Sentry Firewall CD(tm) HOWTO

Stephen A. Zarkos, Obsid@Sentry.net

v1.4.0, 2004-09-10

This document is designed as an introduction on how the Sentry Firewall CDROM works and how to get started using the system.

---

1. Introduction

2. How the CD Works (Overview)

3. Obtaining the CDROM

4. Using the Sentry Firewall CDROM

---

1. Introduction

This is the long-overdue Sentry Firewall CDROM HOWTO. This document is designed to help you get started using the Sentry Firewall CD and answer some of the questions you may have about the system and how it works. The most current version of this howto can be obtained at the following URL: http://www.SentryFirewall.com/files/howto/.

This document is designed to be somewhat brief, but complete. This is not, however, a detailed reference for the Sentry Firewall CD. Please see the "Further Reading" section below for links to more resources and references.

If you would like to add anything to this document, or if you have any questions or comments please feel free to email me, Obsid@Sentry.net.

1.1 Project Goals

The general goal of this project is mentioned several times within the documentation - that is simply, to build a bootable CDROM-based system that can be easily and dynamically configured. In the end, I wanted the configuration to rival that of any commercial router that utilizes configuration files. I also wanted the system to be simple, secure, and highly functional in a large number of operating environments - not just as a firewall. This, of course, has proven to be a difficult balance to maintain.

At the present time, the basic goals have been fulfilled. However, I believe there is still a great deal of development that can and needs to be done in order for the Sentry Firewall to be a truly diverse Linux distribution.

1.2 Further Reading

- Sentry Firewall CD HOWTO
- Sentry Firewall CD Reference Guide
- Sentry Firewall CD FAQ
- Mailing Lists

1.3 Copyrights and Disclaimer

The current copyright and disclaimer can be found on the website; http://www.SentryFirewall.com/files/COPYRIGHT. It applies to the Sentry Firewall CD, and all the scripts and documentation associated with it.

1.4 Minimum Requirements

The following are the minimum requirements for successfully running the Sentry Firewall CD:

- x86 computer with CD-ROM(486 or better).
- BIOS that supports the eltorito standard(booting from the cdrom).
- 32MB RAM minimum. 64MB or more recommended for firewall/router/DNS server. 96MB+ recommended if running other services(ie. snort, squid).
- Easy access to coffee/tea/soda or equivalent stimulant.
- Floppy disk drive(optional).

2. How the CD Works (Overview)

This section is just an overview to explain how the Sentry Firewall CD works, that is, from the process of loading the kernel to running the Sentry Firewall CD configuration scripts located on the RAMDisk.

2.1 The Boot Process

Booting from the CDROM is a fairly familiar process. The BIOS execs the bootloader(Syslinux) which then displays a boot prompt and loads the kernel and ramdisk into memory. Once the kernel is running the ramdisk is then mounted as root(/) and then "init" is run which then starts running the startup scripts.

As of version 1.5.0-rc11 the Sentry Firewall CD utilizes a tmpfs file system for its root partition. The tmpfs file system, also known as "virtual memory file system" or "shm fs", provides two major advantages:

- Tmpfs file systems reside in virtual memory, which means they can be swapped out.
- Tmpfs file systems can be resized.

As a result of this development, the boot sequence of the Sentry Firewall CD has changed to the following(roughly):

- Isolinux boots kernel
- Kernel decompresses, boots, mounts initrd as root(/) on /dev/ram0.
- /linurc is run which -
  - Mounts a tmpfs file system on /root
  - Places the files for the new root in /root.
  - Moves old root to /root/initrd and makes /root the new root.
  - init is run, rc.S unmounts/deletes /initrd and flushes /dev/ram0.

2.2 ISOLINUX

Early versions of the Sentry Firewall CD utilized the 2.88MB floppy emulation method, along with either lilo or syslinux to boot the kernel and load the ramdisk. This method proved very limiting for two reasons; A) the total size of the compressed ramdisk AND kernel was limited to 2.88MB, and B) it was quite slow compared to the current method.

The Sentry Firewall CD is currently utilizing the isolinux.bin boot record with no emulation in order to properly boot the CDs. This allows us to use a much larger ramdisk and offer a choice of several kernels to boot at boot-time.

More information about syslinux can be found at syslinux.zytor.com.

2.3 The CD Configuration Scripts

An obvious necessity for deploying CDROM based systems is the ability to dynamically configure the system for various environments with different configurations - which is what a good majority of this project is dedicated to building. A simple way to do this is to give the user the ability to customize the startup scripts and files located in /etc before they are actually used.

At boot time, the "/etc" directory and subdirectories are nearly empty. On many Linux systems, the first startup script to run is /etc/rc.d/rc.S, /etc/init.d/rcS, /etc/rc.d/rc.sysinit or something similar - the /etc/inittab file is actually responsible for defining this. It is from this startup script where we run the configuration scripts that look for a configuration file, called "sentry.conf", and place the proper configuration and system files in /etc and various subdirectories under /etc.

The sentry.conf file tells the configuration scripts where to go to obtain current copies of files that should be replaced. These files will often be configuration or startup files and will mainly reside in /etc, although they could be placed anywhere on the system the user prefers. If a configuration directive for a specific file is not found in the sentry.conf file, or if a configuration file cannot be found at all, then the default system files are used - which are located in /etc/default/* on the ramdisk.

As previously mentioned, our configuration scripts are generally run from an init script located in /etc/{rc.d,init.d}/. The first of our configuration scripts to run is called ’cd-config.pl’, which is essentially the mainline for the entire program. The other scripts that are used are called ’get_config.pl’, ’process_conf.pl’, ’do_config.pl’ and ’networking.pl’. These scripts were written specifically for this project, and are essentially the mainstay of the entire configuration process.

In depth review of these scripts is a little beyond the scope of this document, but is covered a bit in the Sentry Firewall CD Reference Guide available on the website (www.SentryFirewall.com). The files are written in perl and are actually responsible for much of the overall configuration of the system. In short, however, they perform the following tasks:

- Read in and parse the "sentry.conf" configuration file.
- Locate and retrieve the files detailed in the sentry.conf file.
- Replace the system default files with those the user has defined in the configuration file.
- For those files not declared in the sentry.conf file, replace them with the system default files.

