



# Ethical Hacking and Countermeasures

Version 6



**Module XXII**

Linux Hacking

Bryan was a network administrator with top-shopper.com, a small online shopping portal. He was an expert on Windows Platform but lacked in other OS. Due to strategy changes the portal was shifting from Windows to Linux systems, and because of time and human resource constraints Bryan was entrusted with the responsibility of installing Linux in their systems. While installing Linux, Bryan selected default options as he was not familiar with the kernel components of Linux. Within a week, the portal was hacked and their systems were taken off the Internet.

**What went wrong?**

PC World / Business Center / Operating Systems / Linux / Unix / News

## Attack Against Linux Apache Servers Intensifying

Ellen Messmer, Network World

20 recommend 

Tuesday, January 22, 2008 4:00 PM PST

A mass attack ongoing for the past month against [Linux Apache Web servers](#) has become increasingly successful because its break-in method makes use of an automated password and installation process, according to a [security](#) researcher monitoring its progress.

[Don Jackson](#), senior security researcher at [SecureWorks](#), says the attack, which was first thought to have compromised several hundred Web sites, has hit at least 10,000. He says the attack relies on making use of stolen passwords to Linux Apache servers by automating the installation process to force it to serve up attacks against vulnerabilities on [Windows](#) clients.

"The Web server ends up serving up vulnerabilities from 2006 related to Windows malware," Jackson says. "The whole attack is very mysterious. It's based on a botnet but it doesn't match the Russian and Chinese groups and may be Western Europe or North American."

The attack, which makes use of the well-known Rbot and Sdbot malware, targets at least nine software vulnerabilities associated with [QuickTime](#) exploits, [AOL SuperBuddy](#) and [Yahoo! Messenger](#) to try and compromise Windows-based desktops. [SecureWorks](#) says most antivirus vendors can detect the malware.

The ingenuity is that the attacker has managed to install code that modifies Apache memory to monitor requests and inject the script tag, script contents or the Rbot executable, according to SecureWorks. Some Linux Apache network managers are finding it hard to clean their servers of the attack code, he notes.

Source: <http://www.pcworld.com/>

## Excuse me sir: there's a rootkit in your master boot record

By [Dan Goodin in San Francisco](#)

Published Wednesday 9th January 2008 05:34 GMT

**Security mavens have uncovered a new class of attacks that attach malware to the bowels of a hard drive, making it extremely hard to detect and even harder to remove.**

The rootkit modifies a PC's master boot record (MBR), which is the first sector of a storage device and is used to help a PC locate an operating system to boot after it is turned on. The result: the rootkit is running even before Windows loads. There have been more than 5,000 infections in less than a month, researchers say.

"Master boot record rootkits are able to subvert the Windows kernel before it loads, which gives it a distinct stealth advantage over rootkits that load while Windows is running," said Matthew Richard, director of the rapid response team for iDefense, a security provider owned by VeriSign. "It gives it a great stealth mechanism that allows it to persist even after removal." Such rootkits can even survive reinstallation of the operating system, he said.

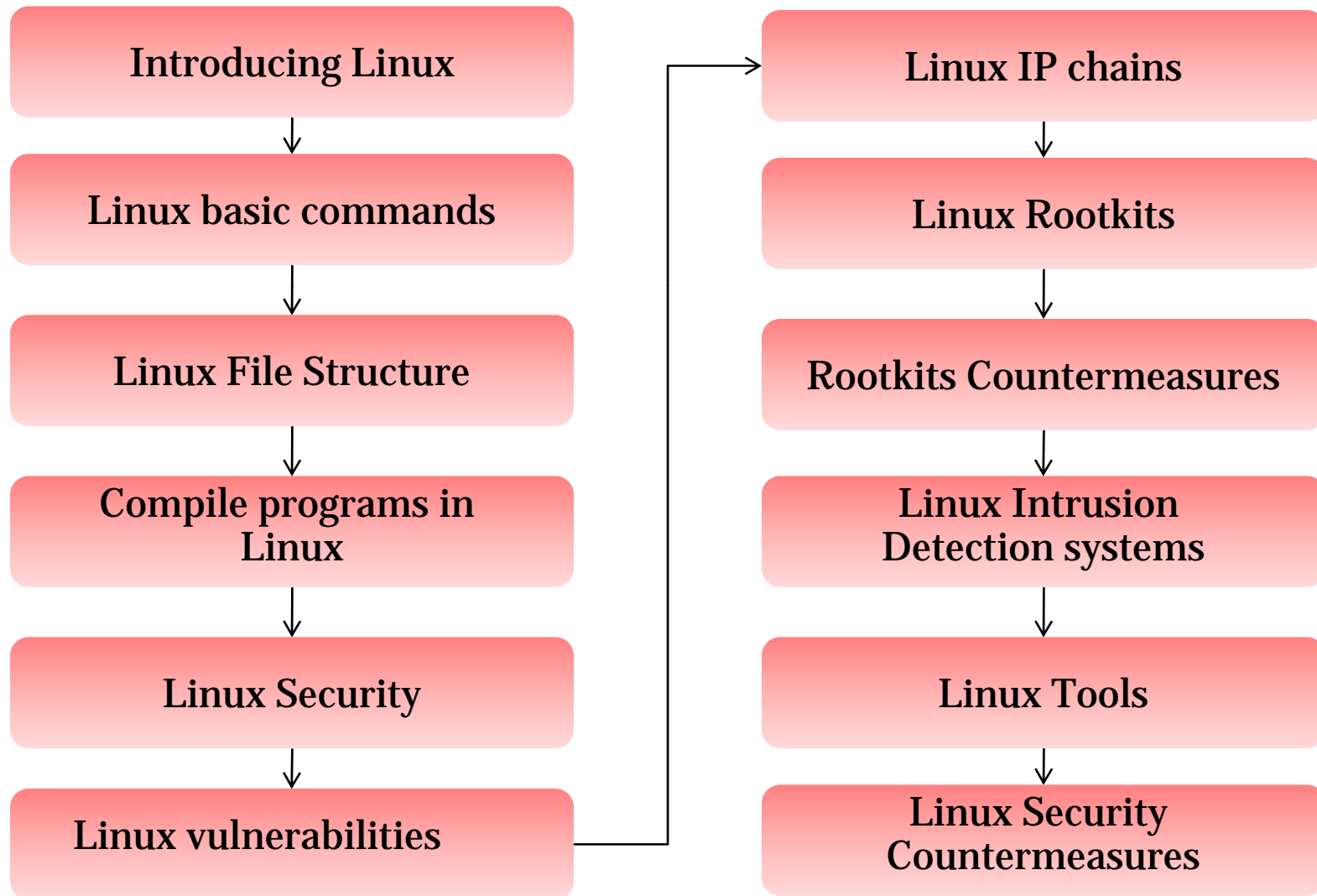
Because the rootkit lurks deep within the hard drive, well below the operating system, most antivirus programs don't detect the malware. Symantec's antivirus program is an exception, however. It labels the pest Trojan.Mebroot, according to Javier Santoyo, a senior manager for emerging technologies at Symantec.

