

Resetting the Root Password in Linux

By Whil Hentzen

So, Bunky, you forgot your root password, eh? Or maybe you've inherited a machine for which someone else set the password and said password wasn't handed down to you. Or perhaps an administrative mistake left a server hidden away in some back corner with an old password that no one remembers anymore. In any case, if you've got physical access to the machine and a little bit of time, you can probably reset the root password – which in itself is a cautionary tale about security (but that's another article.)

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	2006/10/21	Original	WH
1.0.1	2006/10/31	Minor modifications on passwd, file systems ! mounted, init termination	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 MLUG members Aaron Schrab, Dale Noll, Darrick Hartman, Glenn Holmer, Jason Hecker, Justin Purdy and Will Maier, as well as all sorts of stuff, some of which was even correct, on the Web.

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 SuSE 9.0, SuSE 10.1, and Fedora Core 5.0 and assumes a beginner's familiarity with use of Linux via the GUI and the Command Window.

2. Resetting the Root Password concepts

If you're reading this document, you're likely aware that the Linux 'root' user is equivalent to the Windows 'administrator' user – the account that has complete control over the machine. Every piece of Linux literature on the planet tells the new (and experienced) user to not use the root account for their day-to-day work; instead, create a regular user account and use that. One problem that users then occasionally run into is that, since the root account is infrequently used, the password assigned to the root user is infrequently used as well, and thus prone to being forgotten.

Other reasonable scenarios for losing the root password are also easy to come by. One reviewer of this article recounted the story of how their company would change the root passwords on all of their servers on a regular schedule. One time, one server that had been recently moved was not included in the mass password change, and when it came time to update the passwords again, several months later, that server's password was out of sync with the rest, and no one could remember what the previous password had been. Result: lost password, which meant the server needed to have its password reset.

There are several commonly discussed methods for resetting a root password. As various distributions of Linux are set up differently, a method that works for one distro may not work for another.

3. File of interest to password-resetting folks

There are four files that are of special interest to folks who need to reset the root password.

/etc/passwd

The first is the file that contains the users, /etc/passwd. This is a standard text file with a row for each account on the machine, and values separated by colons. A typical row looks like this:

```
root:x:0:0:root:/root:/bin/bash
```

In olden days, the user's password was located in the second position in the row, where the 'x' is displayed in the line. The 'x' now is a placeholder indicating that the real password is located in a second file.

/etc/shadow

The second file, /etc/shadow, contains the actual passwords, encrypted, of course. Values are again separated by colons. A typical line looks like this:

```
root:$w83j$jfSo83j1LL4usjUf8s83$iWjHF:13388:0:99999:7:::
```

The encrypted password is, again, the second position.

/etc/fstab

While, strictly speaking, not used for password handling, /etc/fstab can be useful for determining what partitions are on the machine and where they are mounted. In some scenarios, this information will be useful.

/boot/grub/menu.lst

The menu.lst file contains the actual commands that the GRUB boot loader executes during startup. A typical menu.lst file (for Fedora Core) looks like this:

```
#boot=/dev/sda
default=0
timeout=5
title Fedora Core (2.6.17-1.2187_FC5smp)
  root (hd0,0)
  kernel /vmlinuz-2.6.17-1.2187_FC5smp ro root=LABEL=/ rhgb
  initrd /initrd-2.6.17-1.2187_FC5smp.img
title Fedora Core (2.6.15-1.2054_FC5smp)
  root (hd0,0)
  kernel /vmlinuz-2.6.15-1.2054_FC5smp ro root=LABEL=/ rhgb
  initrd /initrd-2.6.15-1.2054_FC5smp.img
```

This menu.lst file contains menu choices for two separate kernels that can be selected during boot. The lines that begin with the word "title" identify the kernels. The three lines after each title line are the actual commands that are executed during bootup. We'll come back to the line that starts with "kernel" shortly.

The LILO boot loader has something similar, but I don't use LILO, so I don't have an actual example handy.

4. Commands of interest to password-resetting folks

There are several Linux commands that will be of interest to folks who need to reset the root password.

passwd

The first is the granddaddy of them all, passwd. Note that there is no 'o' or 'r' in the command's name. This command, without any parameters, will reset the root password, like so:

```
root> passwd
Changing password for user root.
New UNIX password:
```

```
Retype new UNIX password:
passwd: all authentication tokens updated successfully.
root>
```

More generically, 'passwd' will change the password for the account that runs it. So if you ran it while logged in as 'herman', without any parameters, it would change herman's password. 'passwd' tries to be intelligent; if you try to use a stupid password, like 'bob' or 'everywhere', you will be informed of your poor choice:

```
BAD PASSWORD: it is too short
BAD PASSWORD: it is based on a dictionary word
```

Note that passwd will still let you make a bad decision; it's just trying to help.

vi

You may need to edit a text file in a place where you have limited resources (i.e. no graphical text editor is anywhere around.) In such a spot, the editor 'vi' will still likely be available. However, many users aren't familiar with using vi, and it's interface is, er, peculiar. Here's the nickel tour.

To edit a file in vi, pass the file name as a parameter:

```
vi /etc/passwd
```

Now you're in for it. vi has two modes: insert (when you're typing) and command (when you're issuing commands to do something, such as save or quit). When you start up vi, you're in command mode. Type a colon (":") to initiate a command, followed by the letter(s) of the command(s), and then press Enter. For example, to write your changes and quit, type

```
:wq
```

and press Enter. To abandon your changes and quit, type

```
:q
```

and press Enter. To switch between modes, type a, i, o, c or s; to go from command to insert, type Esc to go from insert to command.

If this type of interface gives you a sudden thrill, as well it might, you're on your own to search for more exhaustive references on this trusty editor. Although <http://www.techtutorials.net/tutorials/unix/vi.shtml> or <http://www.eng.hawaii.edu/Tutor/vi.html> would be good places to start.

