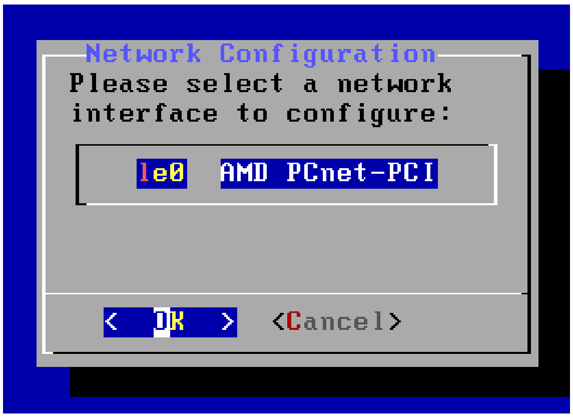
```
Extracting distribution files...

base.txz...      \ [       8%        ]
kernel.txz         [     Pending     ]
src.txz            [     Pending     ]

  Overall Progress:

            2%
```

File copying will take a while.



```
FreeBSD Installer
========================

Please select a password for the system management account (root):
Typed characters will not be visible.
Changing local password for root
New Password:
```

Set the password for root (admin) user. Use a safe, long password that may not be guessed. Your birthday, phone number and 1234 are all unsafe passwords!

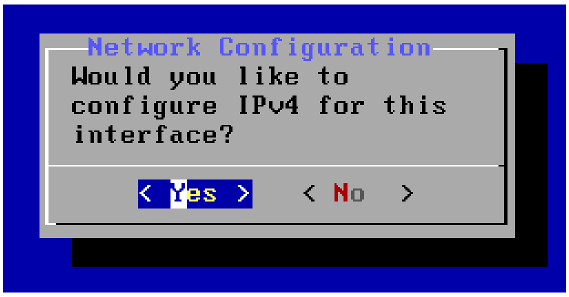


Network Configuration

Please select a network interface to configure:

le0 AMD PCnet-PCI

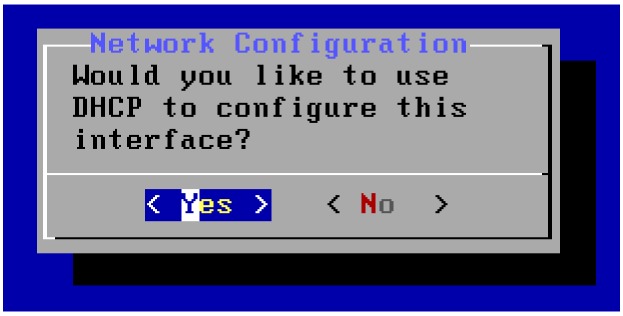
The above dialog will only appear if the installation program detects a network card. If more than one network cards are detected, their models will be listed here allowing you to choose which one to configure. If your system is equipped with a WiFi adapter that is recognized by FreeBSD, a number of dialogs will be presented allowing connection to the home network after typing a password (if applicable).



Network Configuration

Would you like to configure IPv4 for this interface?

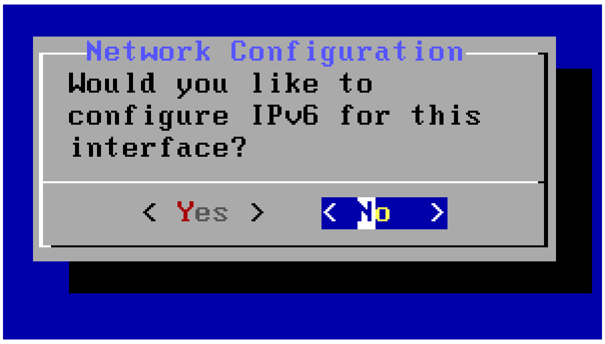
The answer here is obviously yes.



Network Configuration

Would you like to use DHCP to configure this interface?

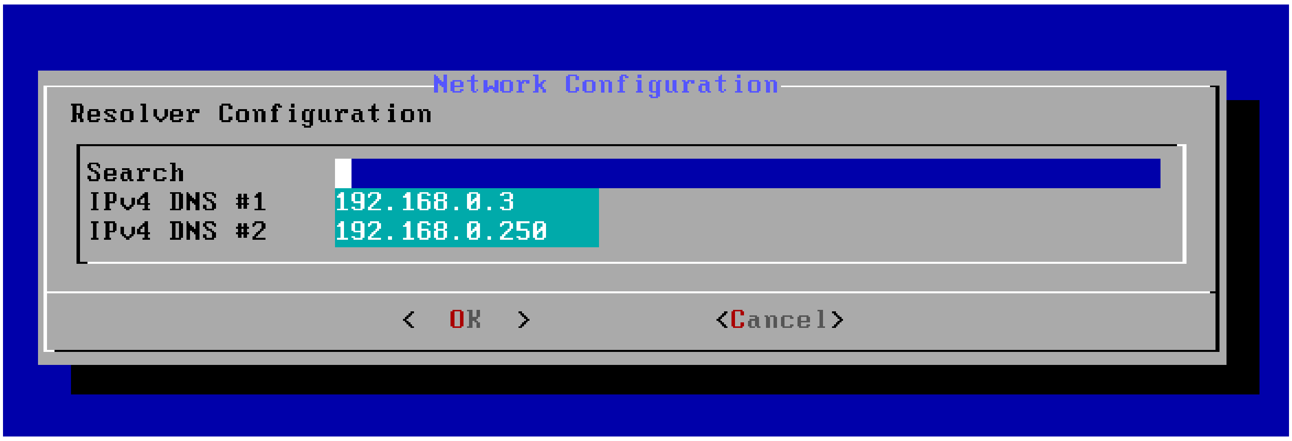
If **Yes** is selected, the network card will be autoconfigured by a DHCP server on your network (most probably your home router). Otherwise, a number of dialogs will be shown, allowing the settings to be made manually.



Network Configuration

Would you like to configure IPv6 for this interface?

If IPv6 is available select **Yes**. Otherwise select **No**.