Source: <http://www.channelregister.co.uk/>

This module will familiarize you with:

- Linux
- Basic Commands in Linux
- Linux File Structure
- Compiling Programs in Linux
- Linux Security
- Linux Vulnerabilities
- Linux IP chains
- Linux Rootkits
- Rootkit Countermeasures
- Linux Intrusion Detection systems
- Tools in Linux
- Linux Security Countermeasures

# Module Flow



Majority of servers around the globe are running on Linux/Unix-like platforms

Linux is easy to get and easy on the wallet

There are many types of Linux-Distributions/Distros/ Flavors, such as Red Hat, Mandrake, Yellow Dog, Debian, and so on

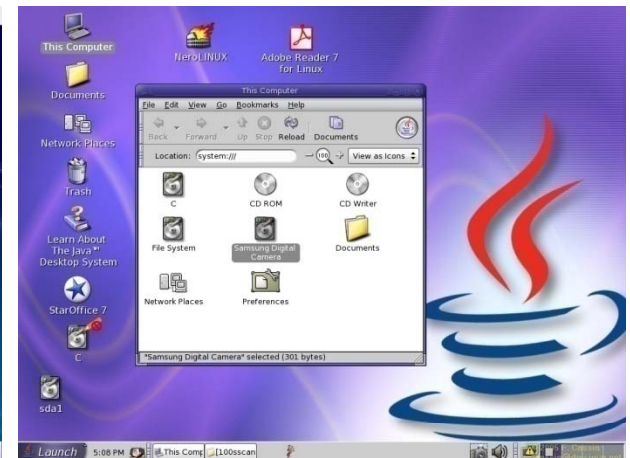
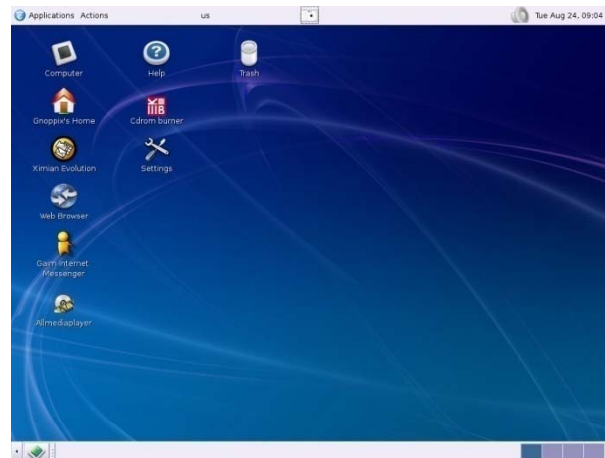
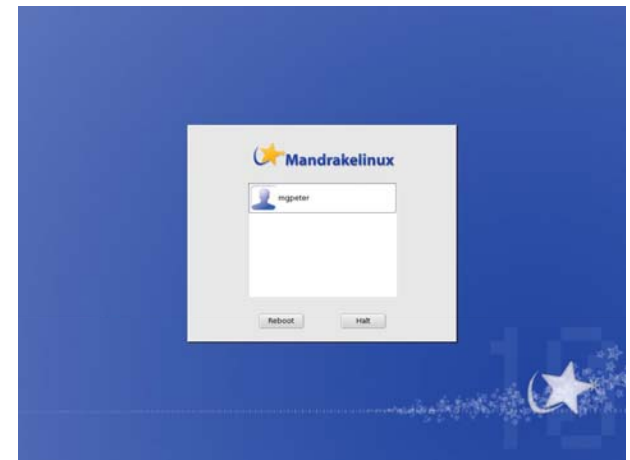
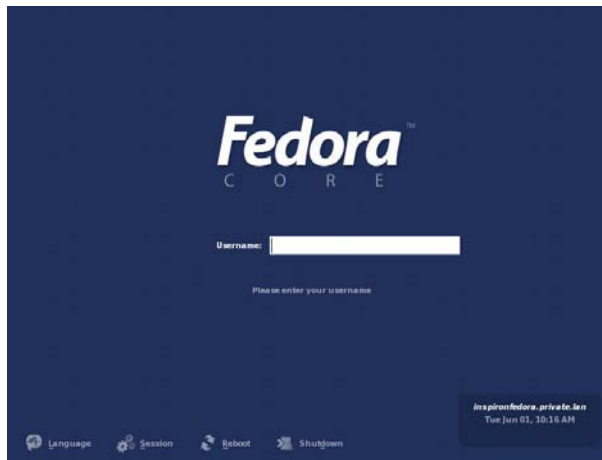
Source code is available in Linux