5. Scenarios where a root password can't be reset

A machine can be protected in a number of ways in addition to password protecting the root account. Most machines offer a BIOS-level password, which prevents access to the BIOS without the password. The Linux boot loaders, GRUB and LILO, can both also be set up with passwords, providing a second level of protection. And the contents of filesystems can be encrypted (using yet another password), affording yet another level of protection.

The rest of this article assumes that none of these conditions are applicable for your scenario, or that you know these passwords – it's just the root password that you don't have. Otherwise, you're out of luck – you're going to have to start backing up your data and then fish around for your installation disks.

6. Overview of password resetting methods

This article will discuss three detailed, step-by-step methods for resetting a root password, and then cover, in more general terms, a couple more possibilities. Distributions can be divided into two general types: those that can be booted to a shell prompt, such as to runlevel 1, without needing to know the root password (Fedora Core is one example), and those that require the root password to login even at runlevel 1 (SuSE is one example).

The first method we'll discuss is to boot the machine to runlevel 1, which loads the machine with a shell prompt, logged in as root, with no password necessary. Then use the 'passwd' command to change the root password, hopefully to something a bit more memorable this time. I'll use Fedora Core to illustrate.

The second way, used for machines that require a root password even when booting to runlevel 1, is to boot the machine directly to a shell prompt, which doesn't even make it to where the runlevel 1 init script is executed. The unfortunate side effect of this method is that because the init scripts do not run, the partitions aren't mounted either, (except for the root file

system – which is mounted as read-only.) This means that you can't just run the 'passwd' command, because the disk that the password file is on isn't available for writing to. Thus, this method requires that you remount the partition that contains the /etc directory so that it can be written to, and then make the necessary changes to the root password. I'll use SuSE 9.0 to demonstrate.

The third method is to use a "live-CD" distribution, such as Knoppix, to load an instance of Linux into RAM, mount the filesystem on the hard disk, and, similar to the second method, make the necessary changes to the root password. Other methods discussed in brief are using a distribution's rescue CD and simply reinstalling the operating system on top of itself. I'll use SuSE 9.0 and Knoppix 5.0 to demonstrate this method.

Note that the first and second methods require you to be able to change how the machine boots up via additional parameters. Different distributions use different interfaces and thus require slightly different methods for passing these parameters. I'll discuss these in due course.

7. Boot the machine to runlevel 1 (Fedora Core)

Fedora Core's boot process initially displays a countdown screen that makes you react fairly quickly if you want to interrupt it, as shown in **Figure 1**.

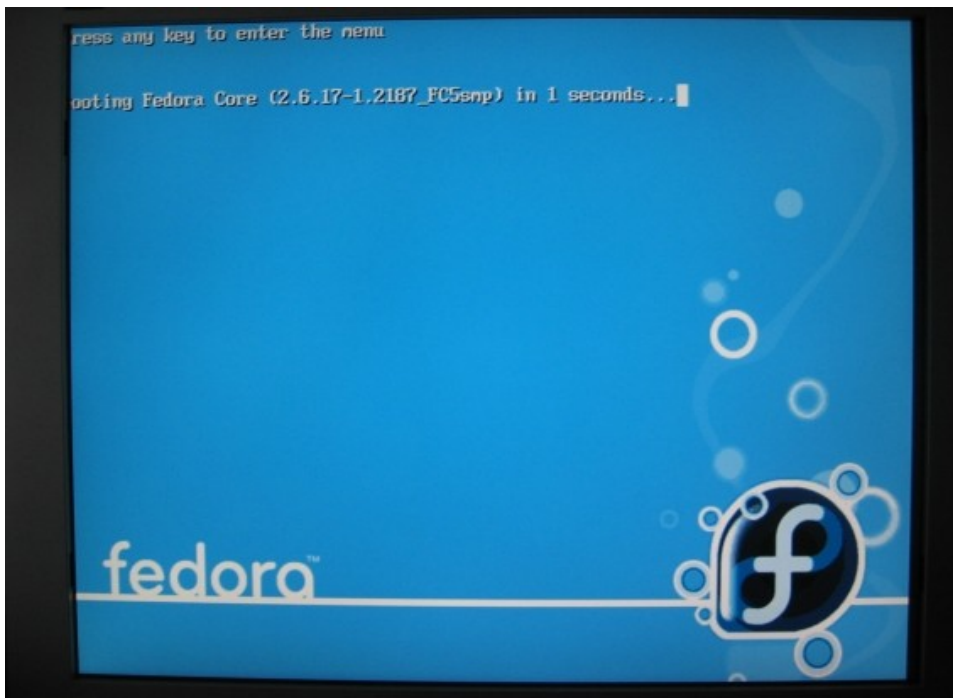


Figure 1. Fedora Core 5 boot screen.

You have (by default) five seconds to hit a key in order to display the menu of kernel choices (from the menu.lst file). Hitting a key will display the kernel menu, as shown in **Figure 2**.

