



Basic Arch Linux Tutorial



A GUEST



JUL 27TH, 2015



420



NEVER



Not a member of Pastebin yet? [Sign Up](#), it unlocks many cool features!

text 7.31 KB

raw

download

report

```
1.  ====
2.  Part I: A Barebones Install, Serving a Whole System
3.  ====
4.
5.  ## There will be a list of commands that you will be using a lot at the end of this, as well as a briefing on some of the software
   you will likely be using.
6.
7.
8.  $wifi-menu
9.  # UI to connect to wifi
10.
11. $ping -c 5 google.com
12. # ping, count, site
13.
14. $lsblk
```

```
15. # list storage devices
16.
17. $cfdisk /dev/sda
18. # format tool, drive
19. INSTRUCTIONS: Delete partitions, create a primary root partition first, then use remaining space to create a swap partition. Flag
    root as bootable, change swap to type:linux-swapsomething
20.
21. $ mkfs.ext4 /dev/sda1
22. # format root partition with ext4
23.
24. $ mkswap /dev/sda2
25. $ swapon /dev/sda2
26. # use swap partition
27.
28. $ mount /dev/sda1 /mnt
29. # mount root partition as /mnt
30.
31. $ pacstrap -i /mnt base base-devel
32. # install base system
33.
34. $ genfstab -U -p /mnt >> /mnt/etc/fstab
35. # u need it idk why enter it carefully
36.
37. $ arch-chroot /mnt
38. # entering your new install as root before initial boot
39.
40. $ nano /etc/locale.gen
41. # edit language file
```

```
42. INSTRUCTIONS: Uncomment any languages you want to use by deleting the '#' sign. In my case, " en_US.UTF-8". Nano uses the shortcut
    CTRL+X to exit, y to save during exit, and enter to confirm the filename.
43.
44. $ locale-gen
45. # generate locale file
46.
47. $ echo LANG=en_US.UTF-8 > /etc/locale.conf
48. $ export LANG=en_US.UTF-8
49. # does shit
50.
51. $ ls /usr/share/zoneinfo/
52. $ ls /usr/share/zoneinfo/
53. # lists countries and timezones
54. # choose one in the below syntax
55.
56. $ ln -s /usr/share/zoneinfo/America/New_York > /etc/localtime
57. # set time zone as eastern US
58.
59. $ hwclock --systohc --utc
60. # configure the hardware clock to UTC
61.
62. $ nano /etc/pacman.conf
63. # opens pacman package manager's list of software repositories
64. INSTRUCTIONS, uncomment the [multilib] repository, add the following repository in consistent format:
65. [archlinuxfr]
66. SigLevel = Never
67. Server = http://repo.archlinux.fr/$arch
68.
```

```
69. $ pacman -Sy
70. # updates your repository list
71.
72. $ echo your-host-name > /etc/hostname
73. # changes your computer hostname as desired
74.
75. $ passwd
76. # changes your root password
77.
78. $ useradd -m -g users -G wheel,storage,power -s /bin/bash your-user-name
79. # adds a user
80.
81. $ passwd your-user-name
82. # edits the user's password
83.
84. $ pacman -S sudo
85. # installs the ability to get superuser privileges via the sudo command
86.
87. $ EDITOR=nano visudo
88. # opens a file that you need to edit (lel
89. INSTRUCTIONS: Uncomment the line "%wheel ALL=(ALL) ALL".
90.
91. $ pacman -S grub
92. $ grub-install --target=i386-pc --recheck /dev/sda
93. # downloads grub bootloader, installs grub to the selected hard drive
94.
95. $ pacman -S os-prober
96. # package that has grub check for other oses, recommended.
```

```
97.
98. $ grub-mkconfig -o /boot/grub/grub.cfg
99. # adds arch to grub's boot list
100.
101. $ ip link
102. $ systemctl enable dhcpcd@NAME-OF-THE-ETHERNET-NETWORK.service
103. #lists network adapters (e.g. 'eno1', 'lo'), enables network service (the intranetz) on said adapter
104.
105. $ pacman -Syu
106. # make sure all repositories and packages are up to date
107.
108.
109. ## It is at this point in the tutorial that you have a working, barebones installation of Arch. However, you do NOT currently have
any shred of a GUI. To end the installation here (you can do what you want any time), exit the chroot ($exit), unmount your root
partition ($umount -r /mnt), and reboot ($reboot)
110.
111. ====
112. PART II: A working GUI, Ready for a Desktop Environment
113. ====
114.
115. $ pacman -S xorg-server xorg-server-utils xorg-xinit xorg-twm xorg-xclock xterm
116. # installs xorg, the base of your GUI
117.
118. $ pacman -S mesa
119. # basic graphical drivers, choose option 1 for ATI/AMD and option 2 for nVidia.
120.
121. $ pacman -S bash-completion yaourt
122. # installs a couple of utilities you want and need
```

```
123.  
124. [OPTION 1]  
125. $ pacman -S xf86-video-ati  
126. # installs open source AMD drivers, not recommended for gaming  
127.  
128. [OPTION 2]  
129. $ pacman -S xf86-video-intel  
130. # installs open source Intel drivers  
131.  
132. $ pacman -S lightdm lightdm-gtk-greeter  
133. # lightdm is a display manager, basically a launcher that allows you to login and choose your distro  
134. # there are others, but they are shit  
135.  
136. $ pacman -S a52dec faac faad2 flac jasper lame libdca libdv libmad libmpeg2 libtheora libvorbis libxv wavpack x264 xvidcore  
    gstreamer0.10-plugins  
137. # installs media codecs which you will most likely require for daily use  
138.  
139.  
140. ## Now you have a working Arch install, entirely ready to accept the desktop environment or window manager (advanced) of your choice!  
    You can end the install here and $reboot (after leaving chroot and unmounting) if you think you're ready to go out on your own. If  
    you do, be sure to enable your new display manager with "$systemctl enable lightdm.service" so that you can see it when you reboot!  
    Or do it later to continue booting into the shell.  
141.  
142. ====  
143. STEP III: Choosing a Desktop Environment  
144. ====  
145.  
146. Possibly the simplest step of all, usually done by "$sudo pacman -S desktop-environment"
```

