Tlhis chapter339.7(459.6[yoalluse)hist-263339.7(ishicker’(hi339.7(ultimateta339.7(goal)-339.7(i(hi339.7(nothi

Internet) they'd like to make available, or maybe they just want to store MP3s or their MPEGs of questionable moral content.

They Want Your Data Crackers may want your business's trade secrets for personal use or to sell. Or they may want your bank records or credit card numbers.

They Want to Destroy They may just want to wreak havoc. The sad fact is that some people in than that they can. Maybe they think it is cool, or maybe they have destructive personali-

If that operating system is proprietary—that is, owned and controlled by one company, person, or entity—this usually means that the software is

are strong words from a company that makes a profit on closed source, proprietary software and one that considers Open Source software a real threat—so real, in fact, that a Microsoft employee wrote a document that is now known as the "Halloween Document" (because it was leaked and published publicly on Halloween 1998; see http://www.opensource.org/halloween/). In that document, the Microsoft employee admitted that Open Source software is a threat to proprietary software and laid out the Microsoft strategy to fight the emergence of Open Source software.

LINUX USERS

Since Linux is a

printing functions (`lp` stands for line printer). The actual system users on your machine depend on your Linux distribution and the software you have installed.

Linux Groups

Linux implements the concept of [illegible]. A group is a collection of one or more users. It is often convenient to collect a number of users together to define properties for the group, suc927.s onvtrolson ywhatthe y can or cannot access

against a boss (that was never sent, of course), or an idea for a new dot-com business (ripe for stealing).

□

If you don't like all this octal number crunching, you can use symbolic notation for your `umask` setting if you are using bash. The

```
$ /jdoe$/umask -S
u=rwx,g=rwx,o=r
$ jdoe$ umask u=rwx,g=r,o=
$ jdoe$ umask -S
u=rwx,g=r,o=
$ jdoe$ umask
037
```

As an administrator, you can also add `umask` changes in the global file `/etc/profile` to have it apply to all users.

strictions are present in other UNIX-like systems as well, but they may be foreign ideas to our underprivileged Windows brethren.

Signals

In Linux, users can send *al* to processes. A signal is a message sent from one process to another. A common signal to send to a process is the TERM, or terminate, signal. This signal is sent to a process to force the process to terminate and is often used to kill a runaway process. This example shows a user killing a process:

```
jdoe$  kill -TERM 13958
```

This command sends the TERM signal to the process with process ID 13958 .

Here is an example using killall :

```
root#  killall -HUP httpd
```

This killall

In this example, chroot will change root to the /usr/local/convict directory and then run your program, /bin/convict. Because the chroot is performed first, the program /bin/convict actually resides in /usr/local/convict/bin/convict on the real filesystem.

Setting Up a chroot Jail Directory

One problem with chrooting your software is that all the programs and libraries that are needed by your software must be copied into the chroot directory, which we usually call a chroot jail (since you can't ever get outlTf1t(utlTf1twsince)-42oftwacejail alTf1twsincethe

Most Linux distributions do handle man pages safely, because the `troff` program, which is called b

Format String Countermeasures

Many format string attacks use the same principle used with buffer overflows—overwriting the function's return call—and can thus be prevented by the buffer overflow

You might think of replacing all the stat calls with this instead:

```
# just delete - don't bother checking unlink $FILE
```

However this is still vulnerable to a symlink attack before the open occurs, allowing the attacker to overwrite a file as root, or between the open and the chown, which would allow the attacker to take ownership of any file on the system.

Use Atomic System Calls

The best way to avoid race conditions is to use functions that are atomic (system calls that execute uninterrupted inside the kernel). The `open()` system call can take an argument that says “only create this file if it does not exist.” Unfortunately you can’t use this with Perl’s standard `open` function, but you can with `sysopen`. Our code becomes:

```
#!/usr/bin/perl
# runForward - no longer vulnerable to a race condition.  Uses
# sysopen to avoid symlink open attack, and fchown system call
# to avoid symlink race between create and chown.

use POSIX;  require "syscall.ph";

($username,$email) = @ARGV;
($uid,$gid,$home) = (getpwnam($username))[2,3,7]      || die

$FILE = "$home/.forward";
unlink $FILE;   # if it fails, sysopen will catch it.

sysopen( FORWARD, $FILE, O_RDWR|O_CREAT|O_EXCL, 0600) || die;
syscall(&SYS_fchown, fileno(FORWARD), $uid, $gid)==0  || die;

print FORWARD "$email\n";
close FORWARD;
```

There are many pn2-238.2ln