3. Obtaining the CDROM

3.1 Which Branch?

The Sentry Firewall CD project has several "branches". To understand what the differences are between these branches, let me first explain basically how the CD is designed.First, the actual Linux system on the rootdisk and ISO is centered around a "host" Linuxsystem, which is based on a particular Linux distribution. Then there are the configurationscripts, written in perl, that are kept on the ramdisk and run shortly after the ramdiskis mounted as root(/). These configuration scripts are what parse the sentry.conf fileand allow us to do all this neat configuration stuff via a floppy disk or a remote server(explained later).

Currently, there are three main "stable" branches and three "devel" branches, one for each of the stable ones. Each branch is based on a different Linux distribution, and as such often have different configuration and init files. For example, the first versions of the Sentry Firewall CD were based on Slackware, now called the "SENTRYCD" branch. Slackware usesrc.whatever type files located in "/etc/rc.d/", whereas other branches use SystemV-ish initfiles and other configuration files. As a result the configuration directives available foreach branch will vary. I will explain more about the sentry.conf file and configurationdirectives later.

Which branch to use is, of course, your choice. However, at the time of this writing, the SENTRYCD branch(Slackware-based) is by far the most stable. But this will likely change as theSENTRYCD-DEB branch becomes stable. The following is a current list of available Sentry FirewallCD branches:

- **SENTRYCD** - Slackware-like Sentry Firewall CD.
- **SENTRYCD-DEB** - Debian-like Sentry Firewall CD. (In Development)
- **SENTRYCD-RH** - R*dH*t-like Sentry Firewall CD. (Deprecated)

3.2 Downloading

The CDROM is distributed as a gzip or bzip2 compressed iso image, and is generally between 95-105MB in size. Uncompressed ISO images are generally between 250-350MB in size. Available download mirrors are listed on the websites; http://www.SentryFirewall.com/ or http://Sentry.Sourceforge.net/.

3.3 Purchasing

Although the iso image is free to use and distribute, copies of the Sentry Firewall CD can be mailed to you at a minimal cost. Custom versions of the CD and support can also be made available and tailored to a specific network configuration.

For more information about these services, please email me.

3.4 Obtaining Updates

Updated versions of the Sentry Firewall CD are made available on occasion when new features are added or packages updated. There are two ways to update the Sentry Firewall iso; you may either download the new iso in its entirety(approx. 110MB) or download a binary "patch" file to update your current iso image.

3.5 Patching the ISO Image

Recently, updated versions of the Sentry Firewall CD ISO have been distributed as binary "patch" files that are generally considerably smaller than the complete ISO image. In order to utilize these patch files you will need the "xdelta" utility available at http://sourceforge.net/projects/xdelta/.

- Retrieve a binary patch from one of the download sites that matches the iso version you currently have. For example, if you have version 1.5.0-rc5, you will want to download the file 1.5.0rc5-1.5.0rc6.xdelta.patch to update your iso image to -rc6.
- Locate and uncompress your current Sentry Firewall CD iso image:
  Example: **bzip2 -d sentrycd-1.5.0-rc5.iso.bz2**
- Now, use the xdelta utility to apply the patch:
  Example: **xdelta patch 1.5.0rc5-1.5.0rc6.xdelta.patch sentrycd-1.5.0-rc5.iso**

Applying the patch can take several minutes. You will also want to make sure you have ampledisk space available since xdelta does not modify the old iso image, but instead uses the patchfile and the iso image to create a new, updated iso image.

3.6 Burning the CDROM

This section will attempt a general overview on how to burn the CD ISO image once you have obtained it from one of the mirrors. All the commands presume you're working in Linux. Burning ISO images in Windows is not covered in this howto. If you are using windows then check out the CD Burning Howto First, let's decompress the ISO image:
NOTE: Make sure you have enough disk space, the uncompressed iso image can be somewhere between 250MB and 350MB.

```
blah@wherever:~$ gzip -d sentrycd.iso.gz

or

blah@wherever:~$ bzip2 -d sentrycd.iso.bz2
```

Verify the integrity of the iso image,

```
blah@wherever:~$ md5sum -b sentrycd.iso
```

Now, let’s try to burn the CD. You’ll need the ’cdrecord’ utility available, it can be obtained here.You will want to run ’cdrecord -scanbus’ in order to find the ’dev’ valuerequired for the following command. You will also need to know the write speedof your CD-RW. Details on how to set this all up are beyond the scope of thisdocument, please refer to the CD Writing HOWTOfor more details.

```
blah@wherever:~$ DEV="DEV_LINE_HERE" SPEED="SPEED"
blah@wherever:~$ cdrecord -v -data speed=$SPEED dev=$DEV sentrycd.iso
```

That’s it, you now have a Sentry Firewall CDROM. By the way, you may have to be ’root’ to do all this.

If you simply want to look at the ISO image without actually burning the CD, you can mount the image on a loopback device;

```
blah@wherever:~$ mount -o loop ./sentrycd.iso /MOUNT_POINT
```

Where "MOUNT_POINT" is where you would like the CD mounted. You may then ’cd’ to the MOUNT_POINT directory and poke around - don’t forget to ’umount’ the image once you’re finished. This assumes you have support in your kernel for the loopback device. You probably do, but once again, recompiling kernels is beyond the scope of this document.

4. Using the Sentry Firewall CDROM

4.1 Introduction

The Sentry Firewall CD configuration scripts are run shortly after the rootdisk is mounted as root(/). The first objective of these scripts is to look for and parse a configuration file called ’sentry.conf’. The scripts will first try to find the file on a floppy disk - which, if found, will be mounted on "/floppy". If a sentry.conf file is not found in "/floppy", then a default configuration will be used.

In order to configure the Linux system for use in any particular environment the user must have the ability to replace the system default files with his/her own copies. The ’sentry.conf’ file basically tells the configuration scripts which files it should replace and where those files are.