Linux is easy to modify

It is easy to develop a program on Linux

# CEH Linux Distributions

Certified Ethical Hacker



Source: <http://distrowatch.com>



Aliased commands can pose a security threat if used without proper care

Linux shell types - `/sh`, `/ksh`,  
`/bash`, `/csh`, `/tcsh`

Linux user types, groups, and permissions

Overview of linux signals, logging and `/etc/securetty`



# Linux Live CD-ROMs

A LiveCD is an operating system (usually containing other software as well) stored on a bootable CD-ROM that can be executed from it, without installation on a hard drive

Knoppix Live CDs are widely used in the Linux community

It is completely customizable



Source: <http://www.knoppix.org>

Copyright © by **EC-Council**

All Rights Reserved. Reproduction is Strictly Prohibited

# Basic Commands of Linux: Files & Directories

Everything is a file

256 characters maximum

They are case sensitive

Extension not necessary



## Special characters

- Begin with . (period)
- Don't use /, ?, \*, -
- Avoid spaces; use underscores instead

## File system

- Hierarchical tree
- No drive letters
- Starts at root with /

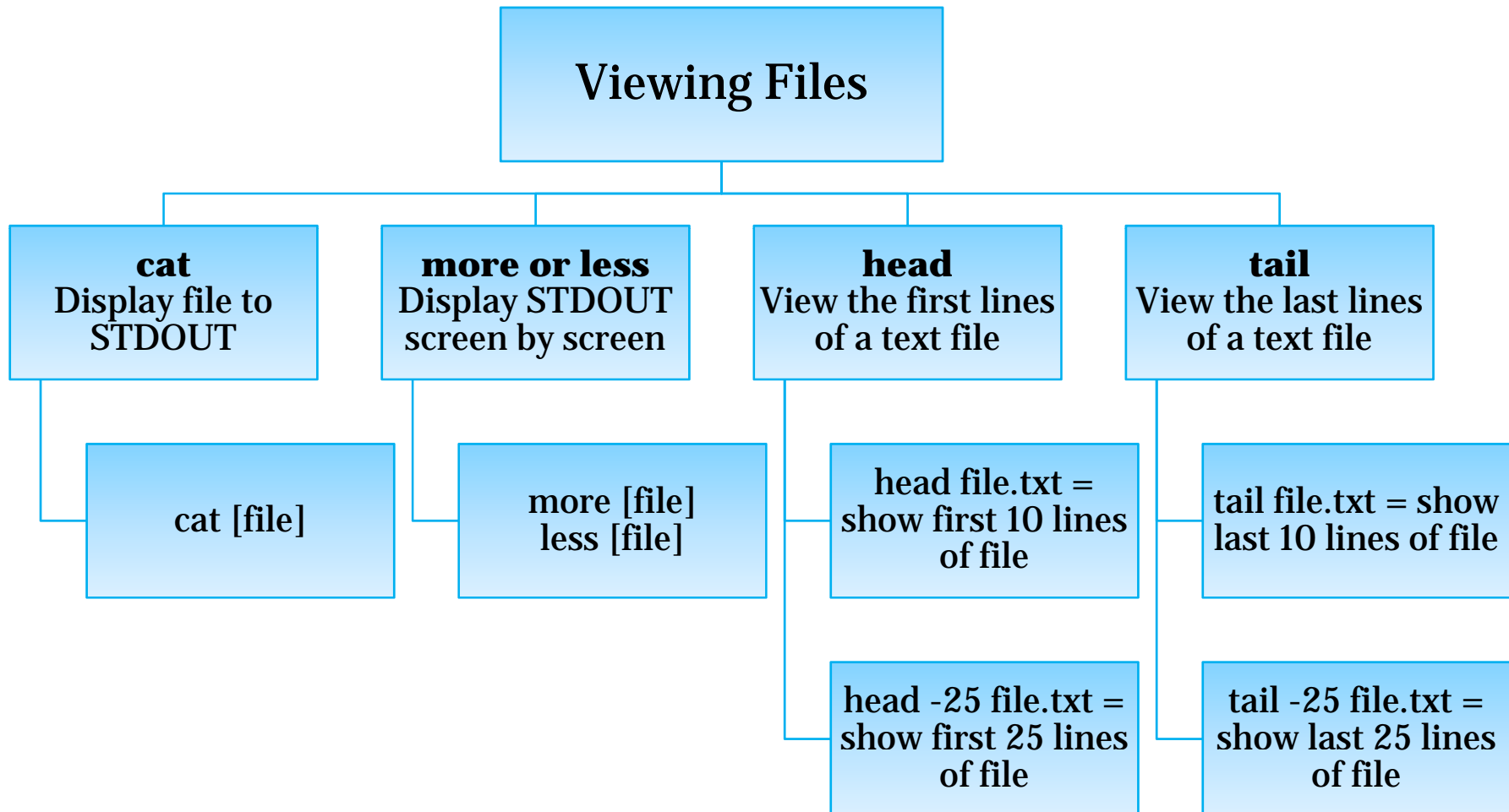


## Getting Information

- **man**
- **man [command]**
- **Within man:**
  - spacebar/f = forward
  - b = back
  - q = quit
  - / = search forward
  - ? = search backward
  - n = repeat search



# Basic Commands of Linux (cont'd)



## Getting Around

- `cd . cd ~`
- `cd . cd ..`
- `ls . ls -a`
- `ls -l`



## Files & Directories

- `cp`
  - `cp file newfile`
- `mv`
  - `mv file newfile`
- `mkdir`
  - `mkdir [directoryname]`
- `rm`
  - `rm file`
- `find`
  - `find / -name *gnome* -print`