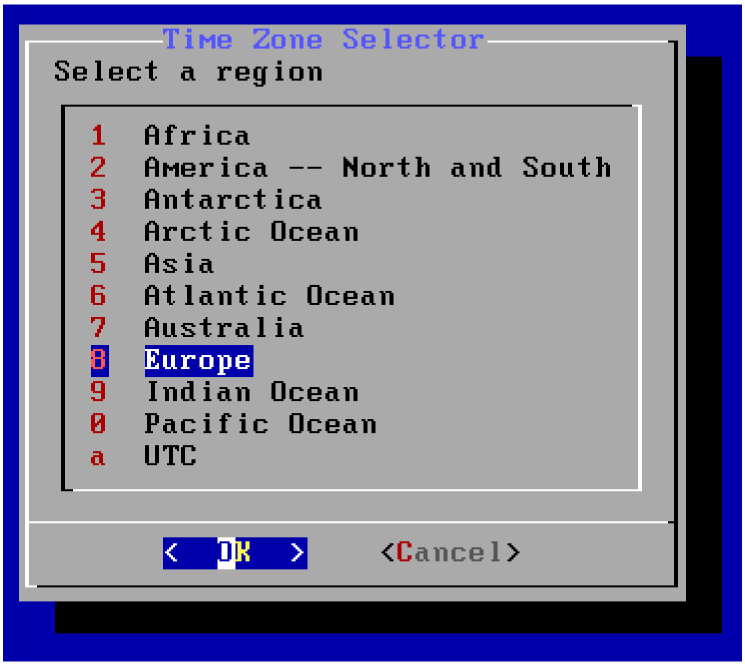


Network Configuration

Resolver Configuration

```
Search
IPv4 DNS #1      192.168.0.3
IPv4 DNS #2      192.168.0.250
```

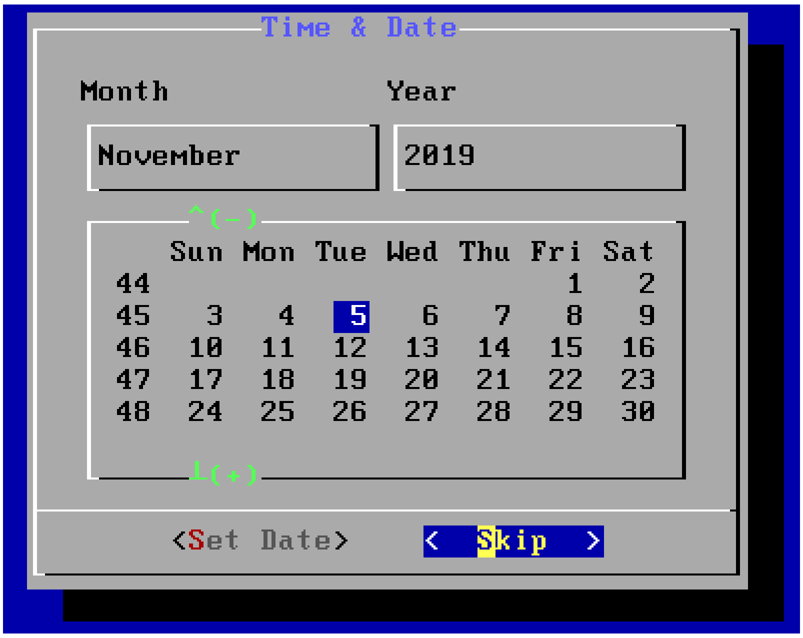
Having selected DHCP for the network settings, this dialog will show some of the settings obtained by the DHCP server. These are the DNS servers that will be used (will be automatically written to the generated **/etc/resolv.conf** file).



Time Zone Selector

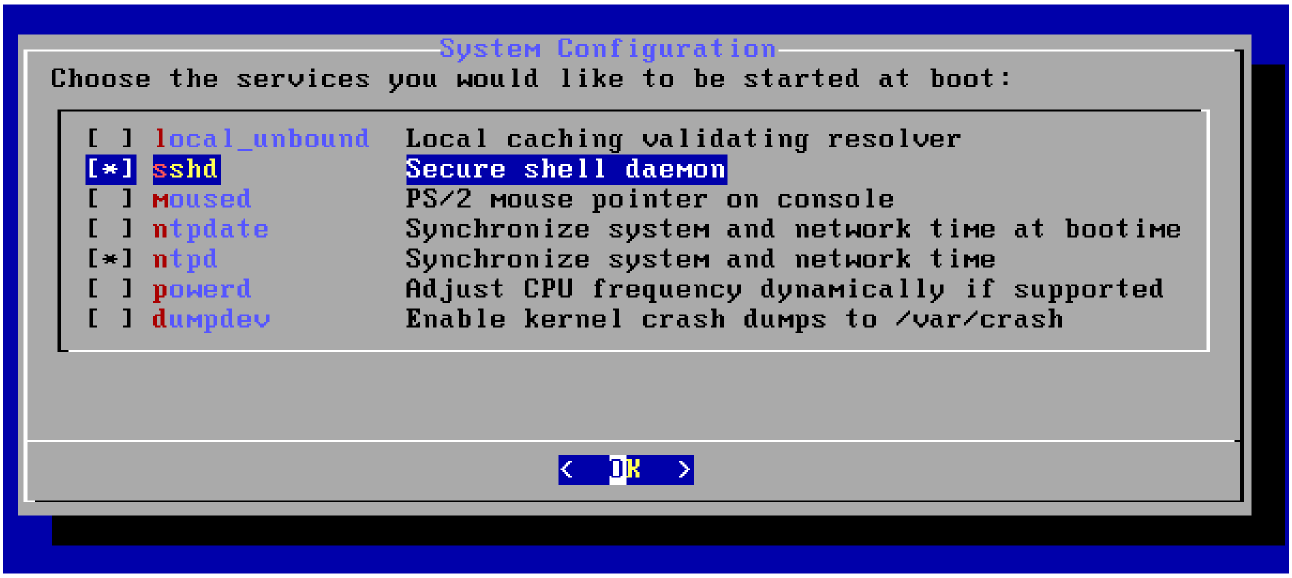
Select a region

Select your region and country / time zone.



```
                 Time & Date

  Month                 Year
  November              2019

           ^(-)
      Sun  Mon  Tue  Wed  Thu  Fri  Sat
  44                                1    2
  45    3    4    5    6    7    8    9
  46   10   11   12   13   14   15   16
  47   17   18   19   20   21   22   23
  48   24   25   26   27   28   29   30
           v(+)

        <Set Date>       <  Skip  >
```