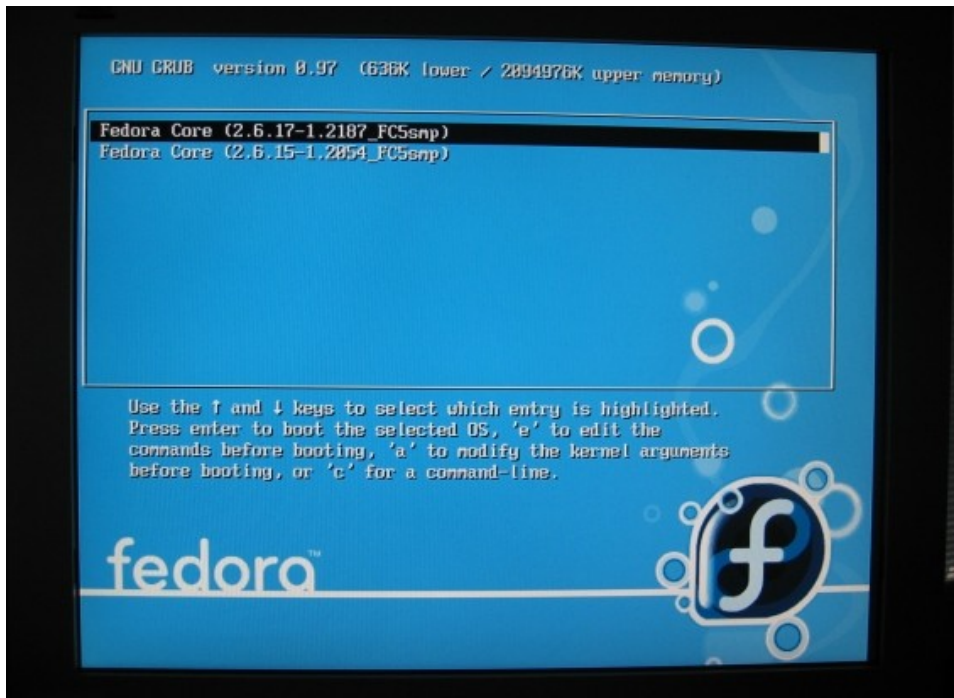


Figure 2. Kernel choices from the menu.lst file

You can load a specified kernel by highlighting the row and pressing Enter. For our situation, however, we're going to want to modify the GRUB boot commands – those three lines after the 'title' line in menu.lst described in Section 3 earlier.

In order to edit the grub boot options for a specific kernel, highlight that row in the menu and press 'e'. The boot commands for the selected kernel will be displayed, as shown in **Figure 3**.



Figure 3. Displaying the boot commands in the menu.lst for a specified kernel.

Once the boot commands are displayed, you can navigate from one command to another with the up and down arrows. Highlight the line that begins with "kernel", as shown in **Figure 4**.

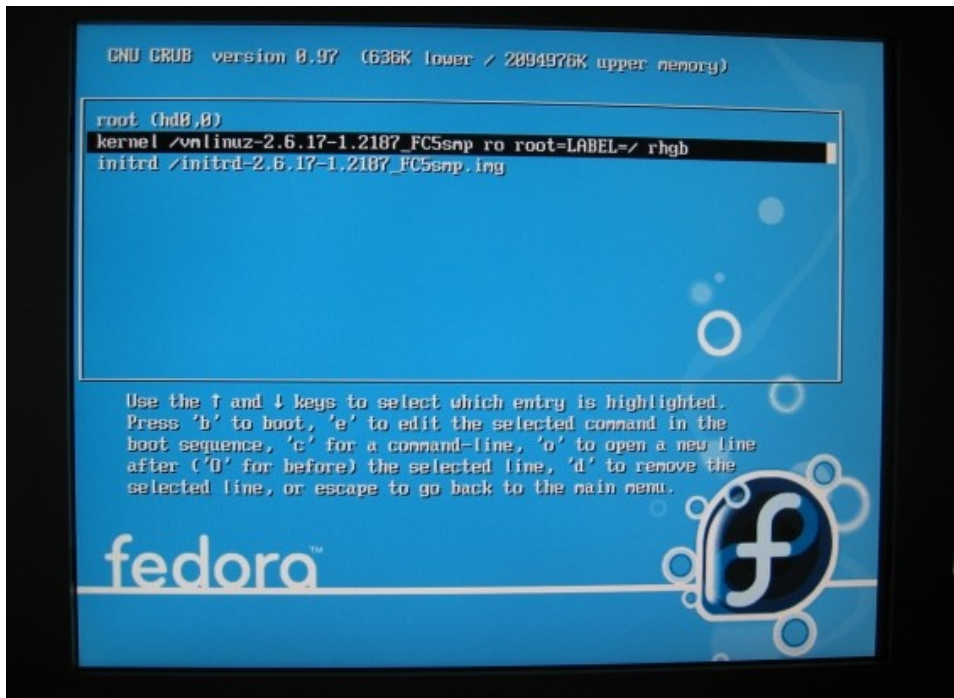


Figure 4. Selecting the kernel options line in the menu.lst file.

Once you've selected the line to edit, press 'e' in order to modify the line. The line you've chosen to edit will be displayed in single line editing mode, as shown in **Figure 5**.