4.2 The sentry.conf file

The main configuration file for the system is called ’sentry.conf’. The file accepts several configuration directives, many of which will be discussed below. It may also be a good idea to take a look at the default sentry.conf file which is available on the ISO in the "<CDROM>/SENTRY/scripts/cd-config/" directory, on the disk images(discussed later), oron the project website. This file containsall the currently supported directives for the particular branch it was designed for, as wellas a bunch of useful comments inline.

The configuration scripts will attempt to mount several devices in its attempt to locate the sentry.conf file.

- Floppy disk, **/dev/fd0** mounted on **/floppy**.
- USB thumbdrive, **/dev/sda1** mounted on **/floppy** - version 1.5.0-rc9 or above.
- Hard disk, **/dev/hda1** mounted on **/mnt**.

Example

A basic configuration file looks like the following (everything after a '#' sign is interpreted as a comment):

```
----snip----
## Basic Sentry Firewall CD config file(sentry.conf)

rc.local = /floppy/config1/rc.local
fstab = /floppy/config1/fstab

passwd = /floppy/config1/passwd
shadow = /floppy/config1/shadow

# EOF #
----snip----
```

The syntax is pretty simple, the default 'rc.local' file will be replaced with the user defined 'rc.local' file located in the '/floppy/config1/' directory. Same goes for 'fstab', 'passwd', and the 'shadow' file. But it is important to remember, the first place the sentry.conf file will be looked for is on /dev/fd0 or /dev/sda1(USB device),which if found, will be mounted on /floppy. This is why all these files appear to be locatedin the /floppy directory, it is simply the mount point for the floppy disk or the USB device. **NOTE:** As of version 1.3.0, a user may now omit the '/floppy' prefix. So, for example a line in sentry.conf that says the following:

```
shadow = config1/shadow
```

Will be assumed to mean(in most cases) the following:

```
shadow = /floppy/config1/shadow
```

As long as the directory config1/shadow exists on the configuration floppy.

Unfortunately, you cannot arbitrarily replace files, for example the following will likely not be parsed correctly:

```
foo.conf = /floppy/config1/foo.conf
```

The configuration scripts only recognize a certain number of configuration files, so it probably won't know what to do with "foo.conf". There are other very easy ways to copy unknown configuration files into their proper location, however. These methods will be discussed below.

4.3 Network Configuration

As of version 1.0.5, a new syntax for the configuration directives are recognized; those with an "http://" or "ftp://" prefix. This basically means that the following syntax is now supported:

```
inetd.conf = ftp://[user:pass@]123.123.123.123/config1/inetd.conf
hosts = http://[user:pass@]123.123.123.123/config1/hosts
```

As of version 1.3.0, "https://", "scp://", and "sftp://" URLs are also supported. For example:

```
shadow = scp://<user>:<pass>@123.123.123.123/dir/shadow
passwd = sftp://<user>:<pass>@123.123.123.123/dir/passwd
fstab = https://[user:pass@]123.123.123.123/dir/fstab
```

NOTE: The username and password fields are required when retrieving files via scp or sftp. Empty passwords are not permitted.

In order to accomplish this, the configuration scripts need to have the ability to set up an ethernet interface, as well as obtain nameserver information from the sentry.conf file. We use the 'device' directive to set up an interface for network configuration support.

```
Usage:
    device{1..10} = <device>:<driver>:<IP address>[|Gateway_IP]
    OR
    device{1..10} = <device>:<driver>:dhcp[|Hostname]
```

And to set up a nameserver:

```
Usage:
    nameserver = <IP_ADDRESS>
```

Additionally, when retrieving files using "http", "https", or "ftp", you may also set up a proxy server. The following directives will allow you to do so (they may not all be required for your setup):

```
http_proxy = http://<hostname>/
ftp_proxy = http://<hostname>/
proxy-user = <PROXY_USER>
proxy-passwd = <PROXY_PASSWORD>
```

Passive FTP may also be required. If so, use the 'passive-ftp' option, ie:

```
passive-ftp = <on|off>  ## Default == off
```

For example to set up an interface called "eth0", which uses the "tulip" driver and can obtain its ip address from a DHCP server, we can use the following line:

```
device1 = eth0:tulip:dhcp
```

As you can see, a total of 10 devices are allowed. Let's say we now want to set up an interface "eth1" that uses uses the "8139too" driver, and has a static IP(192.168.1.2) and a default gateway(192.168.1.1):

```
device2 = eth1:8139too:192.168.1.2|192.168.1.1
```

IMPORTANT NOTES:

- **1)** <hostname> and <gateway> are optional, but sometimes required.
- **2)** Only one <gateway> can be declared, that is, you cannot set up more than one default gateway.
- **3)** Devices set up with the 'device{1..10}' directive are TEMPORARY and are taken down after the configuration process is complete. See rc.inet1{.conf} for more permanent network setup.
- **4)** Please see file: /SENTRY/scripts/cd-config/networking.pl for list of supported devices. Most

10/100BaseT ethernet devices should be supported.

Example

```
----snip----
## Basic Sentry Firewall CD config file to retrieve files via
HTTP(S)/FTP/SCP/SFTP.
device1 = eth0:tulip:192.168.1.2|192.168.1.1
nameserver = 123.123.123.123  ## This should be the IP of your DNS server.

rc.M = ftp://user:pass@config.sentry.net/node1/rc.M
rc.inet1 = http://user:pass@config.sentry.net/all_nodes/rc.inet1

passwd = scp://user:pass@config.sentry.net/all_nodes/passwd
shadow = sftp://user:pass@config.sentry.net/node1/shadow

# EOF #
----snip----
```

4.4 Files You Should Replace

The particular files you need to replace depends entirely on your needs as well as the Sentry Firewall CD branch you are working with. Please take a look at the sample sentry.conf file for a list of the available configuration directives for each branch.