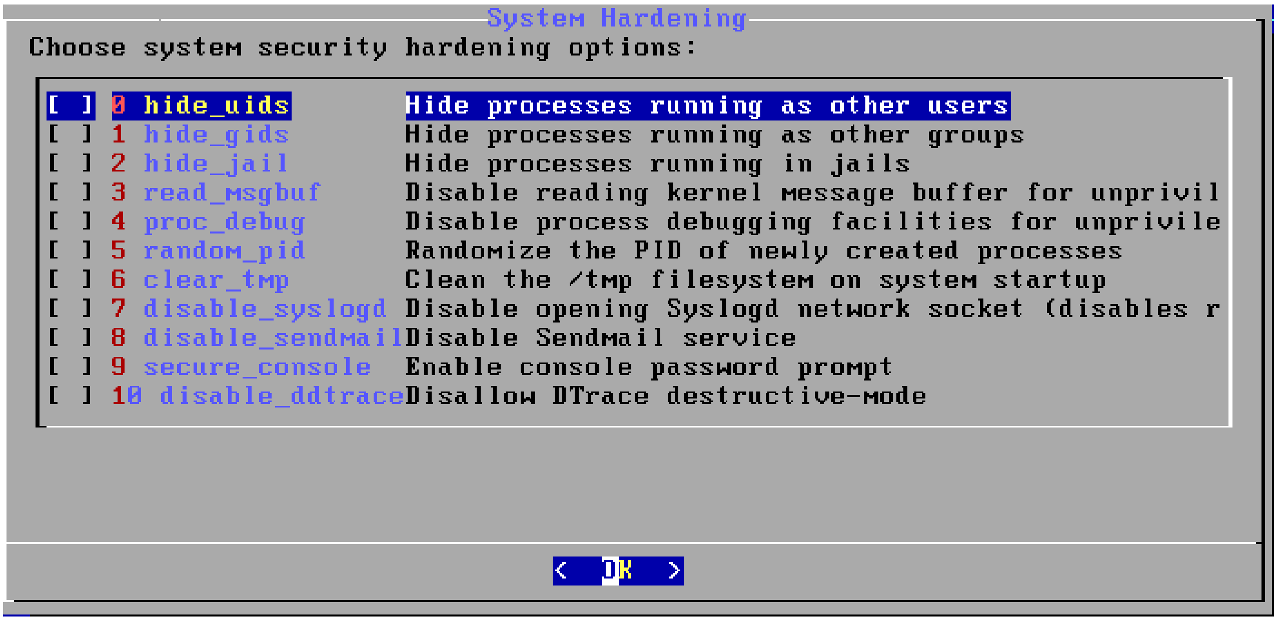
The above dialog provides a way to confirm / set the correct date for your system. It is followed by a similar dialog allowing time setting. Please confirm the settings are correct. FreeBSD assumes the system BIOS uses UTC time.



```
                        System Configuration
 Choose the services you would like to be started at boot:

   [ ] local_unbound   Local caching validating resolver
   [*] sshd            Secure shell daemon
   [ ] moused          PS/2 mouse pointer on console
   [ ] ntpdate         Synchronize system and network time at boottime
   [*] ntpd            Synchronize system and network time
   [ ] powerd          Adjust CPU frequency dynamically if supported
   [ ] dumpdev         Enable kernel crash dumps to /var/crash

                             <  OK  >
```

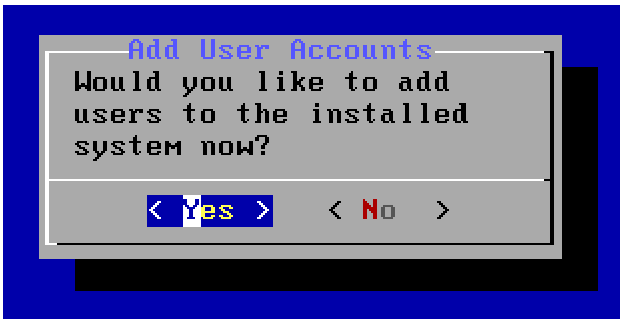
Select the services (or daemons) to run at startup:

- For remote shell access, select **sshd.** This is usually enabled.
- For mouse operation on the console, select **moused**
- For automatic, Internet based clock synchronization, select **ntpd**. This option is usually enabled.
- For CPUs with power saving features, you may wish to select **powerd**.
- **dumpdev** is probably not useful for typical desktop use.



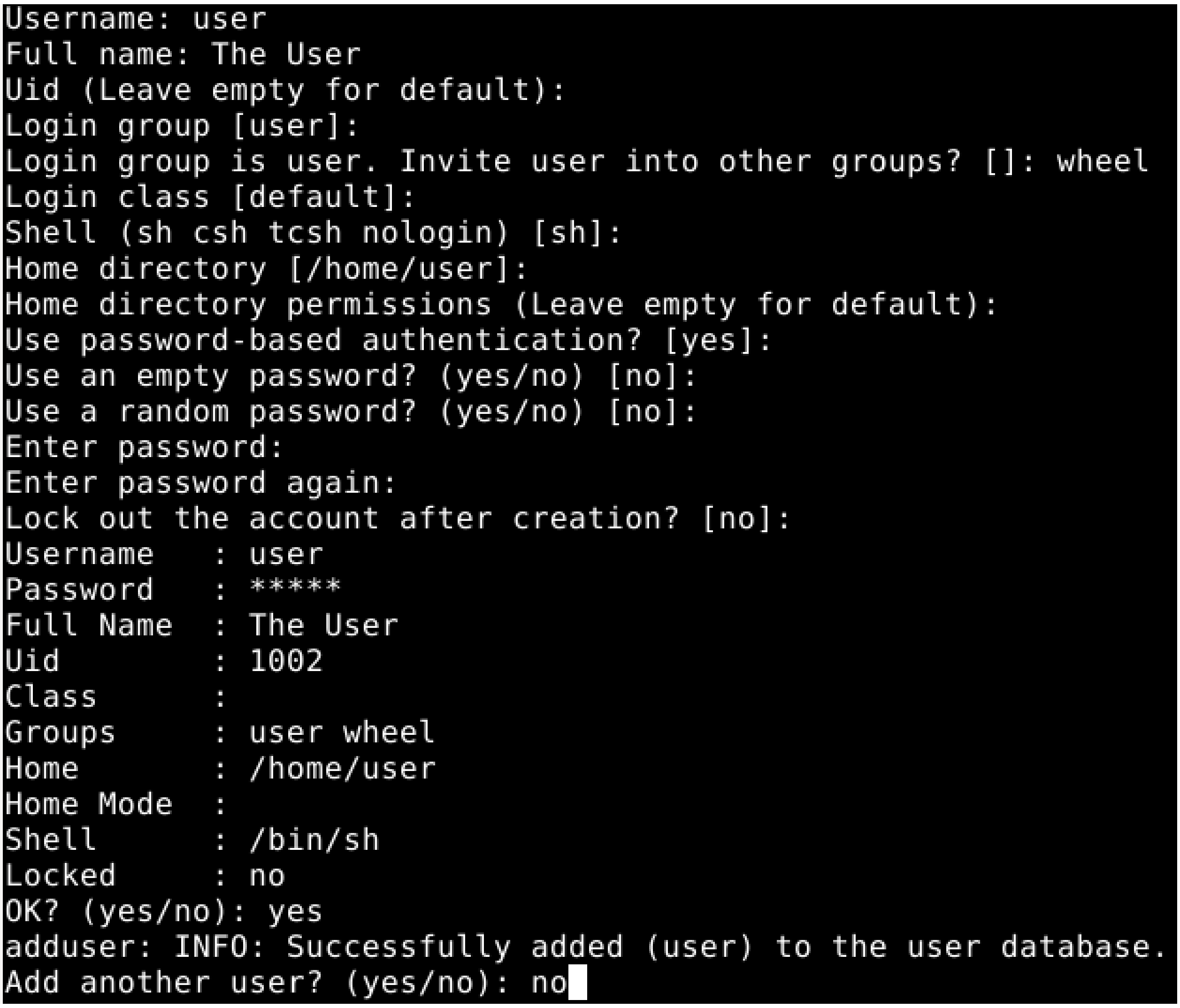
```
System Hardening
Choose system security hardening options:

[ ] 0 hide_uids         Hide processes running as other users
[ ] 1 hide_gids         Hide processes running as other groups
[ ] 2 hide_jail         Hide processes running in jails
[ ] 3 read_msgbuf       Disable reading kernel message buffer for unprivil
[ ] 4 proc_debug        Disable process debugging facilities for unprivile
[ ] 5 random_pid        Randomize the PID of newly created processes
[ ] 6 clear_tmp         Clean the /tmp filesystem on system startup
[ ] 7 disable_syslogd   Disable opening Syslogd network socket (disables r
[ ] 8 disable_sendmailDisable Sendmail service
[ ] 9 secure_console    Enable console password prompt
[ ] 10 disable_ddtraceDisallow DTrace destructive-mode

                          <  OK  >
```

