

The Arch Linux FAQ

Dennis "Gyroplast" Herbrich
<herbrich@veloxis.de>

Mon Nov 3 18:19:24 2003

Abstract

This FAQ document covers all areas of the ARCHLINUX distribution, and mostly contains distilled information from the official user forum, personal experiences, and more or less general issues. It contains an overview over the distribution itself and it's paradigms, as well as a troubleshooting section.

Contents

1	INTRODUCTION	2
1.1	Copyright	2
1.2	Where to get the latest version of this FAQ	2
1.3	Where to find more information	2
1.4	About ARCHLINUX	2
2	INSTALLATION	3
2.1	CD Install	3
2.2	FTP Install	4
2.3	General	4
3	CONFIGURATION	5
3.1	BootUp / RunCom and INIT Structure	5
3.2	Modules	5
3.3	IDE/SCSI Devices	6
3.4	Networking	7
3.5	Multimedia	7
3.6	XFree86	8

4	PACMAN	8
4.1	Usage	8
4.2	Problems	10
5	ARCH BUILD SYSTEM	11
5.1	General	11
5.2	Building Packages	11
6	MISCELLANEOUS	12

1 INTRODUCTION

1.1 Copyright

The ARCHLINUX FAQ is copyright (C) 2002 by Dennis Herbrich.

This document may be reproduced or distributed in any form without my prior permission. Modified versions of this document, including translations into other languages, may be freely distributed, provided that they are clearly identified as such, and this copyright is included intact.

This document may be included in commercial distributions without my prior consent. While it is not required, I would like to be informed of such usage. If you intend to incorporate this document in a published work, please contact me to make sure you have the latest available version.

This document is provided “AS IS”, with no express or implied warranties. Use the information in this document at your own risk.

1.2 Where to get the latest version of this FAQ

The latest version of this document can be found at
<http://archlinux.veloxis.de/faq/>
in various formats, including the original L^AT_EX file.

1.3 Where to find more information

Your first trip should lead you to <http://www.archlinux.org>, the official home page of the ARCHLINUX distribution. There you’ll find news, the list of currently available packages, download and installation instructions and a useful user forum. Should that alone be not enough for you, try contacting the developers on IRC (irc.freenode.net), channel #archlinux, or sign up for the mailing list by sending an email with “subscribe” as the subject to arch-request@archlinux.org.

1.4 About ArchLinux

[Blatantly ripped from the official homepage]

Arch Linux is an i686-optimized linux distribution that was originally based on ideas from CRUX, a great distribution developed by Per Lidén.

Arch is fast, lightweight, flexible and simple. Those aren’t very fancy buzzwords but they’re all true. Arch is optimized for the i686 processor, so you get more for your cpu cycle. It’s lightweight compared to RedHat et al,

and its simple design makes it easy to extend and mold into whatever kind of system you're building.

This is backed by an easy-to-use binary package system that doesn't get over-complicated (I can almost write the whole thing in bash) and a simple build system that will hopefully encourage users to build their own packages (Arch Build System). Rebuilding your entire system is as simple as running 'makeworld'.

Arch Linux strives to maintain the latest stable version of its software. We currently support a fairly streamlined package set with a growing collection of contrib packages made by myself and other AL developers. A brief component list is below.

In its goal to be simple, flexible, bla bla bla, I've adopted Per's ideology and have left out the relatively useless portions of a linux system, things like /usr/doc and the info pages. In my own personal experience, these are rarely used, and the equivalent information can be obtained from the net if need be. Manpages all the way...

Arch Linux also strives to use some of the newer features that are available to linux users, such as ext3/reiserfs and devfs support.

2 INSTALLATION

This section deals with various questions that tend to crop up during or before installing the ARCHLINUX distribution. The FAQ section covers only the potential pitfalls during installation, and is by no means intended as a detailed installation guide. If you expected that, go elsewhere¹.

2.1 CD Install

Questions specific for the CD based installation process are answered here.

Installation CD boot fails with "kernel panic: vfs: unable to mount root fs"

Obviously there's a problem reading the kernel image from the cdrom. If you have two cdroms in your machine, try running the install from the other cdrom. Alternatively create a boot floppy with the *boot.img* file from the CD, boot from it, and supply a suitable "root" parameter to the kernel to point to the correct CD-ROM drive containing the pesky CD. Details can be found in the forum here ².

¹<http://www.archlinux.org/archdoc.html>

²<http://bbs.archlinux.org/viewtopic.php?t=6>

I get CD read errors!

Install from a CD that is NOT defective.

2.2 FTP Install

Questions and Answers regarding the FTP installation procedure.

2.3 General

More generic installation questions are answered in this section.

How does the DevFS naming scheme work?