# Linux File Structure

```
lrwxrwxrwx # owner group size_in_bytes last_modified_date_&_time filename.txt  
^ \_ / \_ / \_ /  
| v v v  
| | | |  
| | | World permissions  
| | |  
| | | Group permissions  
| | |  
| | | Owner permissions  
|
```

Type of file:

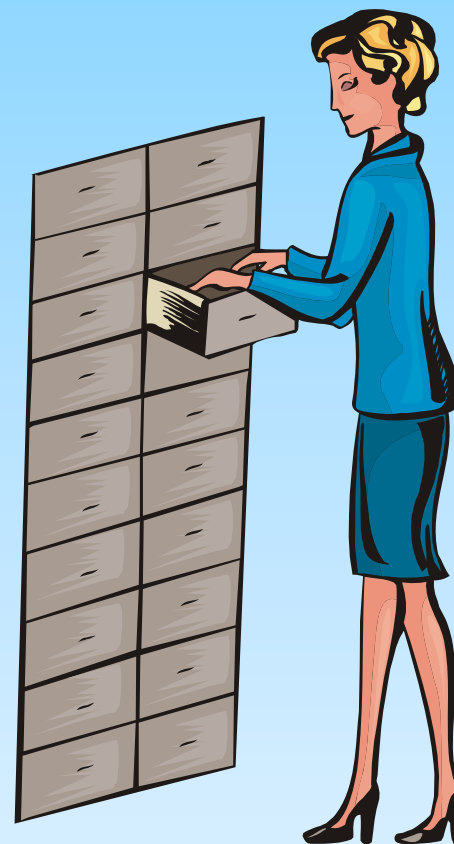
= file

l = link

d = directory

b = block device (disk drive)

c = character device (serial port or terminal)



# Linux Networking Commands

`arp`

- Command is mostly used for checking existing Ethernet connectivity and IP address

`ifconfig`

- Command line tool configures or checks all network cards/interfaces

`netstat`

- Summary of network connections and status of sockets

`nslookup`

- Checks the domain name and IP information of a server

`ping`

- Sends test packets to a specified server to check if it is responding properly



# Linux Networking Commands (cont'd)

`ps`

- Lists all existing processes on the server

`route`

- Lists the routing tables for your server

`shred`

- Deletes a file securely by overwriting its contents

`traceroute`

- Traces the existing network routing for a remote or local server

`ps`

- The `ps` command displays all of the existing processes

# Directories in Linux

**bin**

- Binary files (executables)

**sbin**

- System binary files (to be used by administrators)

**etc**

- Configuration files

**include**

- Include files

**lib**

- Library files

**src**

- Source files

**doc**

- Document files

**man**

- Manual files

**share**

- Shared files

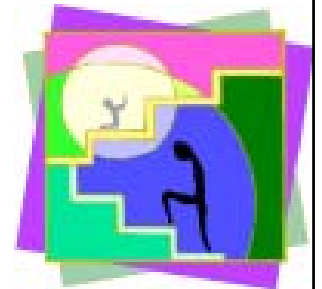


# Installing, Configuring, and Compiling Linux Kernel

Download the latest kernel from [www.Linux.org](http://www.Linux.org)

## Step1

- login as 'root'
- 'cp linux-2.4.2.tar.gz /usr/src/'
- 'cd /usr/src/'
- Check the source of old kernel in `/usr/src/linux`
- Move the current version 'mv /usr/src/linux linux-X.X.X' as a backup for future use
- 'tar -zxvf linux-2.4.2.tar.gz'
- Move new Kernel source, 'mv /usr/src/linux /usr/src/linux-2.4.2'
- Create a link to it 'ln -s /usr/src/linux-2.4.2 /usr/src/linux'



# Installing, Configuring, and Compiling Linux Kernel (cont'd)

## Step 2

- Configure the Kernel
  - `cd` to your kernel source directory in `/usr/src`
  - Type `make menuconfig` if you prefer text mode, but `xconfig` is recommended

## Step 3

- Go back to your command line and type: `make dep` for kernel compilation

## Step 4

- Clean all the files (.o, or object files) created during compilation
  - `Make clean`

## Step 5

- Create a bootable Linux image (actual Linux file)
- Make bzImage
- Make new modules for installation
- Make modules
- After finishing compilation type
- Make modules\_install
- Move the BzImage file to the location of the kernel
- `mv/usr/src/linux-2.4.17/arch/i386/boot/bzImage /boot/vmlinuz-2.4.17`



## Step 6

- Locate the new file to linux boot manager LILO
- Edit the file `/etc/lilo.conf` , add these lines
- `image=/boot/vmlinuz-2.4.17`  
`label=linux-2.4.17`  
`root=/dev/hda3`  
`read-only`
- Save the `lilo.conf` file
- Run the lilo program `/sbin/lilo`
- Reboot the machine



# Installing, Configuring, and Compiling Linux Kernel (cont'd)

## Linux Kernel Configuration: File systems

```
File systems --->
<*> Reiserfs support
<*> Ext3 journalling file system support
<*> JFS filesystem support
<*> Second extended fs support # this is the ext2 filesystem
<*> XFS filesystem support
```

## Linux Kernel Configuration: Ethernet Controller

```
Device Drivers --->
Networking support --->
Ethernet (10 or 100Mbit) --->
EISA, VLB, PCI and on board controllers
<*> RealTek RTL-8139 PCI Fast Ethernet Adapter support
```

# How to Install a Kernel Patch

Download the Linux kernel patch from [www.linux.org](http://www.linux.org)

Copy the downloaded kernel to `/usr/src/linux` directory

Navigate to the downloaded directory `cd /usr/src/linux`

Extract the patch into the `/usr/src/linux` directory using tar, gzip, etc.