Here is a short list of files of particular interest:
Generic Directives(not branch-specific):

- **shadow** - Contains your own password hashes.
- **hostname** - Hostname for the machine.
- **resolv.conf** - List of nameservers.
- All the ssh_host_{rsa,dsa}_key directives.

SENTRYCD Branch

- **rc.inet1** or **rc.inet1.conf** - Ethernet interface setup.
- **rc.inet2** - Start daemons and services.
- **rc.firewall** - Enable firewall and/or NAT rules.
- **rc.keymap** - Load a new keymap.
- **rc.local** - Local initialization file.

SENTRYCD-DEB Branch

- **Under Construction**

Please keep in mind that the Sentry Firewall CD is capable of performing a great many tasks in a number of operating environments. Ramdisk space permitting, you may replace and customize as many or as few files as you wish to suit your needs.

It is worthwhile to note, however, that "/usr" on the ramdisk is actually a symlink to "/cdrom/usr", and thus is read-only. Therefore, files in /usr cannot be added or replaced using the sentry.conf file. The only way to manipulate the contents of the /usr directory is to rebuild the ISO image, which will be discussed in later sections.

4.5 Other Useful Configuration Directives

Copy a File.

Copy file '/floppy/someconfig.conf' to '/etc/someconfig.conf'

```
Usage:
    /floppy/someconfig.conf |= /etc/someconfig.conf

    OR, this does the same thing -
       /etc/someconfig.conf = /floppy/someconfig.conf

    and this is also possible -
       ftp://<server>/someconfig.conf |= /etc/someconfig.conf
       /etc/someconfig.conf = ftp://<server>/someconfig.conf
```

Create a Symlink.

Make a symlink called '/etc/someconfig.conf' that points to '/etc/otherconfig.conf'.

```
Usage:
    /etc/someconfig.conf => /etc/otherconfig.conf
```

Make a Directory.

Note: The 'mkdir' directive is only available with the Sentry Firewall CD versions 1.5.0-rc14 or newer.

```
    Syntax:
        mkdir <PATH/DIRECTORY>[:MASK]
```

Make a directory in the specified location with the specified permissions(MASK). MASK is optional and defaults to 0755(rwxr-xr-x). This directive can be useful if you want to copy files at boot-time to a directory that does not exist on the ramdisk by default.

The 'include' Directive.

This directive grabs another sentry.conf file either from another location.

```
Usage:
    include = ftp://user:pass@config.sentry.net/node1/sentry.conf
```

NOTE: Any configuration directives parsed from the new sentry.conf file will clobber any identical directives that were previously declared.

The 'path<#>' Directive.

Note: The 'path<#>' directive is only available with the Sentry Firewall CD versions 1.5.0-rc13 or newer.

Path statements tell the configuration scripts where to look for files. These can specify a path on a local or remote system. The variables "path1" to "path10" are allowed.

```
Syntax:
     path<#> = <PATH>
     path<#> = <URI>
```

NOTE: <URI> should point to a directory on a remote system, NOT just a file.

```
Examples:
     path1 = /floppy/node1/config/
     path2 = scp://user:pass@someserver/node123/config/
     path3 = http://user:pass@someserver/node123-backup/config/
     etc etc...
```

You may then use the following syntax when declaring a file within your sentry.conf:

```
Examples:
     squid.conf = squid.conf
     or
     /etc/someconf.conf = someconf.conf
```

The configuration scripts will first look for "squid.conf" or "someconf.conf" in $m_point, which is usually /floppy. If it isn't found, then the system will try path1..path10 in order until "squid.conf" or "someconf.conf" is found. This not only makes for less typing when creating your sentry.conf, but it also allows you to add some redundancy to the configuration process.

The 'cdrom' Directive.

The 'cdrom' directive defines which device the CDROM is. Most of the time the CDROM is detected and mounted using the /etc/rc.d/rc.cdrom script. But this makes the process less error-prone.

```
Usage:
    cdrom = <DEVICE>

Example:
    cdrom = /dev/hdc
```

The 'cron' Directive.

The 'cron' directive replaces a user's crontab file.

```
Usage:
    cron:<USERNAME> = </LOCATION/OF/CRONTAB_FILE>
```

Where <USERNAME> is a valid username on a running Sentry Firewall system.

The 'hostname' Directive.

The 'hostname' directive defines the hostname of the local machine. This directive can be used to either point to a file containing the hostname of the local machine, or to define the hostname itself.

```
Usage:
    hostname = </path/to/file>
    or
    hostname = MYHOSTNAME
```

The ’add_swap’ Directive.

Note: The ’add_swap’ directive is only available with the Sentry Firewall CD versions 1.5.0-rc11 or newer.

The ’add_swap’ directive tells the configuration scripts to add a swap partition at configuration time. If the ":format" option is appended to the variable, then the configuration scripts will also format the partition before activating it.

Warning: An improper setting of this variable could cause serious damage to data.

```
Usage:
    add_swap = /dev/hda1
    add_swap = /dev/hda1:format
```

The ’root_size’ Directive.

Note: The ’root_size’ directive is only available with the Sentry Firewall CD versions 1.5.0-rc11 or newer.

The ’root_size’ directive allows one to change size of root(/) at configuration time(before any other files are copied). By default the root filesystem is around 18MB in size. This option allows you to change the size of the root filesystem if you need more/less space. Also - since root is mounted on a tmpfs filesystem - this area can be swapped out as needed. The suffix g, m, or k is accepted for binary kilo, mega and giga. If no suffix is added, a size in megabytes is presumed.

```
Usage:
    root_size = "18M"
```

The size of the root file system can also be changed after configuration by simply remounting it, ie "mount -oremount,size=24M /"

4.6 Setting the Timezone.

Set the timezone from your sentry.conf file.

```
Example:

    /etc/localtime => /usr/share/zoneinfo/GMT
    /etc/hardwareclock = hardwareclock
```

"/etc/hardwareclock" either contains a line with "localtime" or "UTC". Usually for PCs the clock is set to local time, therefore put "localtime" into the file.

There are more than a thousand timezones under "/usr/share/zoneinfo/" to select from. Under slackware this can also be done interactively with the command "timeconfig".