This dialog allows selection of system hardening options. We will be leaving these blank for now, as explaining them is beyond the scope of this guide. You are advised to study the relevant sections in the FreeBSD Handbook to decide whether to use any of these options. Note that all these options may be changed at any time after the initial install of the system.



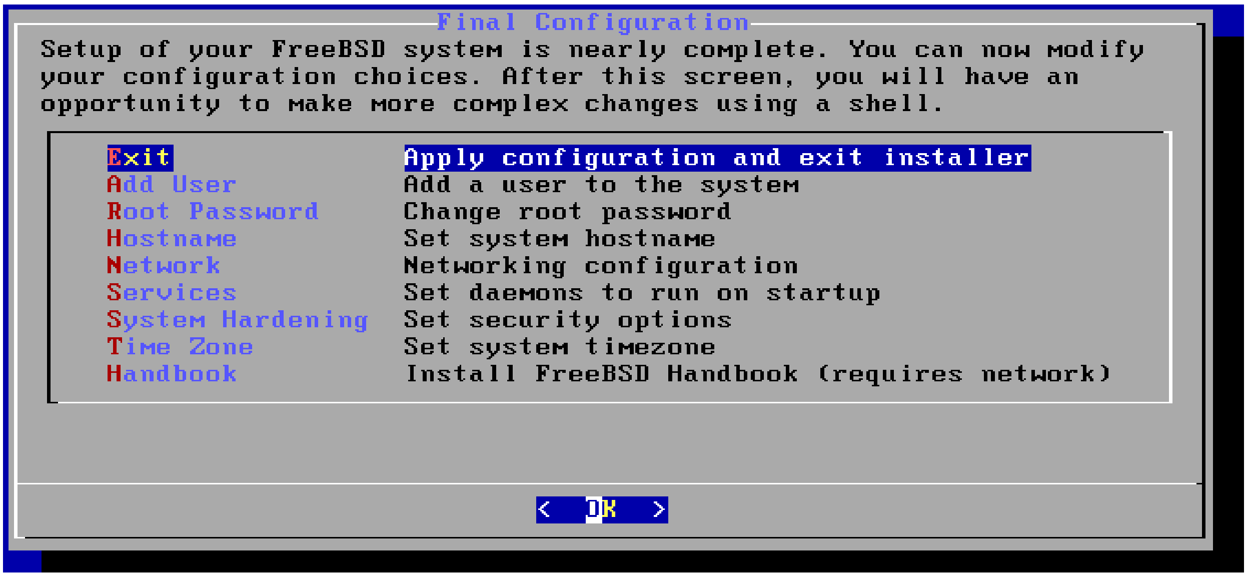
```
Add User Accounts
Would you like to add
users to the installed
system now?

 < Yes >    < No  >
```

Add users to the system. It is advisable to add at least one standard user now or immediately after the installation is finished.



```
Username: user
Full name: The User
Uid (Leave empty for default):
Login group [user]:
Login group is user. Invite user into other groups? []: wheel
Login class [default]:
Shell (sh csh tcsh nologin) [sh]:
Home directory [/home/user]:
Home directory permissions (Leave empty for default):
Use password-based authentication? [yes]:
Use an empty password? (yes/no) [no]:
Use a random password? (yes/no) [no]:
Enter password:
Enter password again:
Lock out the account after creation? [no]:
Username    : user
Password    : *****
Full Name   : The User
Uid         : 1002
Class       :
Groups      : user wheel
Home        : /home/user
Home Mode   :
Shell       : /bin/sh
Locked      : no
OK? (yes/no): yes
adduser: INFO: Successfully added (user) to the user database.
Add another user? (yes/no): no
```

Like hostnames, usernames are entirely up to choice. The example shows a rather simplistic **user** username. It is good practice to make at least one of the standard user accounts a member of the **wheel** group. Members of **wheel** can use the **su** command to become root when needed (without having to logout/login). For the rest of the users, the More users may be added after the initial installation using the **adduser** command as root. This is the same command actually used in the installer above.

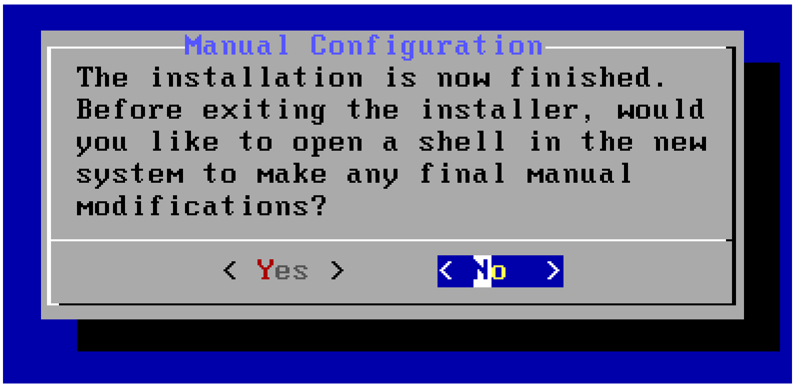


Final Configuration

Setup of your FreeBSD system is nearly complete. You can now modify your configuration choices. After this screen, you will have an opportunity to make more complex changes using a shell.