In a nutshell, the DevFS paths reflect the physical layout of your IDE bus, ie. you (usually) have two buses on one host, with 2 devices (targets) each, for a total of four IDE devices. In the case of HDDs, further down the path you'll be confronted with device entries for each partition as well, slightly similar to the old `/dev/hda*` scheme. For example, the first primary partition on the primary master IDE drive would be `/dev/hda1` in the old way, and `/dev/ide/host0/bus0/target0/lun0/part1` in the DevFS scheme. Similarly, the old `/dev/hdb5` would be `/dev/ide/host0/bus0/target1/lun0/part5`. Albeit a lot more to type, it's surely a more structured approach of naming devices. A more detailed explanation of DevFS can be found here³, in a FAQ maintained by the DevFS creator. Should be official enough.

My freshly installed ArchLinux fails to boot! Root FS fails to be mounted!

You boot your system, and after initializing a couple of hardware drivers your bootup freezes with an error message similar to this one:

Kernel Panic : VFS unable to mount root FS on 03:02

No big deal, you just screwed up your bootloader configuration, and LILO isn't quite able to mount your designated root filesystem. This often happens after finishing the install and not checking the initial `lilo.conf` thoroughly for correctness. To remedy this ugly situation, boot the kernel with the additional "root" parameter, pointing to the correct root partition. If your root partition is the first logical partition on your primary master IDE HDD for instance, you would enter `arch root="/dev/ide/host0/bus0/target0/lun0/part5"` to boot correctly into your system. Alternatively you may of course use a

³<http://www.atnf.csiro.au/people/rgooch/linux/docs/devfs.html>

rescue disc to accomplish the same thing. After getting a working system at your fingertips, edit the `lilo.conf` to reflect your setup, remember to rerun `lilo`, and there you go.

Check this ⁴ forum article for details.

3 CONFIGURATION

This section covers questions and problems with configuring your ARCHLINUX system and it's packages.

3.1 BootUp / RunCom and INIT Structure

Information on how to handle the rc scripts and the init process is found here.

The system hangs after starting a daemon I added to the rc.local script.

The `/etc/rc.local` script does not automatically run any commands in the background. The daemon that is seemingly locking up your boot process is simply running in the foreground, so bootup won't continue until it exits. Press CTRL-C or whatever you need to kill this process to get to your login. If the process simply won't be killed that way, start Linux with the boot parameter `"init=/bin/bash"` to bypass the rc.local (and pretty much everything else as well) completely. After login, mount the root-fs read-/writable with the command `"mount -o remount,rw /"`, to be able to tinker with the configuration, then make sure you review the command line used to start the offending daemon and add an ampersand or appropriate option to force this daemon into the background.

3.2 Modules

Getting modules to be loaded on demand can be a hassle sometimes. Let this section give you some pointers on how to do it in ARCHLINUX.

Where can I assign modules to certain hardware?

As usual in the `/etc/modules.conf` file. See *man modules.conf* for details, there's nothing special here in ARCHLINUX.

⁴<http://bbs.archlinux.org/viewtopic.php?t=128>

How does module autoloading work with DevFS?

Different. By default there's a line `LOOKUP .* MODLOAD` in your `/etc/devfsd.conf` that allows autoloading for all modules you might specify in `/etc/modules.conf`. If it's missing, you might run into strange problems here. In case you're wanting to limit autoloading to certain devices, I suggest reading the excellent DevFS FAQ⁵.

How can I load a module on bootup without using modules.conf?

Add the module name to the `MODULES` array in `/etc/rc.conf`, and it will be loaded automatically at bootup without the need to bind any specific hardware to it with the `alias` command in `modules.conf`. This is a good place for the `nvidia` kernel module, for example.

3.3 IDE/SCSI Devices

Problems that may arise with IDE and SCSI devices are handled here.

How do I install my CD/DVD-Writer?

First off, you do *not* have to build a custom kernel in most cases. If you have a SCSI model, things should work for you out of the box (untested! please confirm somebody!) if you can already use the device at all. Install the "cdrtools" package and run `cdrecord -scanbus` to show the installed SCSI devices. If your burner does not show up, double check the hardware for missing terminators or duplicate LUNs. Maybe your SCSI BIOS needs a nudge or two. Diagnosing this, however, is out of the scope of this document. If you are using an IDE burner, you'll first have to enable SCSI emulation for this device. The stock kernel has the necessary options compiled as modules already, so all you have to do is to add "ide-scsi" to the `MODULES` array in `/etc/rc.conf` and add the line "append="hdc=ide-scsi"" to the image section of your OS in the `/etc/lilo.conf`. The latter option supplies a boot parameter to the kernel and instructs it to not bind the designated device ("hdc" is only an example, and must really be the device name of YOUR IDE burner, of course) to the `ide-cd` driver, but rather to the `ide-scsi` driver. The first option then loads the needed `ide-scsi` module during bootup, which should find the still unbound CD-Writer and take care of it. Testing can then be done with `cdrecord -scanbus` to see whether your burner shows up. You should also have the appropriate entries below `/dev/scsi/` if all went well loading the module. If you compiled your own kernel, read the CDWriting-HOWTO and find out what you did wrong.

