

Hentzenwerke Whitepaper Series

Managing Linux Services with chkconfig

By Whil Hentzen

The Linux command, `chkconfig`, allows you to configure the automatic start and stop of services in the `/etc/rc.d/init.d` scripts via a command line interface. Here's a ten minute guide to using `chkconfig`.

1. Preface

1.1 Copyright

Copyright 2006 Whil Hentzen. Some rights reserved. This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs License, which basically means that you can copy, distribute, and display only unaltered copies of this work, but in return, you must give the original author credit, you may not distribute the work for commercial gain, nor create derivative works based on it without first licensing those rights from the author. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/2.0/>.

1.2 Revisions

1.2.1 History

Version	Date	Synopsis	Author
1.0.0	2006/11/17	Original	WH

1.2.2 New version

The newest version of this document will be found at www.hentzenwerke.com.

1.2.3 Feedback and corrections

If you have questions, comments, or corrections about this document, please feel free to email me at 'articles@hentzenwerke.com'. I also welcome suggestions for passages you find unclear.

1.3 References and acknowledgments

Thanks to the folks on the Milwaukee Linux User Group list (www.milwaukeeelug.org), particularly Dale Noll, whose "Hardening a new installation" presentation was the genesis of this article, and who graciously reviewed the final version.

1.4 Disclaimer

No warranty! This material is provided as is, with no warranty of fitness for any particular purpose. Use the concepts, examples and other content at your own risk. There may be errors and inaccuracies that in some configurations may be damaging to your system. The author(s) disavows all liability for the contents of this document.

Before making any changes to your system, ensure that you have backups and other resources to restore the system to its state before making those changes.

All copyrights are held by their respective owners, unless specifically noted otherwise. Use of a term in this document should not be regarded as affecting the validity of any trademark or service mark. Naming of particular products or brands should not be seen as endorsements.

1.5 Prerequisites

This document was written using Fedora Core 6 as the test machine. If you're not familiar with run levels, a brief review is provided in Section 6.

2. Why the need for chkconfig

One of the techniques in every Linux administrator's toolbox to improve security of a box is to turn off unneeded services. It's infinitely more difficult to take advantage of an exploit in a service if that service isn't running. To be more secure, many administrators will boot a new machine in run level 1 (passing the 'single' parameter to the boot loader), even before connecting it to the network, and turning off unneeded services immediately.

For those used to using the GUI in run level 5 to manage services, this begs the question – how to manage services if the only interface available is a shell prompt? Enter chkconfig.

3. Getting to run level 1

My article, Resetting the Root Password in Linux, available at www.hentzenwerke.com, describes how to boot a machine in run level 1, so instead of repeating the those instructions here, please see that article. I'll assume at this point that you know how to pass the 'single' parameter during boot to end up at a prompt like so:

```
sh 3.01#
```

I'll also assume that you know the basics for using the 'vi' text editor to edit a simple text file. Instructions on using vi can be found here:

<http://www.techtutorials.net/tutorials/unix/vi.shtml>
<http://www.eng.hawaii.edu/Tutor/vi.html>

4. chkconfig commands

The 'chkconfig' command allows you to display and change the services running on your machine in all of the run levels. Let's look at the common functions. (You may want to change to /etc/rc.d/init.d now; chkconfig by itself will work without doing so, but other commands won't.)

4.1 Display all services

The 'chkconfig' command all by itself will display all available services in every run level, like so:

```
# chkconfig --list
acpid      0:off  1:off  2:off  3:off  4:on   5:on   6:off
anacron    0:off  1:off  2:on   3:on   4:on   5:on   6:off
apmd       0:off  1:off  2:on   3:on   4:on   5:on   6:off
...
crossfire  0:off  1:off  2:off  3:off  4:off  5:off  6:off
...
yum-updated 0:off  1:off  2:off  3:on   4:on   5:on   6:off
```

(I've truncated this listing for readability's sake.)

4.2 Display running services

Depending on the number of services that your distribution installed, this list may be a bit overwhelming. Since we're interested in turning off unneeded services, we don't need to see services that are already off. The command

```
# chkconfig --list | grep :on
```

will display just those that are turned on in any run level, like so:

```
# chkconfig --list | grep :on
acpid      0:off  1:off  2:off  3:off  4:on   5:on   6:off
anacron    0:off  1:off  2:on   3:on   4:on   5:on   6:off
apmd       0:off  1:off  2:on   3:on   4:on   5:on   6:off
...
yum-updated 0:off  1:off  2:off  3:on   4:on   5:on   6:off
```

Even better, the command

```
# chkconfig --list | grep 3:on
```

will display just those that are turned on in run level 3, like so:

```
# chkconfig --list | grep 3:on
anacron    0:off  1:off  2:on   3:on   4:on   5:on   6:off
apmd       0:off  1:off  2:on   3:on   4:on   5:on   6:off
...
yum-updated 0:off  1:off  2:off  3:on   4:on   5:on   6:off
```

4.3 Removing a service from chkconfig control for all run levels

Now that you know how to look for services that are on or off for a certain run level, how do you get rid of them? The command

```
# chkconfig --del isdn
```

will remove a service from chkconfig control by removing the symlinks in all of the run level directories, which effectively turns the service off.