Exit   Apply configuration and exit installer
Add User   Add a user to the system
Root Password   Change root password
Hostname   Set system hostname
Network   Networking configuration
Services   Set daemons to run on startup
System Hardening   Set security options
Time Zone   Set system timezone
Handbook   Install FreeBSD Handbook (requires network)
< OK >

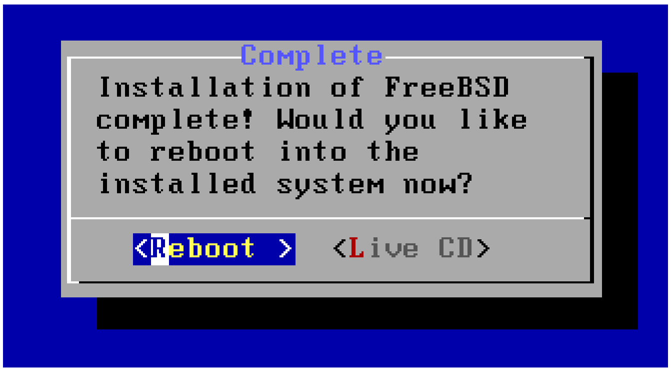
We’re at the end of the installation, and we may choose **Exit**.



Manual Configuration

The installation is now finished. Before exiting the installer, would you like to open a shell in the new system to make any final manual modifications?

No more changes are needed, select **No.**

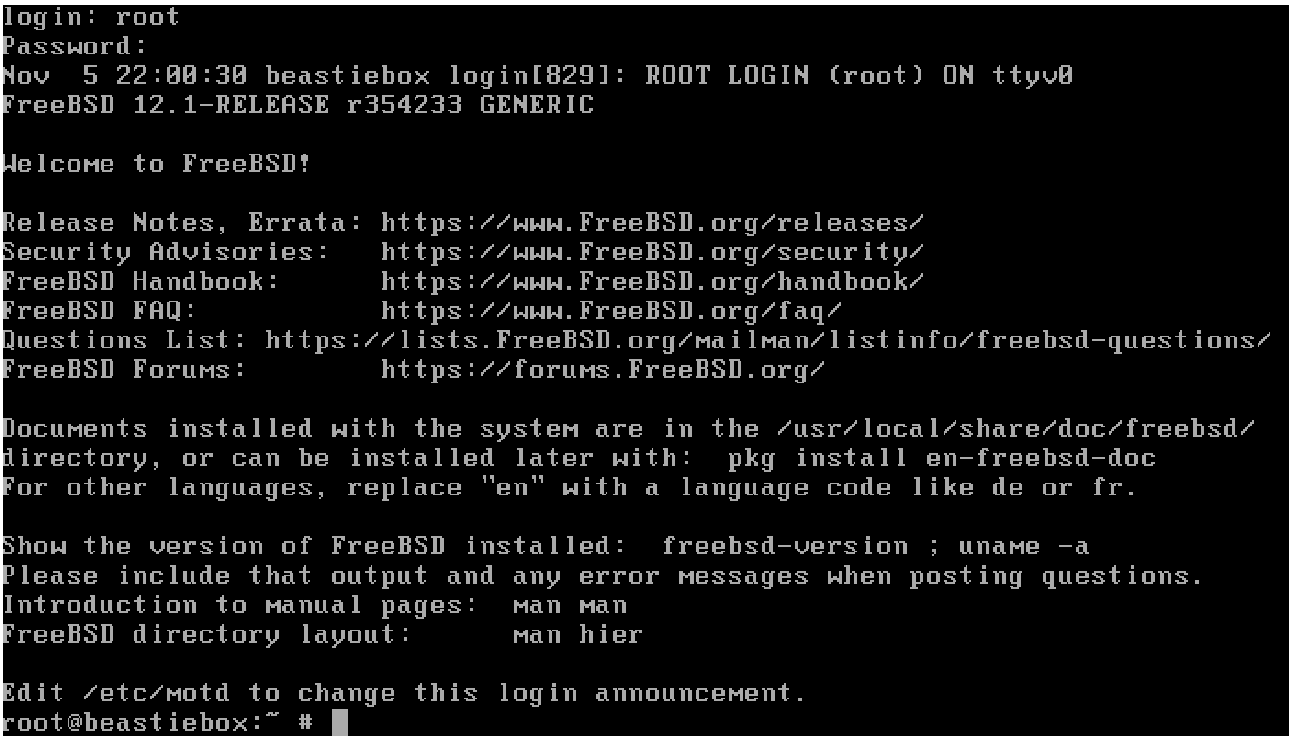


Complete

Installation of FreeBSD complete! Would you like to reboot into the installed system now?

Finally, select **Reboot** to start your new system. Remember to remove the installation media before restarting!

At the end of the first startup, at the **Login:** prompt, enter **root** and the use the password typed earlier in the installation program.



```
login: root
Password:
Nov  5 22:00:30 beastiebox login[829]: ROOT LOGIN (root) ON ttyv0
FreeBSD 12.1-RELEASE r354233 GENERIC

Welcome to FreeBSD!

Release Notes, Errata: https://www.FreeBSD.org/releases/
Security Advisories:   https://www.FreeBSD.org/security/
FreeBSD Handbook:      https://www.FreeBSD.org/handbook/
FreeBSD FAQ:           https://www.FreeBSD.org/faq/
Questions List: https://lists.FreeBSD.org/mailman/listinfo/freebsd-questions/
FreeBSD Forums:        https://forums.FreeBSD.org/

Documents installed with the system are in the /usr/local/share/doc/freebsd/
directory, or can be installed later with:  pkg install en-freebsd-doc
For other languages, replace "en" with a language code like de or fr.

Show the version of FreeBSD installed:  freebsd-version ; uname -a
Please include that output and any error messages when posting questions.
Introduction to manual pages:  man man
FreeBSD directory layout:      man hier

Edit /etc/motd to change this login announcement.
root@beastiebox:~ #
```

We are now ready to start updating our new system and install application programs.

Update the Base System

Note: The “#” (hash) symbol means the next command must be run by the root user. The “$” (dollar) sign means the next command must be run by a standard user. These symbols should not be typed as part of the commands!

Before any application programs are installed, it is good practice to update the base system with the latest security and bug fixes:

```
# freebsd-update fetch install
```

```
Looking up update.FreeBSD.org mirrors... 5 mirrors found.
Fetching public key from update4.freebsd.org... done.
Fetching metadata signature for 12.1-RELEASE from update4.freebsd.org... done.
Fetching metadata index... done.
Fetching 2 metadata files... done.
Inspecting system... done.
Preparing to download files...
The following files will be updated as part of updating to 12.1-RELEASE-p1:
/bin/freebsd-version
/boot/kernel/kernel
...
```

```
Installing updates...  done.
```

Many security updates contain fixes to the kernel and require the system to be rebooted after the installation:

```
# shutdown -r now
```

After rebooting, login again as root.