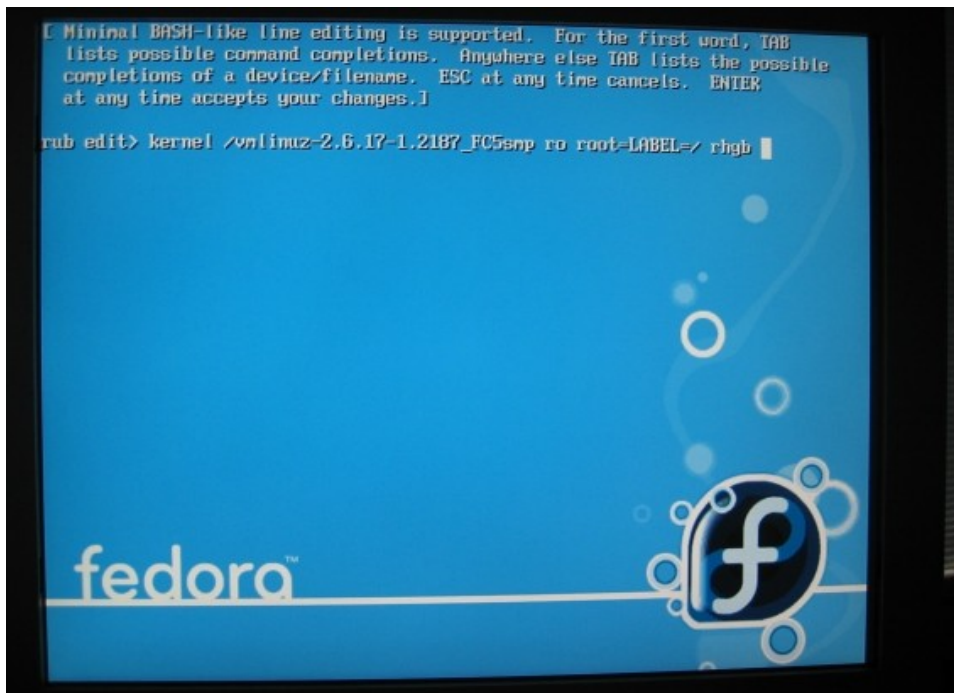


Figure 5. Editing a single boot command in the GRUB menu.lst file.

You can move back and forth through the line with the left and right arrow keys, use Delete and Backspace to get rid of characters, and type characters in the middle or at the end of the line. Once you're done, press Enter to accept your changes and return to the menu.lst file display of Figure 4.

In order to tell the boot process to load to runlevel 1, type a space and then a '1' at the end of the line. Note that the modifications you make aren't saved to the menu.lst file; they're simply applied to the command during this booting process. Then press Enter, as shown in **Figure 6**.



Figure 6. A modified boot command, with a new '1' parameter.

This parameter tells the boot loader to run the scripts for runlevel 1 (rc1.d). Now press 'b' to continue the boot process. Messages indicating the progress of the boot process will scroll by as usual, until INIT is finished with its work to start the system, and leaves you at a shell prompt, as shown in **Figure 7**.

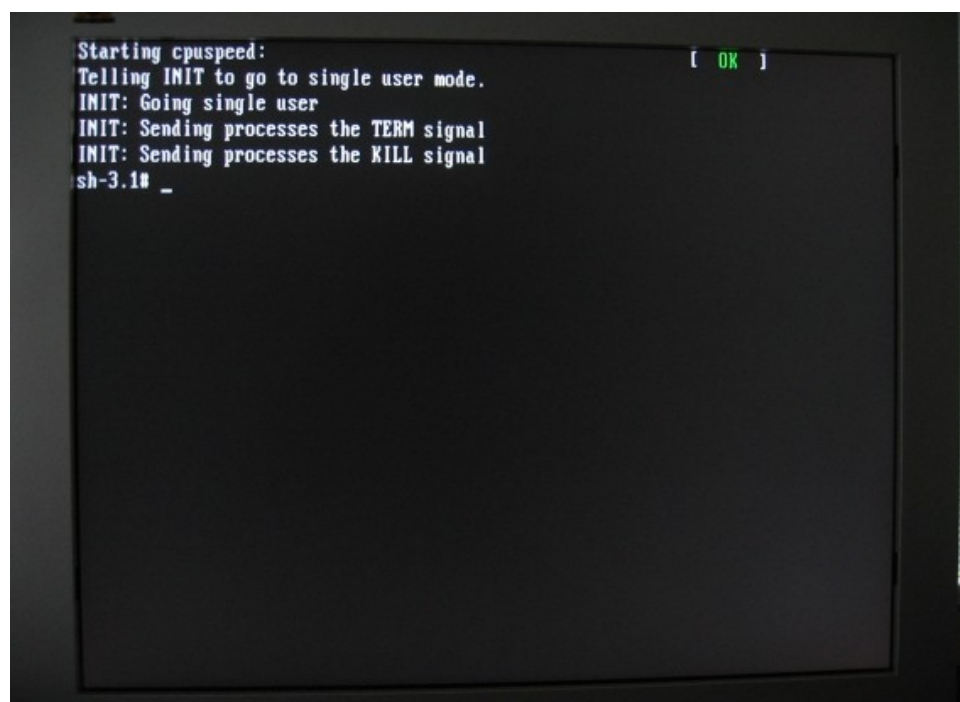
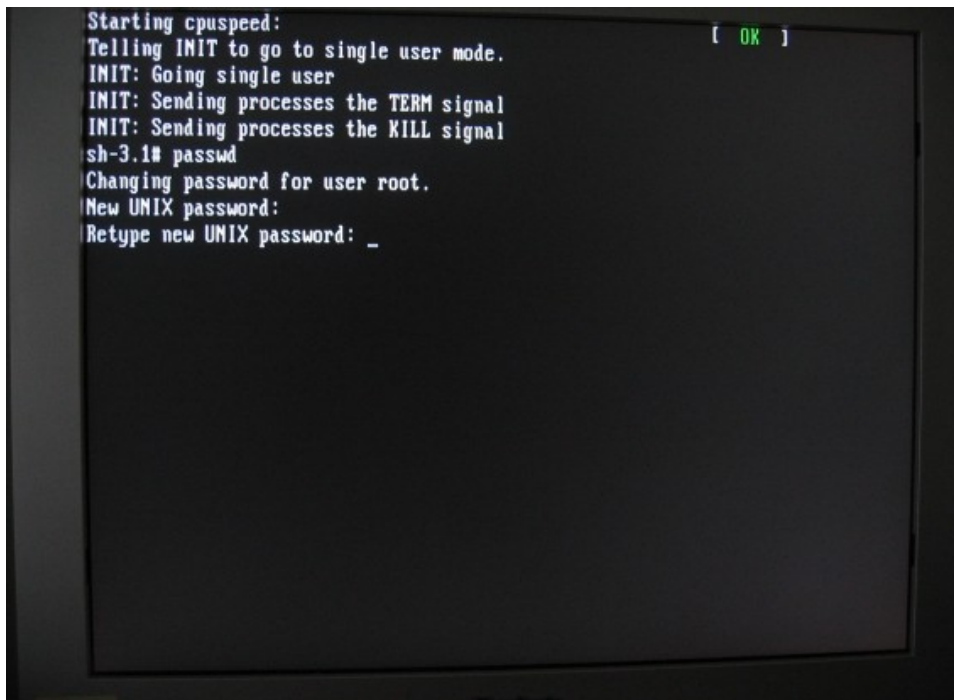


Figure 7. The boot process complete, runlevel 1 loaded to a shell prompt.