⁵<http://www.atnf.csiro.au/people/rgooch/linux/docs/devfs.html>

3.4 Networking

Anything about getting your machines to connect to a network.

Where do I set my machines' hostname?

Right there, in the */etc/rc.conf* file. Look for the HOSTNAME entry, and change it to your liking. Do not add the domain part.

How do I configure my ethernet card?

As a first step, make sure your NIC is supported by the kernel, and all modules are loaded if any are needed. For further details on how to load modules on the fly or during boot, check the modules section of this FAQ. If you can manually run *ifconfig eth0 up* without complains, adding the bootup configuration is a breeze. Simply add a line similar to *eth0="eth0 10.0.0.1 netmask 255.255.255.0 broadcast 10.0.0.255"* to */etc/rc.conf* (using your IPs, of course), and list the card in the INTERFACES array, to let the script configure it at boot time. If you want to deactivate a card, but not remove it's configuration entirely, simply prepend a ! to the card name in the INTERFACES array, like this: *INTERFACES=(lo eth0 eth1 !eth2)*.

What do I need to get DHCP to work?

Not much. Basically all you need is a DHCP client daemon package to be able to query a DHCP server, and tell the bootup script to configure certain network devices with this method. The client package is called *dhcpcd*, found in the *current* tree. To configure *eth0* via DHCP, all you need is the line *eth0="dhcp"* in the */etc/rc.conf* file, as well as a listed *eth0* in the INTERFACES array, of course.

I cannot SSH into my machine!

By default, ARCHLINUX denies all remote access. Edit your */etc/hosts.deny* file and remove the entry. Check the appropriate manpages for more information.

3.5 Multimedia

Issues with sound, video and the like.

Only root can access /dev/dsp!

The file system permissions on the audio devices are quite restrictive by default. You should consider creating a group named "audio" and add these lines to your */etc/devfsd.conf*:

```
# Allow access to sound devices for group "audio"
REGISTER      ^sound/.*      PERMISSIONS      root.audio 0660
```

This will give read/write permissions to the sound devices for root and members of the "audio" group. Simply add any users that need sound to this group. Do not forget HUPing the devfs-daemon to let these changes take effect.

3.6 XFree86

Configuration of programs using X and X itself.

How can I enable scrolling with the wheel of my mouse?

In most cases you only need to add *Option "ZAxisMapping" "4 5"* to the InputDevice section of your XF86Config. If that does not do the trick, try this⁶ link for more resources and instructions.

Some of my X programs (xchat) fail to find any fonts and quit!

You upgraded to XFree 4.3.0 or higher. Run *fc-cache* as root to (re)build the font cache, and things should be fine.

I have problems installing the NVidia driver!

I'd recommend getting the latest self-installing driver package from <http://www.nvidia.com> and read the official README thoroughly as most problems are handled in it. Basically you only need to run the installer, and change the driver in the Device section of your XF86Config from the 2D-only XFree-driver "nv" to "nvidia". For autoloading of the module you might want to add "alias char-major-195 nvidia" to your */etc/modules.conf* or add "modprobe nvidia" to your rc.local file. A successful installation is proven by the appearance of the NVidia logo on X startup (disable it with *Option "NoLogo" 1*) and successfully running *glxinfo* and *glxgears*.

4 PACMAN

This section covers the usage, possibilities and problems with pacman.

4.1 Usage

The most common ways to invoke pacman to finish everyday's tasks.

⁶<http://koala.ilog.fr/colas/mouse-wheel-scroll/>

How do I get access to the unofficial packages?

Open `/etc/pacman.conf` in an editor and remove the hash from one or two close mirrors in the [unofficial] block. The same procedure applies to unstable, current or any other trees.

How do I refresh the list of available packages?

Run `pacman -Sy` as root.

What packages are available?

One possibility is to check the package list on the official homepage⁷ for anything you might want. You could also run `abs` as root to create the "ports tree" of Arch in `/usr/abs` and browse/search locally on your HDD. Another way to search the database is using `pacman -Ss searchstring`, which looks for packages containing the searchstring in their description or name.

How do I keep my system up to date?

Run `pacman -Syu` at regular intervals. Please make sure that pacman is always updated *first* by running `pacman -Sy pacman` before upgrading any other packages to rule out any possible incompatibilities.