Installing the pkg Command, the Interface to Package Management

In recent FreeBSD versions, the default package management system was switched to a new version called **pkgng.** The system allows easy installation of ready made packages that are available in the FreeBSD Project servers, without the need for time-consuming source builds. Ready-made packages are ideal for desktop systems where a lot of complex and large programs are needed for a functional GUI. The traditional Ports Collection method is also still available and may be used when special build options are required or where packages are not available (for example due to licensing issues).

To begin working with packages, the pkg command must first be initialized. As root, simply type:

```
# pkg
```

```
The package management tool is not yet installed on your system.

Do you want to fetch and install it now? [y/N]: y

Bootstrapping pkg from pkg+http://pkg.FreeBSD.org/FreeBSD:12:i386/quarterly, please wait...

Verifying signature with trusted certificate pkg.freebsd.org.2013102301... done

Installing pkg-1.12.0...

Extracting pkg-1.12.0: 100%
```

The program will be downloaded and installed automatically. Update the database of available packages:

```
# pkg update
```

```
Updating FreeBSD repository catalogue...
Fetching meta.txz: 100%    944 B   0.9kB/s    00:01
Fetching packagesite.txz: 100%    6 MiB 504.5kB/s    00:13
Processing entries: 100%
FreeBSD repository update completed. 32397 packages processed.
All repositories are up to date.
```

We are now ready to install some basic command line tools.

Installing Basic Command Line Tools

Some basic tools for the command line are:

bash	The Bash shell is very common in the Linux world and you may wish to use it as a shell for many standard users (not for root!)
screen	Screen is a useful utility when a session needs to be left active with the user logging and reconnecting to it from a different terminal.
zip, **unzip** **unrar**	These utilities allow handling of compressed archives.
sudo	A utility to run commands as root from a user account.

To install all of the above, simply type:

```
# pkg install bash screen sudo zip unzip unrar
```

```
Updating FreeBSD repository catalogue...
FreeBSD repository is up to date.
All repositories are up to date.
Updating database digests format: 100%
The following 8 package(s) will be affected (of 0 checked):
...
```

Setting Up Sudo

To allow a user to issue commands as root, an initial setup of the sudo command has to be performed using the **visudo** utility. When **visudo** is executed, an editor is opened (typical vi) with the sudo configuration file already loaded. To temporarily set another, easier, editor for visudo, change the EDITOR environment variable:

```
# setenv EDITOR ee
```

Execute the visudo command:

```
# visudo
```

Remove the comment symbol (#) from the wheel line:

```
# %wheel = ALL (ALL) ALL
```

so it becomes:

```
%wheel = ALL (ALL) ALL
```

This change will allow everyone in the wheel group to issue commands as root. If this is not desirable, individual usernames may be added like this:

```
nikos = ALL (ALL) ALL
```

Save the file (if using ee, press ESCAPE, Leave Editor and Save). Sudo allows more complex configurations where a user may have limited root abilities, however this is beyond the scope of this guide.

To test sudo functionality, logout out from the root account and login again as the standard user created during setup (which should belong to the wheel group). Issue a simple (harmless) command:

```
$ sudo ls
```

The system will ask the user password and execute the command.

Configuring the Bash Shell for the Standard User

We may change the shell for our standard user to the more common “bash” shell we previously installed. Bash is the default shell in most Linux distributions and is well enhanced compared to the standard **/bin/sh**.

We never change the shell to the root account! Root’s shell is **/bin/csh.**

Logout of root, login as user, and type:

```
$ chsh -s bash
```

You will need to enter your password to confirm the change.

Bash has two primary configuration files: **.profile** and **.bashrc**

In Unix, all files beginning with a dot (dotfiles) are hidden. The **.profile** file already exists and needs a few simple changes, but **.bashrc** must be created from scratch. For convenience, download and install our versions as follows:

```
$ cd ~
```

```
$ fetch http://www.schoolspace.gr/files/dotfiles-us.zip
```

Unzip the files:

```
$ unzip -o dotfiles-us.zip
```

Logout and login as standard user to see the difference.

The **.profile** file is loaded every time the user logs in. The **.bashrc** file is loaded when a non-login shell is started. A non-login shell is started when the user starts bash manually or by opening a graphic terminal in a GUI environment. Both files contain settings for the shell and the user's environment. It is convenient to put most settings in **.bashrc** and have this file loaded in both login and non-login shells. For this purpose, the following lines have been added to the end of **.profile**:

```
if [ -f ~/.bashrc ]; then
  source ~/.bashrc
fi
```

In **.bashrc,** lines like::

```
export LANG=en_US.UTF-8
```

set environment variables for the user. The shell remembers these settings and makes them available to programs that request their values. For example, the line above sets the system language for the user to US English, while the line below sets the default editor to ee:

```
export EDITOR=ee
```

When a program requests an editor to open a file, the easy editor (ee) will be used. There are also alias lines like the following:

```
alias ls='ls -G'
```

When the user executes the **ls** command, **ls -G** will be the one actually executed. The **-G** parameter in **ls** is used to enable color coded files and directories in the output. Finally the line:

```
PS1=...
```

changes the **prompt** of the shell and is responsible for the blue color and the extra info appearing in front of the "$" sign.

Have a look at the rest of the lines by opening the file using ee:

```
$ ee ~/.bashrc
```

Installing and Setting Up a Graphic Environment

Installation of a GUI consists of the following steps:

- Installing **Xorg Server**
- Installing the desired **Desktop Environment (DE)** or **Window Manager (WM)**
- Setting up and testing the installation
- Optionally installing a keyboard layout switching solution (if the user needs to switch between two or more languages / layouts)
- Optionally installing a **login manager** so the system starts by default to a GUI.
- Installing additional programs for media playback, web browsing, document creation etc.

Installing Xorg Server

This is very simple. As root:

```
# pkg install xorg
```

For a **VirtualBox** installation, install the **additions** as well:

```
# pkg install virtualbox-ose-additions
```

Install some extra fonts:

```
# pkg install liberation-fonts-ttf urwfonts-ttf freefont-ttf webfonts
```

Edit the file **/etc/rc.conf** and add this line:

```
dbus_enable=”YES”
```

The **rc.conf** file is one of the most important FreeBSD configuration files. It contains the network, hostname and console settings. It also contains lines describing the services (or daemons) to be started at system startup as well as some of their optional parameters. For every service that starts when the system is booted, there is a line like the following:

```
<servicename>_enable=”YES”
```

The order of appearance of these lines in **rc.conf** is not important. The FreeBSD startup system will always start system services in the correct order (for example, taking care to start services that need the network after the network connection is established).