A file named `patch-2.x.x` or `patch-2.x.x-yy` should be created in the `/usr/src/linux` directory

To apply the patch to the kernel, run `patch -p1 < patch-2.x.x` or `patch -p1 < patch-2.x.x-yy`



# Compiling Programs in Linux

GCC is a command line based compiler

It can be used to compile and execute C, C++, and Fortran code

Many Linux installations include a version of GCC compiler by default

You can download the latest version from **<http://gcc.gnu.org>**



Most Linux hacking tools are written in C. When you download a hacking tool source, it will often be C or C++ source code. You do not need to know C++ programming to compile a program

- Here is a simple c++ code:

```
#include <iostream>

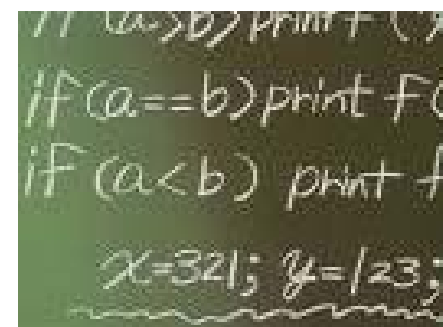
int main ()
{
    std::cout << "Hello,
world!\n";
    return 0;
}
```

- To compile the above code:

```
gcc -Wall hello.c -o hello
```

- To run the program:

```
./hello
```



- ◉ Read the program's [README](#) or INSTALL file for instructions on how to compile the program
- ◉ Sometimes, the compile command for some programs can be very long
- ◉ A Makefile is a command file for compiling programs
- ◉ For example, assume that you have a graphics program called face.cpp, and that the compile line is:
  - `g++ -o face face.cpp -L/usr/X11R6/lib -lm -lX11 -lgd -lg2`
- ◉ You would create a file named "Makefile", and in it you would put the lines
  - `face: face.cpp g++ -o face face.cpp -L/usr/X11R6/lib -lm -lX11 -lgd -lg2`
- ◉ Now to compile face, you would use the command:
  - `make` or
  - `make face`



# Make Install Command

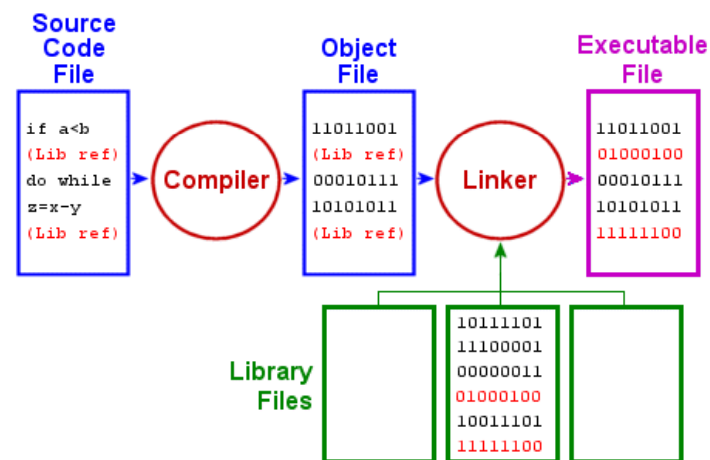
There are four commands to compile, link, and install a program

- `./configure` (may not have this)
- `make`
- `make install`
- `make clean`

The `make` utility handles compiling and linking

`make install` puts the compiled binary file in the proper (`/usr/local/bin`) subdirectory

`make clean` cleans up temporary files that were generated by the compiling and linking processes



# Linux Vulnerabilities

The number of unexploited vulnerabilities in the core Linux kernel is on the rise

The U.S. Computer Emergency Readiness Team, or CERT, reported that more Linux and Unix combined had more than 2,328 vulnerabilities, compared with 812 vulnerabilities for Microsoft Windows

Since the source code for any given Linux project is so widely circulated, it is available to every hacker in the world

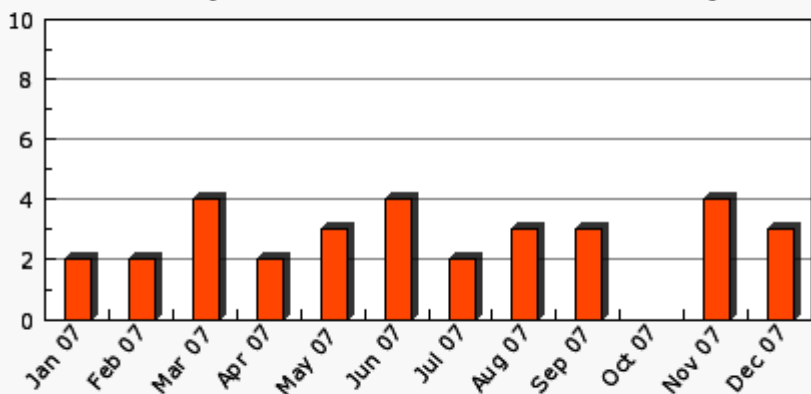


Vulnerabilities were announced in many packages, including:

- apache, balsa, bind, bugzilla, cdrecord, cfengine
- cron, cups, cvs, ethereal (many), evolution, exim, fetchmail (many), fileutils
- gdm, ghostscript, glibc, gnupg, gzip, hylafax, inetd, iproute, KDE, kerberos, kernel
- lprng, lsh, lynx, mailman, man, mozilla, mpg123, mplayer, mutt, MySQL, openssh, openssl
- perl, pine, PHP, postfix, PostgreSQL, proftpd, python, rsync, samba, screen, sendmail, snort, stunnel, sudo, tcpdump, vim, webmin, wget, wu-ftp, xchat, XFree86, xinetd, xpdf, and zlib