At this point, you're now logged onto the machine in single user mode, no networking, no graphical user interface, and you're the root user. You can now use the 'passwd' command to change the root account, as described in Section 4 and shown in **Figure 8**.



```
Starting cpuspeed:
Telling INIT to go to single user mode.
INIT: Going single user
INIT: Sending processes the TERM signal
INIT: Sending processes the KILL signal
sh-3.1# passwd
Changing password for user root.
New UNIX password:
Retype new UNIX password: _
```

Figure 8. Using 'passwd' to change the root password.

Once the password has been accepted, reboot your machine and you've got a brand spanking new root password.

8. Boot the machine to a shell prompt (SuSE)

Now we're going to use SuSE to demonstrate how to boot the machine to a shell prompt in those scenarios when even runlevel 1 requires a root password. You may be wondering how this is going to be different than what was described in Section 7 – it sure seemed like we ended up at a shell prompt there too!

The difference is that in Section 7, we let the init scripts for runlevel 1 execute. Runlevel 1 is defined as loading a shell prompt, so obviously you're going to end up at one. In this method, however, we're going to shortcut the init scripts, not letting any of them load, and instead just run a shell prompt in place of init. You'll see that by doing so, we'll have to manually mount the partition with the /etc/passwd and /etc/shadow files on them, because the init script that would mount that partition won't have run.

Both SuSE 9.0 and 10.0 use the same general process; the boot screens differ only cosmetically. I'll use 9.0 screen shots solely because they came out better.

Booting with SuSE

The SuSE boot process displays the various kernel choices available, as shown in **Figure 9**. (This is equivalent to the kernel choices displayed in Figure 2 for Fedora Core.)

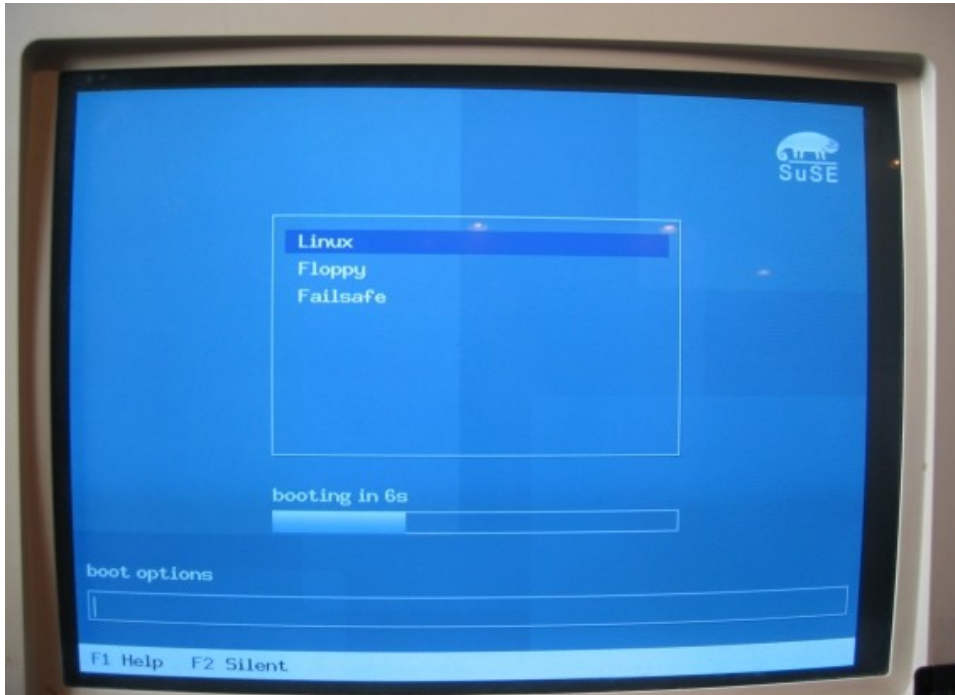


Figure 9. The SuSE 9.0 boot screen, displaying available kernel choices.

The highlighted choice will automatically boot in ten seconds. You can interrupt the process by moving the highlight to another option, or by typing characters into the text box labeled "boot options" at the bottom of the screen. This is where you'll type options to start up SuSE in runlevel 1.

The SuSE 10.x interface isn't much different, as seen in Figure 10. (Sorry for the bad screen shot – now you see why I'll use SuSE 9 for the rest of this discussion.) The "boot options" text box has been replaced by a "Boot Options" line.