If you followed our installation instructions this far, your user account is already a member of the wheel group. If not, type the following command:

```
# pw groupmod wheel –m user
```

The user account may also be added to the video group. The user must be on one of these groups (wheel or video) for Xorg to work correctly:

```
# pw groupmod video –m user
```

(use the account name created during installation if different from ‘user’)

For VirtualBox only:

For virtualbox installations, add these lines to **/etc/rc.conf:**

```
vboxguest_enable=”YES”
vboxservice_enable=”YES”
vboxservice_flags="--disable-timesync"
```

Normally, Xorg configures itself automatically to work with the graphics card / monitor it detects. There is usually no need to create an **xorg.conf** file like the older versions. However graphics card drivers packages may need to be installed depending on your hardware:

- For **VirtualBox** the graphics driver is included in the package **virtualbox-ose-additions** we already installed
- For common Intel graphics cards (included in most laptops and many desktop systems with onboard graphics) install **drm-kmod:**

  ```
  # pkg install drm-kmod
  ```

- For Nvidia based cards there are drivers available directly from Nvidia. To install the latest driver:

  ```
  # pkg install nvidia-driver
  ```

- For AMD based cards, install **drm-kmod:**

  ```
  # pkg install drm-kmod
  ```

You may find more information in the FreeBSD Handbook and the FreeBSD Wiki:

https://www.freebsd.org/doc/handbook/x-config.html

https://wiki.freebsd.org/Graphics

Installing the Desktop Environment

FreeBSD does not come with a default GUI but many DEs and WMs are available as packages and in the Ports Collection. A simple, non-resource intensive GUI with a familiar look and feel is XFCE. Installation is very easy:

```
# pkg install xfce
```

Logout of root and login as a standard user. Use ee to create the **.xinitrc** file:

```
$ cd ~
$ ee .xinitrc
```

Content:

```
export LANG=en_US.UTF-8
exec startxfce4
```

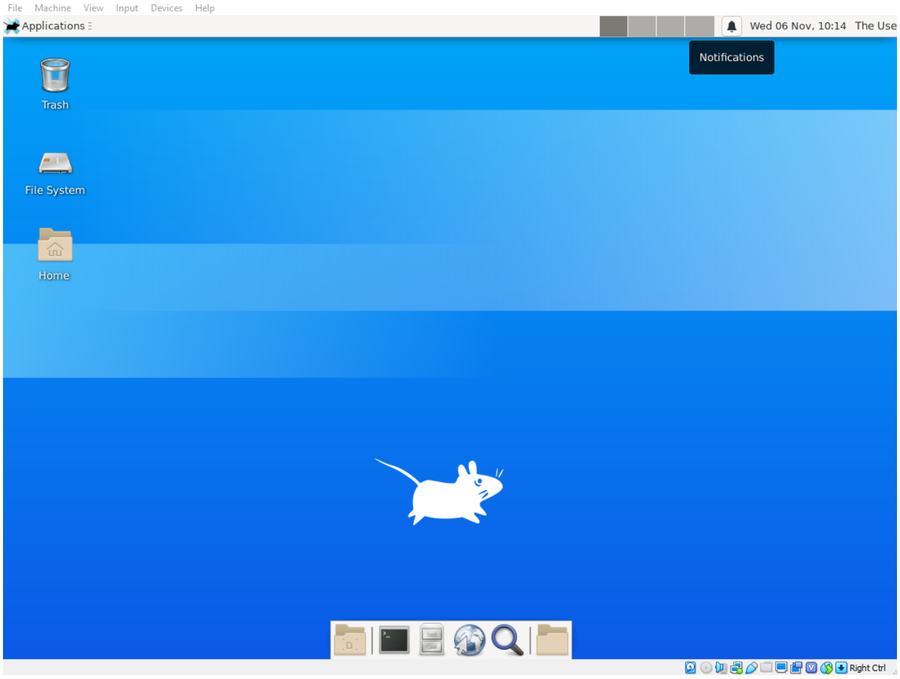
Restart the system to load all relevant drivers and daemons we previously installed:

```
# shutdown –r now
```

Then, login as standard user and execute:

```
$ startx
```

The default XFCE desktop will soon appear:



We never execute a GUI as root!

Starting Up Directly to Desktop Environment

In most desktop systems we completely forego the text-based login and use a **login manager** to directly login in the graphic environment of our choice. XFCE does not provide its own login manager but we can use one like GDM (from GNOME) or KDM (from KDE). For this test installation we picked a simpler login manager called **slim.** To install this, open a terminal and type:

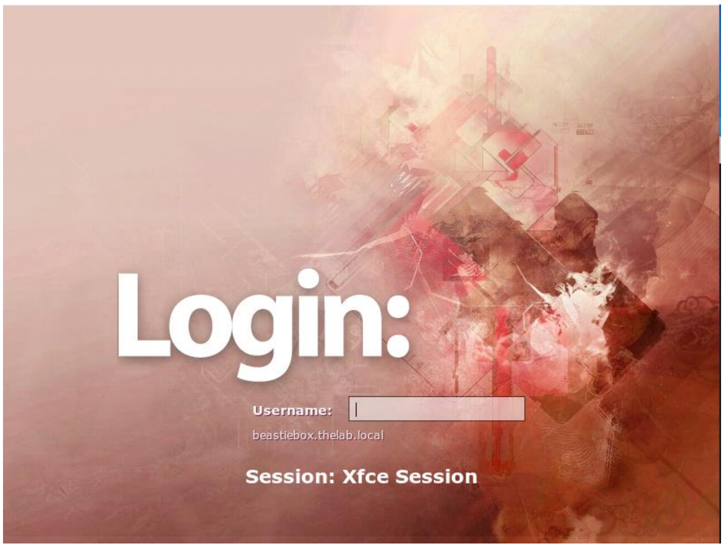
```
$ su -
```

```
# pkg install slim
```

Add the following line (using ee) to **/etc/rc.conf**:

```
slim_enable="YES"
```

After rebooting, the following login screen will appear:



Session: Xfce Session

Enabling Shutdown from the GUI

While the system may always be shutdown using the shutdown command from the terminal (as root), it maybe a good idea to allow any logged in user to shutdown the system using the shutdown menu of XFCE. These options are greyed out at the moment, since a standard user does not normally have the rights to shutdown FreeBSD. To enable the shutdown menu, some rules need to be added to policy kit:

```
$ su -
# cd /usr/local/etc/polkit-1/rules.d
# fetch http://www.schoolspace.gr/files/shutdown.rules
```

It is easy to understand how this works, when the contents of the file are examined. The instructions for creating this file are also shown as a message during the installation of **xfce4-session** package, but since these scroll fairly quickly they are rather easy to miss. However, they can be viewed again:

```
# pkg info –D xfce4-session
```

After restarting your system, the graphical shutdown / restart options of XFCE will be enabled.