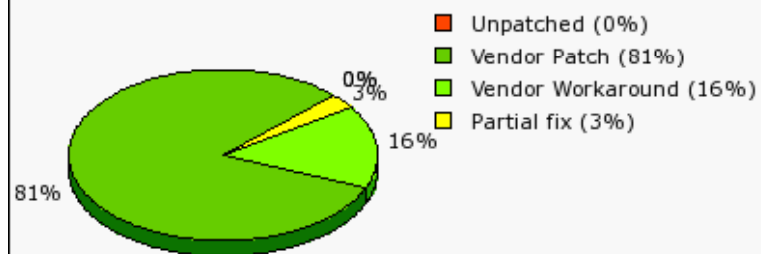
# Linux Vulnerabilities (cont'd)

**Linux Kernel 2.6.x  
 Advisories (Based on 32 advisories from 2007)**



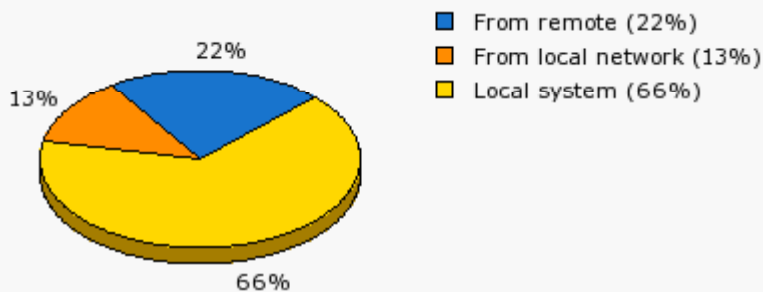
This graph was generated by Secunia.  
 Based on vulnerability information available at <http://secunia.com/>

**Linux Kernel 2.6.x  
 Solution Status (Based on 32 advisories from 2007)**



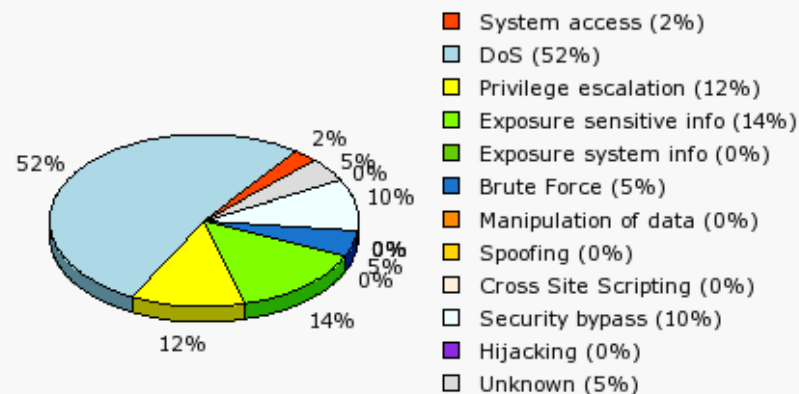
This graph was generated by Secunia.  
 Based on vulnerability information available at <http://secunia.com/>

**Linux Kernel 2.6.x  
 Where (Based on 32 advisories from 2007)**



This graph was generated by Secunia.  
 Based on vulnerability information available at <http://secunia.com/>

**Linux Kernel 2.6.x  
 Impact (Based on 32 advisories from 2007)**



This graph was generated by Secunia.  
 Based on vulnerability information available at <http://secunia.com/>

Linux is an open source Operating System with many vendors providing different security options

Unlike other OSs, Linux is not secure

Linux is optimized for convenience and does not make security easy or natural

The security on Linux will vary from user to user

Linux security is effectively binary: all or nothing in terms of power. Facilities such as setuid execution tend to give way in the middle





# Why is Linux Hacked

Linux is widely used on a large number of servers in the world, making it a 'de facto' backbone

Since application source code is available, it is easy to find out the vulnerabilities of the system

Many applications on Linux are installed by default so they are more vulnerable to attacks

There are many default installed setuid programs

There are many default installed daemons

- The admin must remove unused daemons
- Change `/etc/rc.d` files and `/etc/inetd.conf` file

# How to Apply Patches to Vulnerable Programs

Check the Linux distribution homepage e.g., Redhat, Debian, Alzza, and so on

Go to the respective websites of the vendors from whom the user has bought the program and downloaded the patches



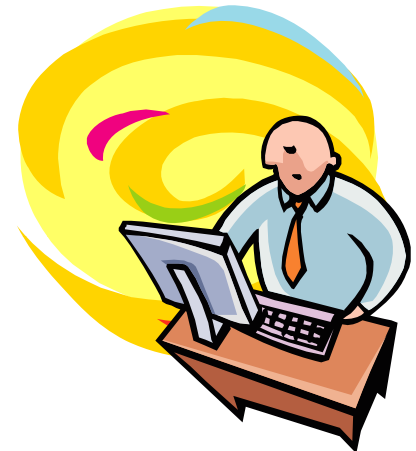
# Scanning Networks

Once the IP address of a target system is known, an attacker can begin the process of port scanning, looking for holes in the system through which the attacker can gain access

A typical system has  $2^{16} - 1$  port numbers with one TCP port and one UDP port for each number