4.4 Placing a service under chkconfig control for default run levels

What if you selected the wrong one when messing around with "--del"?

```
# chkconfig --add isdn
```

will add the script back in for its default run levels – the ones listed in the 'chkconfig' comment line in the script (see Section 6.3 for details.)

4.5 Turning a service off for just select run levels

There may be times when you want to turn a service on or off for selected run levels, instead of for all of them. The "--level" argument does that. In order to turn isdn off in run level 3 (but leave it running for any other level that it was already on for), try this:

```
# chkconfig --level 3 isdn off
```

You can list multiple levels by jamming them all together, like so:

```
# chkconfig --level 1234 isdn off
```

The "--level ... off" argument is not the same as using "--del". "--del" removes the symlinks from all rcN directories while "--level ... off" creates a Kill script for that service at the specified run level. See Section 6 for details on the internals.

5. Determining if a service is needed

Admit it; even the most hardcore Linux admin is, once in a while, stymied by the cryptic naming of some services. So don't feel bad if you look at this list of running services and wonder what they all do, and, more importantly, whether or not you need them or not.

Some folks will snort and condescendingly advise, "Just read the man page...", the unspoken "idiot" at the end of the sentence. But the man page for a given command isn't always that helpful, particularly if you're new to that command. The man pages are more for reference than for tutorial purposes.

What can be more helpful is looking at the service's script itself – the comments in there may give you more clues than the terseness seemingly required elsewhere.

Suppose you want to determine what the 'isdn' service does. Change into the /etc/rc.d/init.d directory and run the more command, like so:

```
# more isdn
```

The isdn script will be opened up in a file viewer. At the top of the script, you'll see a number of comments, indicating with "#" signs:

```
#!/bin/bash
#
# chkconfig: 2345 9 91
# description: start and stop ISDN services
#
```

The 'description' line, obviously, tells you what the service does. Naturally, some scripts are more verbose than others. But even from this simple blurb, you can probably tell that if you're not using ISDN services (which, if you're connected to the network via dial-up or broadband, you're probably not), you can turn this service off.

Ctrl-C to get out of 'more'.

6. Under the hood: what chkconfig does in /etc

One of the great things about Linux is that it allows you to dig into the guts and find out exactly what's going on. Let's take a look at what chkconfig is doing behind the scenes.

6.1 How /etc/rc.d/init.d scripts work

In order to understand how chkconfig works, it's necessary to review how scripts and run level directories are set up. The scripts to run services are all located in /etc/rc.d/init.d (at least on my Fedora Core 6 system; they may be in some place slightly different on your distribution.)

There are a set of 'run level' directories under /etc named rc0.d, rc1.d, and so on, that map to the various run levels. In order to execute a script (start a service) in a specific run level, a symbolic link to the service's script in /etc/rc.d/init.d is created in the directory of interest. So, for example, if the isdn script is to be run for run levels 4 and 5, but not for 0-3 or 6, a symlink to /etc/rc.d/init.d/isdn is created in both /etc/rc4.d and /etc/rc5.d.

This way, if the script for a service needs to be modified, only the original script in /etc/rc.d/init.d needs to be dealt with. The symlinks for each run level automatically pick up the changes.

The order in which scripts are executed relies on how they're named. The script for starting up the isdn service is named "isdn", but when symlinked to a run level directory, the symlink is named 'S09isdn'; the 'S' meaning the startup script, and the '09' indicates the order in which the script will be run – after S08iptables but before S10network. Similarly, the scripts beginning with the letter 'K' indicate scripts run in order to kill a service. You'll note that the values for kill scripts are generally in reverse order, since the services are shut down in the opposite order that they're started up. If two scripts share the same number (like S08iptables and S08ip6tables), the scripts are executed in alphabetical order.

Simple, yet elegant.

6.2 What chkconfig does

In order to have a script execute for a certain run level, all that's needed is to create a symlink in that run level's 'rc' directory to the original script file. 'chkconfig', by virtue of the arguments passed to it, takes care of this – the '--del' argument or the '--levels', for instance.

6.3 How chkconfig knows what to do

So how does chkconfig know what to do? If you pass the '--add' argument, how does 'chkconfig' know which run levels are the default for the particular script? The chkconfig program looks inside the original script (inside /etc/rc.d/init.d) for the comment line that begins "# chkconfig".

```
# chkconfig: 2345 9 91
```

The first argument in this comment identifies which run levels are the defaults – in this case, 2, 3, 4, and 5. The second argument is the order of the script during startup – in this case, '9'. And the third argument is the order that the service should be run when killed. These arguments are used for naming the symlinks for this script for startup and kill.

7. Where to go for more information

This free whitepaper is published and distributed by Hentzenwerke Publishing, Inc. We have the largest lists of “Moving to Linux”, OpenOffice.org, and Visual FoxPro books on the planet.

We also have oodles of free whitepapers on our website and more are being added regularly. Our Preferred Customer mailing list gets bi-monthly announcements of new whitepapers (and gets discounts on our books, first crack at special deals, and other stuff as we think of it.)

Click on “Your Account” at www.hentzenwerke.com to get on our Preferred Customer list.

If you found this whitepaper helpful, check out these Hentzenwerke Publishing books as well:

**Linux Transfer for Windows® Network Admins:
A roadmap for building a Linux file and print server
Michael Jang**

**Linux Transfer for Windows® Power Users:
Getting started with Linux for the desktop
Whil Hentzen**