Switching Keyboard Layouts

Graphic environments and operating systems need to cater for people all around the world. In many cases, been able to switch to different layouts – in order to type in a different language – is essential for everyday desktop use. Depending on the desktop environment installed, this can be achieved in various ways:

- By using a graphic tool to configure alternate keyboard layouts and layout switching hotkeys. XFCE provides such a tool in the Keyboard settings dialog.
- By configuring appropriate settings in **xorg.conf**. These work independently of any desktop installed and can work with light window managers that don't provide graphical configuration tools.
- By running a script from the terminal, either manually after Xorg startup or by using some autostart feature of the desktop environment.

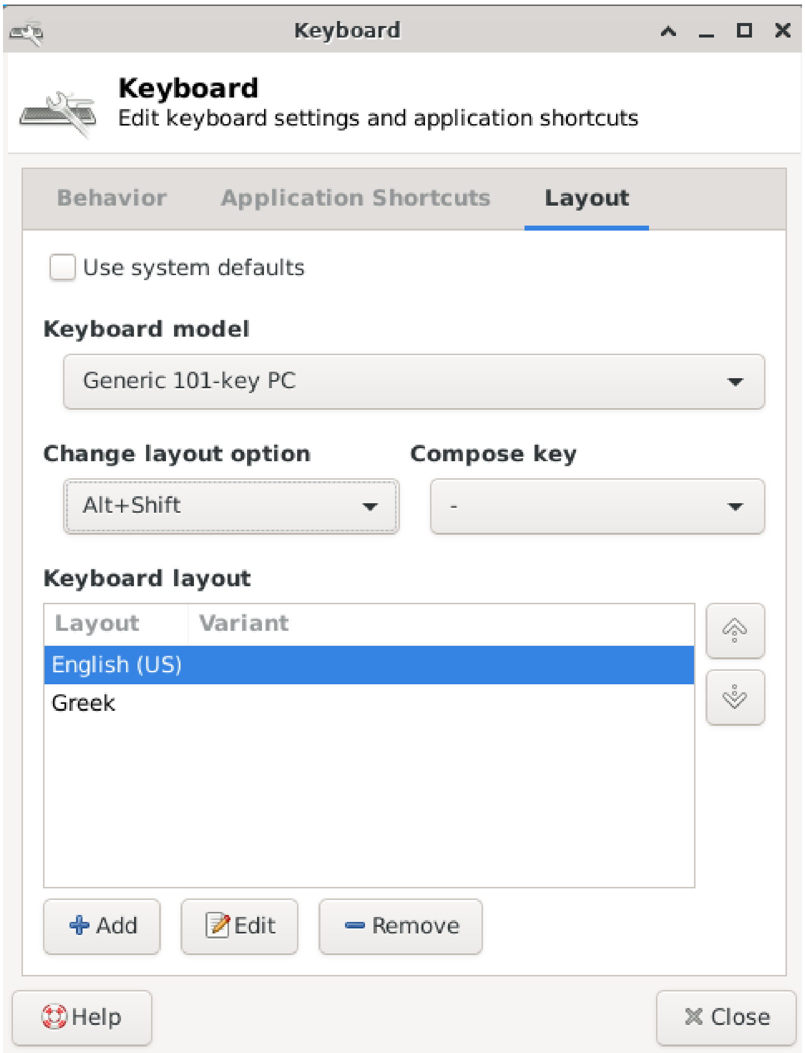
We will be showing all three methods here. The ready-made image provided uses the second method.

Switching Layouts Using a Graphical Configuration Tool

In the XFCE menu:

Applications ➔ Settings ➔ Keyboard

Configure the settings like the dialog below:



Keyboard
Edit keyboard settings and application shortcuts

□ Use system defaults

Keyboard model: Generic 101-key PC

Change layout option: Alt+Shift

Compose key: -

Keyboard layout

Layout	Variant
English (US)	
Greek	

Add the required keyboard layouts depending on language. Greek is used as an example. Typically, switching layouts is performed by **Alt+Shift** keys but other combinations are also possible.

Layout Switching via Xorg configuration

This is the “traditional” way of keyboard layout switching. For the current Xorg version, do the following:

```
$ su –
# cd /usr/local/etc/X11/xorg.conf.d
# fetch http://www.schoolspace.gr/files/kbd-layout-multi.conf
```

Check the contents of the above file to see how layout switching works. This file is automatically loaded during Xorg startup and will work with any desktop environment or windows manager. You need to restart Xorg for the change to take effect (a simple graphical logout / login should be sufficient). Edit the file to adjust the settings for your layout (the one provided switches between Greek and English).

Layout Switching via a Script

The last way to perform keyboard layout switching is by running a script from the graphic terminal, after the desktop environment is initialized. In many desktop environments it is even possible to execute this script automatically at startup. To get the script:

```
$ cd ~
$ mkdir bin
$ cd bin
$ fetch http://www.schoolspace.gr/files/setgrkeyb
```

To execute the script:

```
$ bin/setgrkeyb
```

Edit the script to adjust for you desired layout (the one provided switches between Greek and English).

Installing More Applications (Browsers, Media Players etc)

Additional GUI programs like browsers, media players and productivity software can be installed very easily. Since we are now using a graphic desktop, it is easier to use the XFCE terminal (**Applications Menu ➔ Terminal Emulator**) and run the following commands:

```
$ su -
# pkg install firefox mplayer vlc
```

(Instead of using **su** , you may also use **sudo** as a standard user: **sudo pkg install** etc)

If you intend to work with documents (word processing, spreadsheets etc), a popular application is libreoffice:

```
# pkg install libreoffice
```

Updating the Installed Applications

Every now and then it is a good idea to execute the following commands:

```
# pkg update
# pkg upgrade
```

In this way, all the application packages will be kept up to date. After upgrading some packages, sometimes it is necessary to execute a few extra steps. These are shown as messages after the installation, but may also be viewed again by typing:

```
# pkg info –D <pkg_name>
```

To find more applications for installation, you may search the package database using the command:

```
# pkg search <part_of_package_name>
```

For more information and to better understand this guide, please read the official FreeBSD Handbook:

https://www.freebsd.org/doc/handbook/index.html

Useful info is also available in the FreeBSD Wiki:

https://wiki.freebsd.org/

Please send your comments / suggestions / corrections to **sonic2000gr@gmail.com**