147.
148. Here is your link to any DE you could want: https://wiki.archlinux.org/index.php/Desktop_environment#Officially_supported
149.
150. Just install one you like with "\$pacman -S" and select it from the display manager when you boot up!
151.
152. You may want to install a graphical network manager if you need to do any configuring, otherwise dhcpcd will work just fine!
153.
154. [INSTALL NETWORK MANAGER]
155. \$ pacman -S networkmanager
156. \$ systemctl disable dhcpcd.service
157. # install a network manager, disable dhcp (important!)
158.
159. ====
160. STEP IV: Installing Software
161. ====
162.
163. ## Note that this section is *entirely* based on my opinions, and installs what I believe to be a good set of software to get started.
164.
165. \$ sudo pacman -S deluge chromium iceweasel terminator gvim gimp vlc brasero
166. # installs a basic set of software for daily use
167. # iceweasel and chromium are both browsers, Firefox and Chrome respectively, and thus you only really need one -- you can still uninstall them later.
168.
169. ## Other pacman packages you may want are skype, libreoffice, and steam.
170.
171.
172.

```
173. =====
174. Installing Software, Basic Maintenance, Conclusion
175. =====
176.
177. [YAOURT AND THE AUR]
178.
179. ## Yaourt may be the most powerful program on Arch. It allows you to install packages from the Arch User Repository: a community
    maintained database of software designed/repackaged specifically for use with Arch Linux. Yaourt allows you to quickly and easily
    search for and install these packages.
180.
181. $ yaourt package-name
182. # searches for an AUR package, such as Chrome or Teamviewer, installing is self-explanatory
183.
184. $ yaourt -Syua
185. # updates the AUR and your AUR packages
186.
187. ## Search the AUR: https://aur.archlinux.org/
188.
189. [PACMAN]
190.
191. ## Pacman manages and installs software (and your OS!) from the official open source Arch repositories, as well as any custom
    repositories you might choose to add to the pacman.conf file.
192.
193. $ pacman -S package
194. # installs package
195.
196. $ pacman -Rns package
197. # removes a package and all of its dependencies
```



```
198.  
199. $ pacman -R package  
200. # removes a package  
201.  
202. $ pacman -Sy  
203. # updates pacman's repositories  
204.  
205. $ pacman -Syu  
206. # updates pacman's repositories and installed packages  
207.  
208. [ROOT]  
209.  
210. $ su  
211. # enables root permissions for a given console session  
212.  
213. $ sudo your-command  
214. #enables root permissions for a command  
215.  
216. [BASIC]  
217.  
218. $ sudo nano /directory/file/  
219. # edits a text file  
220.  
221. $ cd /directory/  
222. # moves your terminal's working folder to a directory  
223.  
224. $ ./program  
225. # executes a program in the current directory
```

```
226.  
227. [EVERYTHING ELSE]  
228. wiki.archlinux.org  
229. noob2noobarchhelp@gmail.com  
230. # :)
```

RAW Paste Data

```
====  
Part I: A Barebones Install, Serving a Whole System  
====  
  
## There will be a list of commands that you will be using a lot at the end of this, as well as a briefing on some of the  
software you will likely be using.  
  
$wifi-menu
```



[create new paste](#) / [deals](#) ^{new!} / [syntax languages](#) / [archive](#) / [faq](#) / [tools](#) / [night mode](#) / [api](#) / [scraping api](#)
[privacy statement](#) / [cookies policy](#) / [terms of service](#) / [security disclosure](#) / [dmca](#) / [contact](#)

Dedicated Server Hosting [by Steadfast](#)