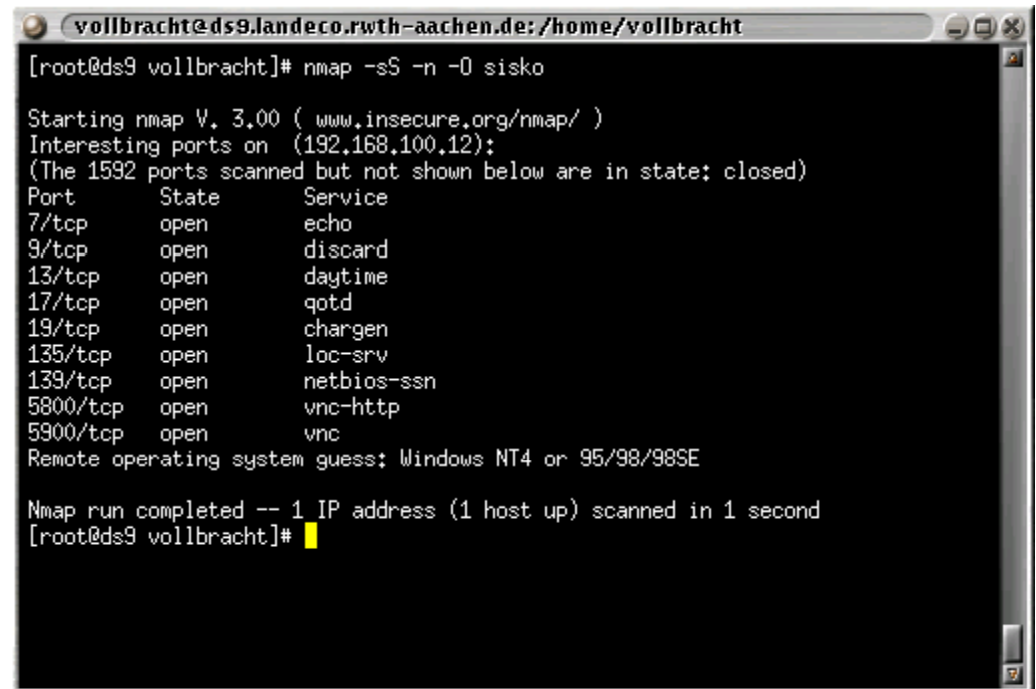
Each one of these ports is a potential way into the system

The most popular scanning tool for Linux is Nmap



**Nmap** is a tool used for determining the hosts that are running and what services the hosts are running

**Nmap** can be a valuable diagnostic tool for network administrators



```
vollbracht@ds9.landeco.rwth-aachen.de: /home/vollbracht
[root@ds9 vollbracht]# nmap -sS -n -O sisko

Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
Interesting ports on (192.168.100.12):
(The 1592 ports scanned but not shown below are in state: closed)
Port      State  Service
7/tcp    open   echo
9/tcp    open   discard
13/tcp   open   daytime
17/tcp   open   qotd
19/tcp   open   chargen
135/tcp  open   loc-srv
139/tcp  open   netbios-ssn
5800/tcp open   vnc-http
5900/tcp open   vnc
Remote operating system guess: Windows NT4 or 95/98/98SE

Nmap run completed -- 1 IP address (1 host up) scanned in 1 second
[root@ds9 vollbracht]#
```

# Nmap: Screenshot

```
# nmap -A -T4 scanme.nmap.org playground
```

```
Starting nmap ( http://insecure.org/nmap/ )  
Interesting ports on scanme.nmap.org (205.217.153.62):  
(The 1663 ports scanned but not shown below are in state: filtered)  
PORT      STATE SERVICE VERSION  
22/tcp    open  ssh      OpenSSH 3.9pl (protocol 1.99)  
53/tcp    open  domain  
70/tcp    closed gopher  
80/tcp    open  http     Apache httpd 2.0.52 ((Fedora))  
113/tcp   closed auth  
Device type: general purpose  
Running: Linux 2.4.X|2.5.X|2.6.X  
OS details: Linux 2.4.7 - 2.6.11, Linux 2.6.0 - 2.6.11  
Uptime 33.908 days (since Thu Jul 21 03:38:03 2005)
```

```
Interesting ports on playground.nmap.org (192.168.0.40):  
(The 1659 ports scanned but not shown below are in state: closed)  
PORT      STATE SERVICE      VERSION  
135/tcp   open  msrpc        Microsoft Windows RPC  
139/tcp   open  netbios-ssn  
389/tcp   open  ldap?  
445/tcp   open  microsoft-ds Microsoft Windows  
1002/tcp  open  windows-icfw?  
1025/tcp  open  msrpc        Microsoft Windows  
1720/tcp  open  H.323/Q.931  CompTek AquaGate  
5800/tcp  open  vnc-http     RealVNC 4.0 (RealVNC)  
5900/tcp  open  vnc          VNC (protocol 3.3)  
MAC Address: 00:A0:CC:63:85:4B (Lite-on Comm)  
Device type: general purpose  
Running: Microsoft Windows NT/2K/XP  
OS details: Microsoft Windows XP Pro RC1+ the  
Service Info: OSs: Windows, Windows XP  
  
Nmap finished: 2 IP addresses (2 hosts up) scanned in 0.213 seconds
```

```
notwist@notwist:~$ nmap localhost  
Starting Nmap 4.20 ( http://insecure.org ) at 2007-04-02 15:50 CEST  
Interesting ports on localhost (127.0.0.1):  
Not shown: 1691 closed ports  
PORT      STATE SERVICE  
22/tcp    open  ssh  
80/tcp    open  http  
139/tcp   open  netbios-ssn  
445/tcp   open  microsoft-ds  
631/tcp   open  ipp  
3306/tcp  open  mysql  
  
Nmap finished: 1 IP address (1 host up) scanned in 0.213 seconds  
notwist@notwist:~$
```

```
31337
# nmap -A -T4 scanme.nmap.org d0ze
Starting Nmap 4.01 ( http://www.insecure.org/nmap/ ) at 2006-03-20 15:53 PST
Interesting ports on scanme.nmap.org (205.217.153.62):
(The 1667 ports scanned but not shown below are in state: filtered)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 3.9p1 (protocol 1.99)
25/tcp    opn   smtp     Postfix smtpd
53/tcp    open  domain   ISC Bind 9.2.1
70/tcp    closed gopher
80/tcp    open  http     Apache httpd 2.0.52 ((Fedora))
113/tcp   closed auth
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.0 - 2.6.11
Uptime 26.177 days (since Wed Feb 22 11:39:16 2006)

Interesting ports on d0ze.internal (192.168.12.3):
(The 1664 ports scanned but not shown below are in state: closed)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      Serv-U ftpd 4.0
25/tcp    open  smtp     IMail NT-ESMTP 7.15 2015-2
80/tcp    open  http     Microsoft IIS webserver 5.0
110/tcp   open  pop3     IMail pop3d 7.15 931-1
135/tcp   open  mstask   Microsoft mstask (task server - c:\winnt\system32\
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds Microsoft Windows XP microsoft-ds
1025/tcp  open  msrpc    Microsoft Windows RPC
5800/tcp  open  vnc-http Ultr@VNC (Resolution 1024x800; VNC TCP port: 5900)
MAC Address: 00:A0:CC:51:72:7E (Lite-on Communications)
Device type: general purpose
Running: Microsoft Windows NT/2K/XP
OS details: Microsoft Windows 2000 Professional
Service Info: OS: Windows

Nmap finished: 2 IP addresses (2 hosts up) scanned in 42.291 seconds
flog/home/fyodor/nmap-misc/Screenshots/042006#
```