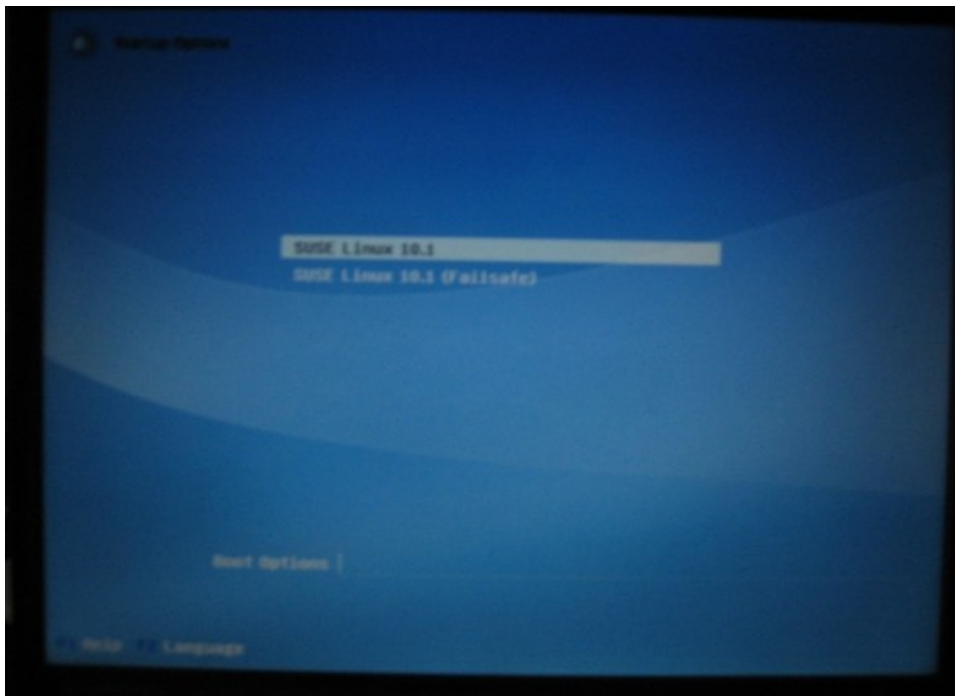


Figure 10. The SuSE 10.1 boot screen, also displaying available kernel choices.

In both versions, you can modify the boot process by typing boot options into the control at the bottom of the screen, as shown in **Figure 11**.

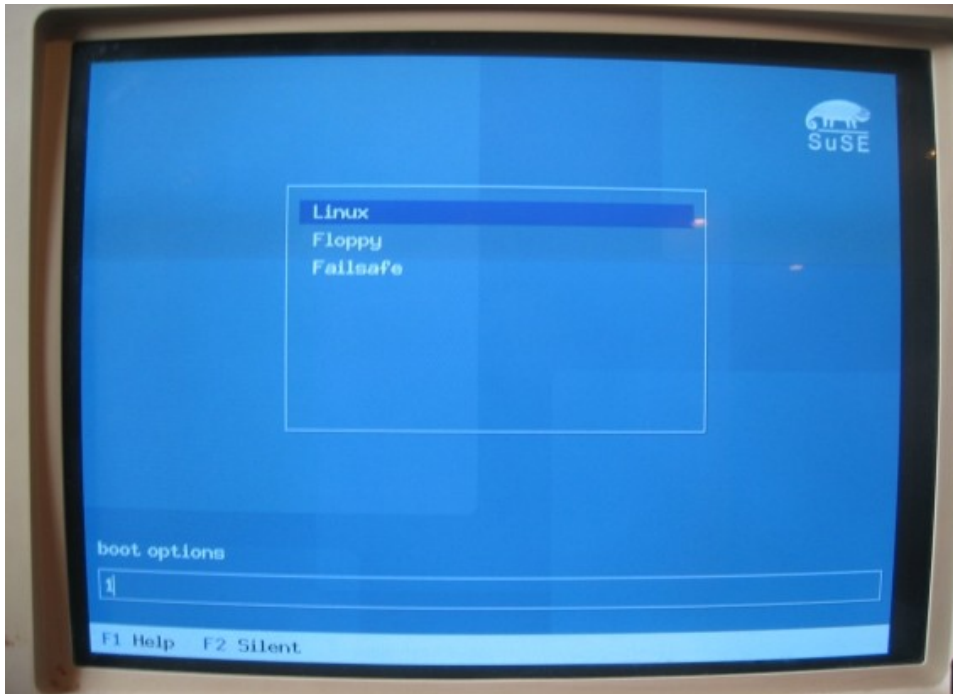


Figure 11. Entering the '1' boot option for the standard Linux kernel choice.

If you select the Failsafe kernel, you'll see that a number of boot options already entered into the Boot Options control, as shown in **Figure 12**.



Figure 12. The Failsafe kernel choice automatically enters a number of boot options.

Why booting to runlevel 1 doesn't work

Playing devil's advocate for a moment, suppose we went ahead and just tried to load SuSE into runlevel 1, like we did with Fedora. (You'd do so by typing a '1' into the Boot Options control, as shown earlier in Figure 11.) The progress messages will display as you'd expect, then `init` would finish up, leaving you at... a shell prompt? Nope, at a request to enter the root password – a much different result than what we saw for Fedora. See **Figure 13**.



Figure 13. Booting to runlevel 1 in SuSE results in a request for the root password.

Your heart sinks. What now? The answer is to enter a command that will shortstop the `init` boot process. Type '

```
init=/bin/bash
```

in the Boot Options control, as shown in **Figure 14**.