4.7 Managing multiple nodes from a single location.

In order to manage multiple nodes at a single location, you can use a bare sentry.conf file located on a floppy disk, and then grab files from your ftp or http servers.

```
----snip----
## Basic Sentry Firewall CD config file.

device1 = eth0:tulip:dhcp
nameserver = <DNS_IP>
include = ftp://user:pass@config.sentry.net/node1/sentry.conf

----snip----
```

The included sentry.conf file will then be parsed, and files replaced via http or ftp if you like. You can now edit your sentry.conf and configuration files at a central server instead of on each individual floppy.

4.8 Example sentry.conf and disk images

An example configuration disk image is available on the CDROM. The disk is an ext2 formatted disk, and is located in the "/SENTRY/images/" directory on the CD. Use a command like the following to create the configuration disk:

```
blah@wherever:~$ dd if=/cdrom/SENTRY/images/ext2-144.img of=/dev/fd0
2880+0 records in
2880+0 records out
```

The disk images and a sample sentry.conf file can also be found on the website, http://www.SentryFirewall.com/

4.9 Saving your configuration

Once you have booted into your new Sentry Firewall system and have configured it to yourneeds, you will need to create a configuration floppy with a sentry.conf file and the files you will want to replace at boot time. One way to do this is to utilize the sample diskimage(see above) to create a disk image, copy the altered files, and edit the sample sentry.conffile by hand. This is not as cumbersome of a task as one might think, and can give you a greatamount of control over how the system is configured.

Another option is to utilize the */sbin/mkconfig* script on the Sentry Firewall system.This is a perl script that uses a dialog(1) based gui that can assist you in identifying thechanged files in /etc, create a sentry.conf file, and copy the changed files to a floppy diskor floppy disk image. Simply run "/sbin/mkconfig" and follow the prompts. This software isstill considered BETA, and its functionality is highly subject to change. Please send anypatches/bugs to Obsid@Sentry.net.

NOTE: The ’mkconfig’ script has only recently become somewhat stable, as of version 0.3-BETA that is included with the Sentry Firewall CD version 1.5.0-rc11. Older versions of the mkconfig script(prior to 0.3-BETA) were not nearly as functional and should probably not be used.

5. Setting Up a Firewall

5.1 Starting the Firewall

Ok, so the project is called the Sentry *Firewall* CD. So where’s the firewall? Well, it’s important to note that this system is capable of quite a bit more than your standard bootable floppy or CD firewall. In fact it is a pretty complete Linux system on a CD, and as with any Linux system the "firewall" is set

up using scripts and various userland utilities such as ipchains or iptables.

IPChains or IPTables firewall scripts generally take the form of shell scripts that are customized by the user and run at boot-time. If you already have a ruleset for your firewall simply edit the "rc.firewall" directive in your "sentry.conf" file to point to your firewall script on your floppy or on a remote HTTP(S)/FTP/SCP/SFTP server as explained above. The firewall will then be run at boot time.

It is also important to note that many of the firewall building tools available - includingsome that are mentioned in later sections - do not simply generate a firewall shell script, butrather set up a running firewall and use the ’iptables-save’ and ’iptables-restore’ utilities todump/load the firewall configuration to/from a file. A file created by ’iptables-save’ must be loaded using ’iptables-restore’, it cannot simply be executed like a shell script. Therefore ifyou want to load your firewall from a file that was created by iptables-save, then you may savethe file on your configuration floppy and declare its location using the "rc.firewall.save"directive in the sentry.conf file, instead of using the "rc.firewall" directive.

5.2 Using FWBuilder with the Sentry Firewall CD

FWBuilder(http://www.FWBuilder.org/) is a firewall configuration and management system. The advantage to this application is that it provides a graphical user interface to develop and modify firewall rulesets on various platforms using various utilities. The Firewall rulesets that are created with FWBuilder are completely compatible with the Sentry Firewall CD, and with just about any Linux firewall.

As with most Linux firewalls there are no X11 binaries or libraries on the Sentry Firewall CD, so you will need to develop the firewall ruleset on a separate workstation using fwbuilder and then upload the ruleset to the various firewalls/routers/nodes on the network. The following are the basic steps required to get your new fwbuilder ruleset running on the Sentry CD:

- Configure your new firewall to your liking with fwbuilder(duh).
- Save your firewall. Choose File->Save As, and choose an appropriate name. The file will normally be saved as "whatever.xml".
- Compile the firewall. Choose Rules->Compile. The ruleset will be compiled and turned into a shell script called "whatever.fw".
- You will then want to copy "whatever.fw" to your configuration floppy and use the "rc.firewall" configuration directive in your sentry.conf file to point to your new firewall script. The firewall script will be copied to "/etc/rc.d/rc.firewall" during the configuration process and run at boot-time.

Please note that it is not necessary to reboot the Sentry Firewall CD every time you update your firewall script. You may simply upload the new script to the Sentry Firewall and run it. But just make sure that you copy the final draft of your script to your configuration floppy so that it will be copied to the ramdisk and run at boot-time.

5.3 Using Webmin with the Sentry Firewall CD

As of version 1.5.0-rc3 Webmin(http://www.webmin.com/) is available on the CD. Among many of the other default modules available with Webmin - of which not all have been fully tested - Webmin includes two modules for generating and managing your firewall setup. These modules are located in the "Networking" section of the Webmin interface. In this section you will see the "Linux Firewall"

and "Shorewall Firewall" modules, either of which are available for your use.

The addition of Webmin also adds four new configuration directives for your sentry.conf file -

```
    start_webmin = <enable | disable>           ## enable|disable webmin.  Default ==
disable.    webmin_config = <path/to/config>            ## Main webmin
config(/etc/webmin/config).    miniserv.conf = <path/to/miniserv.conf>     ## Config file
for webmin http(s) daemon.    miniserv.pem = <path/to/miniserv.pem>       ## SSL cert. for
webmin http(s) daemon.                                                  ## An SSL cert. will
be created by rc.webmin if                                                  ## one is not
specified.    miniserv.users = <path/to/miniserv.users>   ## Password file used for
webmin.                                                 ## Default user:pass is
sentry:SENTRY.                                                  ## NOTE: If this file is not
replaced webmin                                                  ## will NOT start!
```

Note1: By default the miniserv HTTP daemon listens on port 11111 on the loopback interface. You will need to edit the miniserv.conf file to change this behavior.