Which package does file "foo" belong to?

Run `pacman -Qo foo`.

My /var directory is getting bigger and bigger!

This is probably caused by a growing package cache. Check the size of `/var/cache/pacman` and use `pacman -Sc` to delete all cached packages if you need to. You may also just delete files manually, there is no index file or anything fancy that could be corrupted (so far). In the same way, you can remove the source packages downloaded by makepkg with `makepkg -C`.

I removed a package, but there are still some config files left!

Use the purge option of pacman to get rid of these files, too. It is called "nosave", and prevents the creation of config file backups.

How do I install a package I've already downloaded or built myself?

By using the `-p` option, as in `pacman -Up packagefile`.

⁷<http://www.archlinux.org>

Can I remove a package and everything that depends on it?

Yep! The "cascade" option is what you need. Just don't accidentally remove glibc or something similar with this option.

Pacman overwrote my config files!

You should really use the NoUpgrade option in */etc/pacman.conf* to make pacman not touch any config files you have altered. Make sure the *pacman.conf* itself is on the list, too. Useful files to add are:

- `etc/modules.conf` `etc/lilo.conf`
- `etc/rc.conf` `etc/rc.local` `etc/pacman.conf`
- `etc/fstab` `etc/passwd` `etc/shadow` `etc/group`
- `etc/raidtab` `etc/X11/XF86Config` `etc/resolv.conf`
- `etc/hosts` `etc/HOSTNAME`

as well as any config files you ever touch. That is the safest approach.

How do I create my own, local repository?

Read the comments in the */etc/pacman.conf* file, near the bottom. Hint: The first argument to "gensync" must be the directory containing the sub-directories named after the packages with the PKGBUILD files, whilst the second argument is the filename of the package database, which must lie in your repository directory. Use "first source dir, then target dir" as a rule of thumb. Copy your generated packages into the target dir, and after syncing pacman you can transparently use your local repository.

How do you know all this?

man pacman. Really.

4.2 Problems

Troubleshooting more or less annoying pacman errors or strange behaviour.

I want to upgrade the kernel, but the file cannot be found on the server!

Edit */etc/pacman.conf* and make sure only the official mirror, `ftp.archlinux.org`, is not commented. The kernel package seems to not be stored on mirrors.

After installing packages, files are missing or the procedure spat out "weird" error messages

It is *really important* to update pacman itself before installing any new packages. The packaging system is developed aggressively, thus sometimes adding new, "incompatible" packages or functionality which an old pacman version won't be able to handle. When in doubt, install the current pacman version and reinstall the packages causing the problems again.

5 ARCH BUILD SYSTEM

Questions on using the Arch Build System and it's purpose are to be found here.

5.1 General

Miscellaneous questions on handling and usage of the ABS is covered in this section.

5.2 Building Packages

What's there to know about rolling your own?

How do I build my own custom kernel package?

First off, get yourself the current ABS tree by running *abs* as root. Move to the */usr/abs/base/kernel* directory, and find the *config* file therein. This file is used to configure the kernel when running *makepkg*, and you can/should modify it to your liking, either by editing it directly with a text editor (only if you REALLY know what you're doing), or by using whatever other favourite kernel configuration tool you use. A common way probably would be to download the current kernel, unpack it, and run trusty old *make menuconfig* in the kernel source directory. After that, load the *config* file supplied by ABS, modify stuff, save the config back to a new *config* file, and exchange the two. You get the idea. After you've (somehow) created a configuration file suitable for your needs, run *makepkg* as usual, and grab yourself a cup of coffee while you wait. In the end you should find a nicely compiled kernel package, which can be used with pacman like every other package.

Now, wasn't *that* easy?

As usual, check the forum article ⁸ for more information.

⁸<http://bbs.archlinux.org/viewtopic.php?t=149>

6 MISCELLANEOUS

Anything else can be found in this category, be it meta-faqs (faqs about the FAQ), slightly off topic but still important questions, or simply hard to categorize topics.

My manpages have "ESC" and other weird stuff all over them!

You need to have the LESS environment variable set to "-R" to display the manpages correctly. If you are using the latest bash package then LESS should be set in your /etc/profile with *export LESS="-R"*.

My machine does not power off automatically.

Make sure you have APM enabled in your BIOS. Then add "apm=power-off" to the bootup options of the kernel, by using the "append" option of LILO or whatever your bootmanager offers. Reboot, and use "halt -p" or "poweroff" to power off the machine. This works (for me) with the stock kernel.

Is there a "LiveCD" available like KNOPPIX?

Yes! There is an unofficial Project of the AMLUG which distributes a LiveCD basing on ARCHLINUX. More information can be found on their page at

<http://amlug.org/new-projects/live-cd/al-amlug-live-cd.html>.