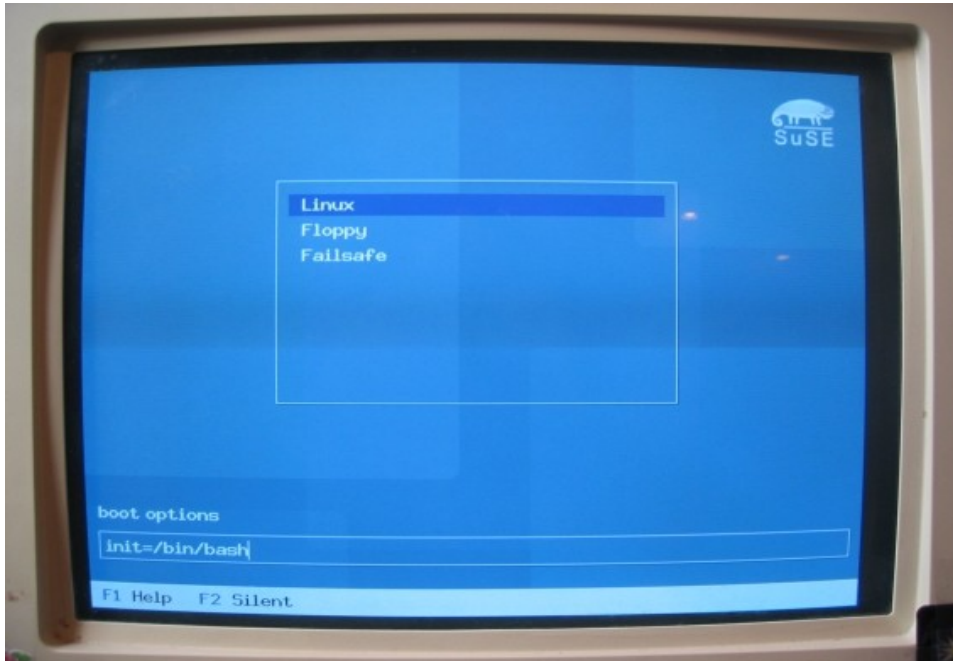


Figure 14. Entering a new command to `init`.

Press Enter, and SuSE will boot as expected. The progress messages will display, and you'll eventually end up at a shell prompt, as shown in **Figure 15**.



Figure 15. Booting to a shell prompt in SuSE.

In this situation, the "real" `init` never starts. Where the kernel would normally start the `init` program, it instead starts a shell since it's been told that that's where the `init` program is.

As mentioned, the difference between this and `runlevel 1` is that no hardware has been touched – you don't have access to the hard disk yet. Time to remedy that, by remounting the "/" ("root") partition to be read-write, manually.

Issue the command


```
# mount -o remount,rw /
```

Now you can issue the "passwd" command as discussed in Section 4, reboot, and you're all set.

Editing menu.lst with SuSE

What if you wanted to edit the menu.lst file during boot, like we did with Fedora Core? In order to get to the GRUB commands like we did in Figure 3, press Escape when presented with the list of SuSE kernels (Figures 9 and 10.) You'll be warned that you're leaving the GUI and asked to confirm your choice, as shown in **Figure 16**.

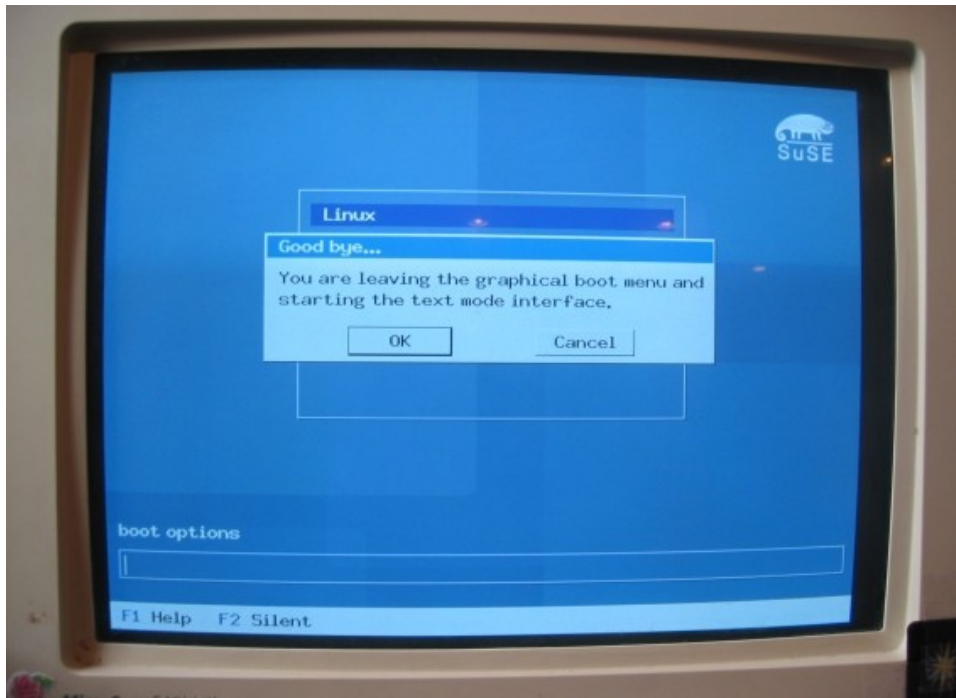


Figure 16. Switching to the text mode interface of SuSE's boot process.

Select 'OK' and the list of kernel choices will be displayed, albeit in a text mode display, as shown in **Figure 17**.



Figure 17. Available kernel choices in SuSE's text mode.

You can edit the menu.lst boot commands by highlighting the kernel of interest (say, "Linux"), and then pressing 'e', just like with Fedora Core. See **Figure 18**.



Figure 18. Displaying a kernel's menu.lst boot commands.

The rest of the steps are the same as with Fedora Core – make your edits in single line editing mode, press Enter to accept your changes and be returned to the screen shown in Figure 18, and then type 'b' to continue booting.

By the way, the difference between adding boot options as shown in Figures 11 and 12 to editing the menu.lst file is that the boot options are just passing parameters to the kernel line while this allows you to make edits to *any* of the lines in the GRUB boot loader.

9. Boot the machine with a Live CD (Knoppix + SuSE)

There's an alternate method to handling a recalcitrant operating system such as SuSE when it comes to root passwords – using a Live CD to load a version of Linux into memory, and using it to write to the password file on the disk. We'll use Knoppix as an example, although most any Live CD distribution would work.

The first step is to boot the machine with the Knoppix CD. The Knoppix boot screen will load, as shown in **Figure 19**.