Note2: The modifications made by these web interface tools are, of course, not permanent. Any files altered will need to be placed on a floppy or on a remote server and declared in your sentry.conf file as explained in previous sections.

As of version 1.5.0-rc3 the Shorewall(http://www.shorewall.net/) firewall scripts are available on the Sentry Firewall CD. Webmin also comes with a module to configure and set up Shorewall, although Shorewall can be configured manually as well. Shorewall utilizes a number of configuration files located in "/etc/shorewall". The sentry.conf file recognizes the "shorewall.conf" configuration directive, but if any of the other configuration files in "/etc/shorewall" need to be replaced you will need to do so manually using the "|=" copy directive.

5.4 Other Sample Firewall Scripts and Tools

Sample firewall scripts can be found in the "/SENTRY/scripts/firewall" directory on the CD. These are just a few firewall scripts I found on the Internet and have put here for your convenience. If you do a search on google or freshmeat.net you will probably find several others pretty easily.

I have also added "Easy Firewall Generator" (http://easyfwgen.morizot.net/) and "IPTables Script Generator" (http://iptables.linux.dk/) to the CD. These are PHP scripts that can assist you in creating a ruleset for your Sentry Firewall CD system. In order to view these you will need to start the Apache web server on a running Sentry Firewall CD system, and then direct your browser to the IP address of your Sentry Firewall. The scripts should be available in the "firewall" directory.

Please note that these web-based scripts will often generate a script for you, but you will still need to take that generated script and place at on a floppy or on a remote server and edit the "rc.firewall" directive in the sentry.conf file to point to your new script.

5.5 Links to Other Firewall Resources

Netfilter HOWTO
Netfilter FAQ
Netfilter Tutorials

If there are any other resources you think I should add to this section, please email me at Obsid@Sentry.net.

6. Setting Up Snort

6.1 Starting Snort

The Snort IDS is available on the Sentry Firewall CD. This allows a Sentry Firewall system to act as an IDS sensor, and either store log data on a local hard drive, or send it to a remote log/mysql server. With the updates to version 1.5.0-rc4, snort is started in a chroot jail in the "/var/chroot/snort" directory. This provides an added layer of protection against a compromise of the running snort process.

Snort is started via the /etc/rc.d/rc.snort file in the SENTRYCD(slackware) branch, and via /etc/init.d/snort in the SENTRYCD-{DEB,RH} branches. Please take a look at this file if you wish to customize the options passed to snort at runtime. By default, log data is kept in tcpdump format, and is stored in the "/var/chroot/snort/var/log/SNORT" directory - "/var/log/SNORT" is a symlink to this directory.

6.2 Customizing Snort Rules

Snort rules are kept in the "/etc/snort" directory. The snort rules are basically the signatures and rules snort uses to match against IP traffic and create logs, alerts, etc. These files are kept current in each release with those available at snort.org. For many setups, however, it will likely be necessary to add/remove/customize the snort rules. To do so, simply edit the file(s) you need to change and place them on a floppy or a remote server, and then use the "=" copy directive in your sentry.conf file to replace each file you altered. For example:

```
/etc/snort/exploit.rules = /floppy/snort/exploit.rules
or
/etc/snort/exploit.rules = scp://user:pass@myserver.com/sentrynode/snort/exploit.rules
```

6.3 The snort.conf File

The snort.conf file is used as the primary configuration file for snort. Again, in many setups it will likely be necessary to customize this file for your network. Once you have tuned the file to suit your environment you can place this file on a floppy or on a remote server and use the ’snort.conf’ configuration directive in your sentry.conf file to declare its location.

Please visit Snort.org and read the Snort Documentation for more information on configuring and using snort on your network(s).

7. Setting Up BIND

7.1 Starting BIND

Since using the Sentry Firewall CD as a DNS server as well as a router or firewall has proven to be a popular choice, I am including some basic info on utilizing BIND. Note that, except for minor variations on where the configuration and zone files are located, start/stopping the daemon and the general configuration of BIND should not vary too much compared to other Linux systems.

Currently, BIND version 9 is the primary DNS daemon available on the CD. Several of the branches, however, will still contain a statically linked BIND 8 for your use as well. Note, however, that BIND 8 will eventually be deprecated and removed from the CD.

In the SENTRYCD branch one can start the server by running "/etc/rc.d/rc.named start". You can run this via the command line, or edit rc.inet2 and uncomment the appropriate linesto allow the server to start at boot-time. You would then, of course, need to place themodified rc.inet2 file on a floppy or a remote server and use the ’rc.inet2’ directive in your sentry.conf file to declare its location. In short, the rc.named file takes the followingarguments:

- **/etc/rc.d/rc.named start** -- Start BIND 9 in chroot environment.
- **/etc/rc.d/rc.named start** -- Restart BIND 9 in chroot environment.
- **/etc/rc.d/rc.named reload** -- Reload zone/configuration files(BIND 9 only).
- **/etc/rc.d/rc.named stop** -- Stop the named daemon(BIND 8/9).
- **/etc/rc.d/rc.named start_named8** -- Start BIND 8 in chroot environment.
- **/etc/rc.d/rc.named restart_named8** -- Restart BIND 8 in chroot environment.

In the SENTRYCD-DEB branch, one can use the command "/etc/init.d/bind9 start" to start named. The "/etc/init.d/bind9" init script also take the arguments "stop", "reload", and "restart" - all of which are pretty self explanatory.

7.2 BIND Configuration

Both versions of BIND are run in a chroot environment, located in "/var/chroot/named". The chroot environment generally looks like the following:

```
/var/chroot/named/
    +-- dev
    +-- etc
    |    +-- namedb
    |         +-- slave
    +-- var
         +-- log
         +-- run
         +-- named <-- symlink to ../etc/namedb
    +-- usr
         +-- sbin
```

The named.conf file is located in "/etc" and "/var/chroot/named/etc". The ’named.conf’ configuration directive automatically places the named.conf file in both locations. Zone files are generally kept in "/var/chroot/named/etc/namedb" and "/var/chroot/named/etc/namedb/slave". The "usr" and "usr/sbin" directories store the statically linked BIND 8 version of the ’named’and ’named-xfer’ binaries, and are not applicable on the SENTRYCD-DEB branch.

7.3 Links to Other Resources

The following are links to other resources to help you work with and configure the BIND DNS server. Please also feel free to post to the mailing list if you run into any Sentry Firewall CD specific configuration issues.

- ISC.org BIND Homepage
- DNS HOWTO
- Chroot BIND9 HOWTO
- Chroot BIND8 HOWTO

8. Troubleshooting

8.1 Booting Problems

Booting problems are generally rare, and generally only occur on old and buggy, or somehow non-compliant hardware. Booting problems can be associated with a number of problems, depending upon at what point during the boot process the failure occurs. The following are possible causes of failure when booting from a CD.

- Old or buggy BIOSes that do not fully support the eltorito standard. System may fail to load the isolinux bootloader or the kernel.
- Problematic CDROM drives can cause various problems when booting the CD. System may not consistently boot, and will generally have trouble accessing files on the CD.
- Damaged CD, obviously can cause a number of problems, similar symptoms as above.
- Insufficient hardware resources. Please see the "Minimum Requirements" section of this howto for more information on what is required to boot the CD.
- In the case of booting the Sentry Firewall CD, old or buggy floppy disk drives or damaged floppy disks can also result in serious problems, including corruption of the data on the floppy disk. The inability for the configuration scripts to read and parse files contained on the floppy disk can seriously inhibit the capability of the system to configure itself properly.

In general, hardware issues cause the majority of problems during the boot process of the Sentry Firewall CD, and may not always be easy to diagnose. Generally, the first step in debugging a general boot problem is to try and boot another CD in the same machine to attempt to rule out a hardware problem. And then attempt to boot the Sentry Firewall CD in another machine to attempt to rule out damage to the CD. If both these tests produce no negative results, then perhaps swap out the CDROM drives in the two machines, if possible, and do the test again. Then perhaps check out the general mailing list(mentioned below) for further assistance.

8.2 Configuration Problems

This section deals with configuration problems with the "sentry.conf" file. The sentry.conf configuration file, as mentioned in previous sections, tells the configuration scripts what to do during boot time to configure the running system. Syntax errors in the script can cause a file to be misplaced, or for the directive to not be parsed at all.

Error messages during the boot process of the Sentry Firewall CD can help greatly in diagnosing potential syntax or other types of errors. So watch the CD boot and write down any error messages that may pop up. Also, during bootup a logfile detailing the configuration process is created at "/var/log/SENTRY_LOG". If you can log in to the system after it has booted, then take a look at this file for any obvious error messages.

8.3 Frequently Asked Questions

A FAQ is currently being maintained on the Sentry Firewall website, it can be accessed via the following URL: http://Sentry.SourceForge.net/files/FAQ.html.

8.4 Mailing List

Thanks to SourceForge.net, there are mailing lists available for the Sentry CD. You can look through the archives, or subscribeto the general mailing list to ask questions or make comments. The following are links forthe general Sentry-Users mailing list. Other mailing lists are listed at SentryFirewall.com.

- Subscribe to Sentry-Users
- Sentry-Users Archives

9. Building a Custom Sentry CD

9.1 Introduction

This section will attempt to describe how to create a custom Sentry Firewall CDROM. I may not be able to go into every detail, but at the very least I will try and provide a thorough overview of the CD creation process.

9.2 The development system(How I do it)

My development system consists of two separate Linux installations of the same distribution; Slackware/Redhat/Debian, depending on what branch I'm working on. My environment looks something like the following:

- / (root)
  Contains the running development system. This is often a pretty full featured Linux installation where I can compile and work on stuff. You can install as much, or as little, as you like on this main system.
- /SENTRYCD-DIR
  This directory contains another Linux installation of the same distribution as the main system. This installation generally has far fewer packages installed - no X stuff or compiling tools. Many of the programs and files contained within this system will end up on the Sentry Firewall CD.
- /SENTRYCD-DIR/CD-FW
  This directory contains those files and directories that will become the actual Sentry Firewall CD ISO image. Much of the files in this directory are pulled from the /SENTRYCD-DIR directory, by hand at first, and then later with a script when updates are required.

So the Sentry Firewall CD ISO is basically an exact copy of the contents of the /SENTRYCD-DIR/CD-FW directory. I use the 'mkisofs' utility to create the ISO image. There is also a script available on the CD called mkiso.sh that can assist you in creating a working CD image.

Unfortunately, it is not currently possible to edit an iso9660 image directly. So, if you simply want to modify the contents of the CDROM and create a new image you can try doing the following:

- Mount the Sentry Firewall CD somewhere, let's say on /cdrom.
  For example: **mount /dev/scd0 /cdrom**
- Make a directory to store the contents of the CDROM.
  For example: **mkdir /CD-FW**
- Copy the contents of the CDROM to the new directory.
  cp -Rdp /cdrom/* /CD-FW/
- Now we need to remove those 'TRANS.TBL' files, these were created by the mkisofs program when the ISO was created.
  find /CD-FW -name 'TRANS.TBL' -type f -print | xargs rm -f
- Unmount the CDROM.
- You may now customize the contents of the /CD-FW directory. Then use the mkiso.sh script to create a new ISO image. More information on creating an ISO image can be found later on in this chapter.