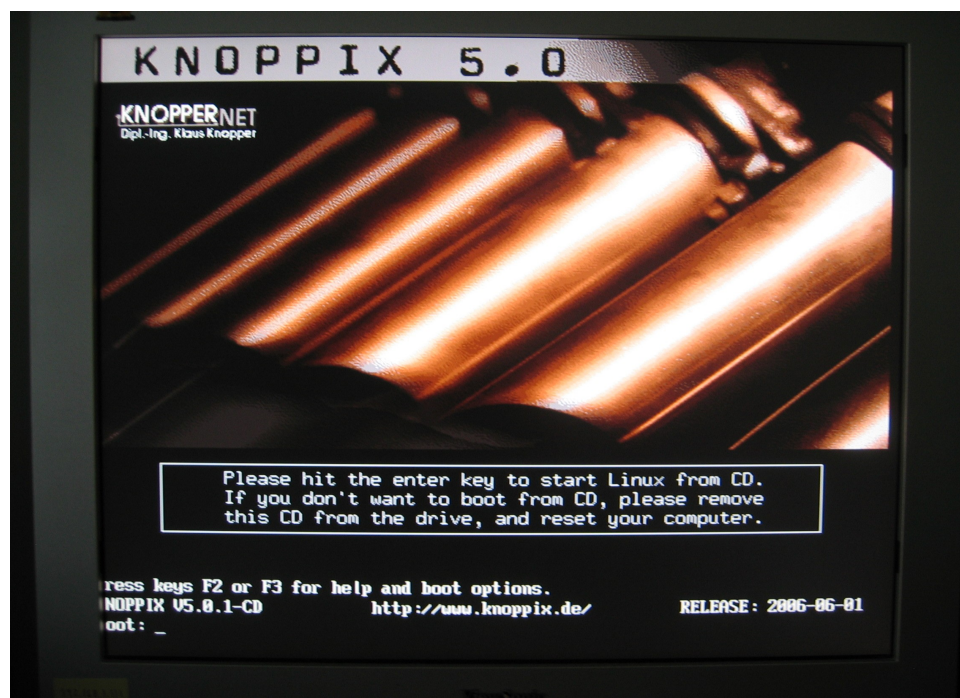


Figure 19. The boot screen for Knoppix 5.0.

Just press Enter at the 'boot:' prompt. You'll eventually be greeted by the Knoppix desktop, as shown in **Figure 20**.



Figure 20. The Knoppix 5.0 desktop.

The first thing you'll want to notice is that the hard disk partitions on the host machine – the machine you're trying to reset the root password for – are displayed on the desktop. If there is a small green arrow in the lower right corner of a partition's icon on the desktop, the drive is mounted. In Figure 20, the gold Knoppix CD icon is mounted, although it's sorta hard to see. See Figure 21 for an easier to view example.



Figure 21. The green arrow shows that hda6 is mounted.

If a partition isn't already mounted, you can right-click on the partition's icon and select the "Mount" command, as shown in Figure 22.

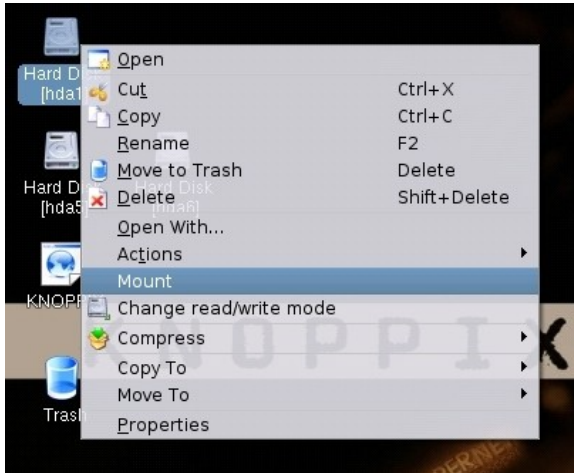


Figure 22. Mounting a partition in Knoppix.

Note that this mounts the partition as read-only. If you want a partition to be read-write, you'll have to select the "Change read/write mode" menu option *after* you mount the partition. (You have to do both!)

Alternatively, you can do this in a terminal window. Click on the Terminal icon in the task bar, change to root (by default, you're running as the "Knoppix" user when you boot up), and then mount the partition of interest. Supposing that "/" (and, thus, /etc) was located on hda3.

```
user> su -
root> mount -o remount,rw /media/hda3
```

Once mounted, it's time to change the root directory, like so:

```
root> chroot /media/hda3
```

You're now able to write to the hard disk's /etc partition, which means that you can use "passwd" to reset the root password:

```
#> passwd
Changing password for user root.
```

...and so on.

10. Using a rescue CD

This being Linux, there are always a half dozen different ways to accomplish most any task. Another way to handle the lost root password problem, if you've got the original installation CDs, is to use the first one to boot into Linux to where it asks you what kind of install or boot you want to do. Select the "Rescue" option.

Now you can edit the /etc/passwd and /etc/shadow files. In /etc/shadow, remove the encrypted password from the root account, so that the line looks like this:

```
root::13388:0:99999:7:::
```

Next, edit the passwd file, removing the 'x' from the second position, so it looks like this:

```
root::0:0:root:/root:/bin/bash
```

Now root can log in without using a password, and use 'passwd' to generate a new password.

11. Reinstalling

The truly desperate, with the original installation disks in hand, might consider (or have to) reinstalling the operating system. If /home was set up on a separate partition, it would be possible that you could format the "/" and "/boot" partitions, while leaving "/home" alone and saving your data.

12. GRUB vs LILO

This discussion has assumed GRUB as the boot loader. If you're using LILO, you can pass the parameter "single" in order to boot into single user mode:

```
LILO: linux single
```

The rest of the boot process is pretty much the same.

13. 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**