9.3 The RAMDisk Image

That's all nifty, but now comes the hard part - making the ramdisk. If you take a look at the /isolinux directory on the CDROM, you will see a bunch of files -

- **initrd.img** - This is the gzip compressed ramdisk image.
- **isolinux.bin** - This file acts as a boot record for the CD(important).
- **isolinux.cfg** - This is the configuration file for isolinux.
- **message.txt** - This contains the message displayed at boot time.

There is also a directory in /isolinux called "kernels". This is where the Linux kernel(s) reside for the Sentry Firewall CD. You can add your own custom kernel(s) or replace the default ones. If you do so you may also need to customize the isolinux.cfg file to declare your new kernels and options. More information about configuring and using isolinux can be found at http://syslinux.zytor.com/.

- **Note:** Syslinux is a DOS-based bootloader, which means you're stuck with the 8.3 naming scheme when naming your kernel and isolinux configuration files.

To look at and modify the initrd.img image, do something like the following:

```
blah@wherever:~$ cp /cdrom/isolinux/initrd.img /tmp/initrd.img.gz
blah@wherever:~$ gzip -d /tmp/initrd.img.gz
blah@wherever:~$ mount -o loop /tmp/initrd.img /MOUNT_POINT
```

You may then cd to /MOUNT_POINT and edit the files on the rootdisk. Once you are finished you can then unmount and gzip the initrd.img file and place it back in the isolinux directory.

mkrootdsk.sh

Although the above method will allow you to customize the rootdisk to your liking, this is not the method I use to create the rootdisk. I use a script called 'mkrootdsk.sh'. This is a bash shell script that formats/mounts the disk image, and then creates or copies the files to the disk image as needed.

If you would like to attempt to use the mkrootdsk.sh script please be sure to read through it first, as it tends to be a bit hacky at times. It runs perfectly on my development system, but may not run well at all on yours. The output from the script should look something like the following:

```
    Sentry Firewall CD-ROM: mkrootdsk.sh
    Copyright (C) Stephen A. Zarkos, Obsid@Sentry.net
    Ok, let's get to it...

    [+] Creating /root/rootdisk/root... Done.
    [+] Ok, starting to copy stuff to the rootdisk...
    [+] Making directories: root dev proc etc sbin bin lib mnt mnt1 mnt2 mnt3 mnt4 opt
cdrom floppy tmp    tmp/drivers var initrd... Done.
    [+] Copying /dev files... Done.
    [+] Working in /var... Done.
    [+] Working in /home... Done.
    [+] Working in /bin... Done.
    [+] Working in /sbin... Done.
    [+] Working in /lib... Done.
    [+] Working in /etc... Done.
    [+] Building drivers-2.4.tar.gz(network config support).
        [+] Using /cdrom/lib/modules/2.4.25GENERIC.
    [+] Tar/Gzipping /root/rootdisk/root... Done.
    [+] Zeroing out file: /root/rootdisk/initrd.img... Done.
    [+] Creating ext2 file system on /root/rootdisk/initrd.img... Done.
    [+] Mounting initrd.img on /root/rootdisk/mnt... Done.
    [+] Copying files to rootdisk... Done.

    [+] /root/rootdisk/initrd.img is still mounted, do you want me
        to unmount it? (y/n) y
        [+] Unmounting /root/rootdisk/mnt... Done.
        [+] Gzipping /root/rootdisk/initrd.img... Done.

    Location of new rootdisk --> /root/rootdisk/initrd.img
```

9.4 Making the ISO Image

I use called 'mkiso.sh' to create the ISO image. The script simply declares a few variables and runs the 'mkisofs' utility. The command I normally run looks like the following:

```
me@mybox:~# cd /SENTRYCD-DIR/CD-FW
me@mybox:/mnt/CD-FW# mkisofs -o sentrycd.iso -R -V "Sentry Firewall CD [v1.x.x]" -v \
   -T -d -D -N \
   -b isolinux/isolinux.bin \
   -c isolinux/eltorito.cat \
   -no-emul-boot -boot-load-size 4 -boot-info-table \
   -A "Sentry Firewall CD v1.x.x" .
... <lots of output> ...
```

And that's it, I burn the ISO to a CD and test it. For reference, the following files are available on the CDROM and online at http://www.SentryFirewall.com/

- /SENTRY/scripts/MK-CD/mkrootdsk.sh (builds the rootdisk)
- /SENTRY/scripts/MK-CD/mkiso.sh (builds final ISO image)
- /SENTRY/scripts/MK-CD/record-cd.sh (burns the ISO to a CD)

10. More About the Sentry Firewall Project

10.1 Supporting the Project

There are various ways one can support this project. The easiest and most common way is to simply utilize the system in a test or production environment and send me suggestions, bugs, or other such feedback. For those interested in assisting with the enhancement of any of the Sentry Firewall CD branches, please check out the TODO file located in '/SENTRY/docs/TODO' on the CD image, or online at www.SentryFirewall.com.

I do, on occasion, make the Sentry Firewall CD available for purchase. I also accept donations including hardware, software, currency, or anything else that you feel can help. Revenues from such donations or CD sales will help support the continued development of the project. If you are interested in supporting this project please feel free to email me at Obsid@Sentry.net.

10.2 About the Author

The Sentry Firewall project has only ever had a single developer, Stephen Zarkos(me) of Bellevue, Washington(USA). I began work on the project around April of 2000, probably ruining 200 CD-Rs before I got my first stable Sentry Firewall CD. And for the last two years I have continued to develop, enhance and maintain the project - give or take a few months here and there while I took a short hiatus(marriage, education, etc).

From the beginning, this project has proven to be quite popular, and has received a great deal of support and feedback from its loyal users. This kind of support has proven invaluable,and has kept me motivated to continue to develop this project. There is nothing I would ratherdo right now than work on and enhance this system, however since I do not get paid to developthis project, it is only a part-time endeavor. Even so, the positive comments and feedback Ireceive has without a doubt made this the most enjoyable project I have ever been a part of.

10.3 Contacting the Author

Mailing Address:
Sentry Firewall CD Project
C/O Stephen A. Zarkos
P.O. Box 6133
Bellevue, WA 98008

Email: Obsid@Sentry.net