The screenshot shows the Nmap Front End v3.49 application window. The target is set to www.insecure.org. The scan type is SYN Stealth Scan. The scanned ports are Most Important [fast]. The scan extensions include OS Detection and Version Probe. The output shows the following results:

```
Starting nmap 3.49 ( http://www.insecure.org/nmap/ ) at 2003-12-19 14:28 PST
Interesting ports on www.insecure.org (205.217.153.53):
(The 1212 ports scanned but not shown below are in state: filtered)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 3.1p1 (protocol 1.99)
25/tcp    open  snmp     qmail snmpd
53/tcp    open  domain   ISC Bind 9.2.1
80/tcp    open  http     Apache httpd 2.0.39 ((Unix) mod_perl/1.99_07-dev Perl/v5.6.1)
113/tcp   closed auth
Device type: general purpose
Running: Linux 2.4.X12.5.X
OS details: Linux Kernel 2.4.0 - 2.5.20
Uptime 212.119 days (since Wed May 21 12:38:26 2003)

Nmap run completed -- 1 IP address (1 host up) scanned in 33.792 seconds
```

Command: `nmap -sS -sV -O -F -PI -T4 www.insecure.org`

# Scanning Tool: Nessus

One essential type of tool for any attacker, or defender, is the vulnerability scanner

These tools allow attacker to connect to a target system and check for such vulnerabilities as configuration errors, default configuration settings that allow attackers access, and the most recently reported system vulnerabilities

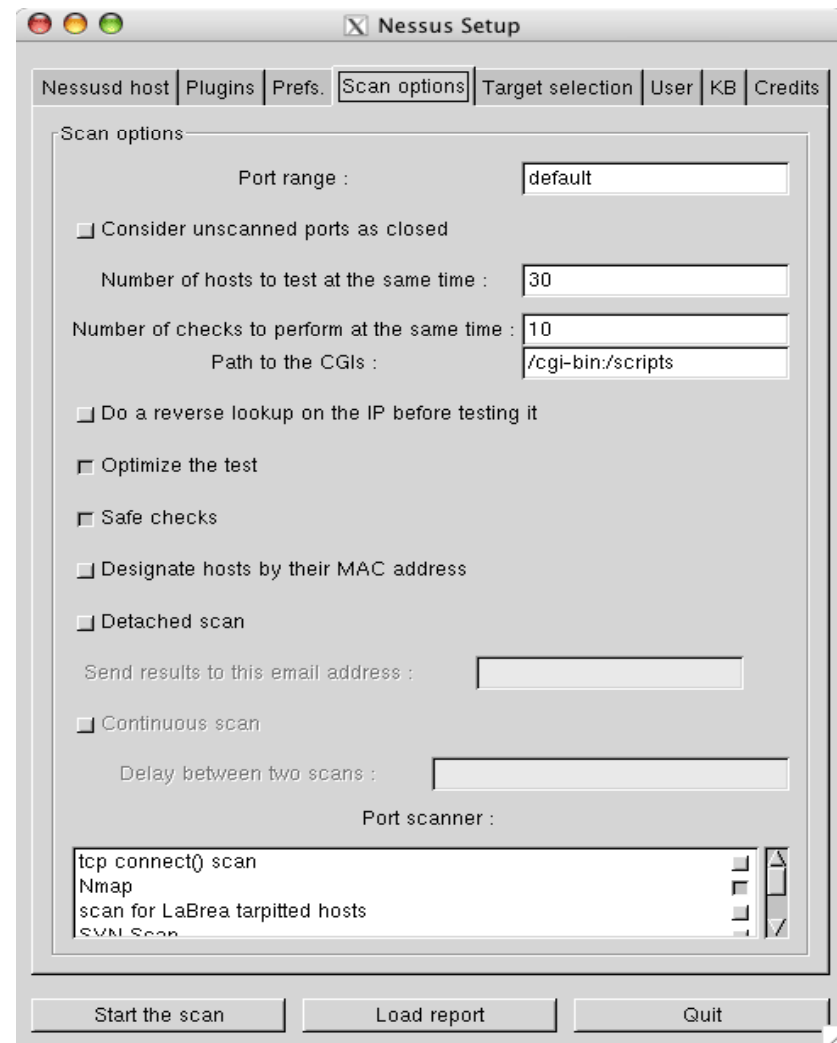
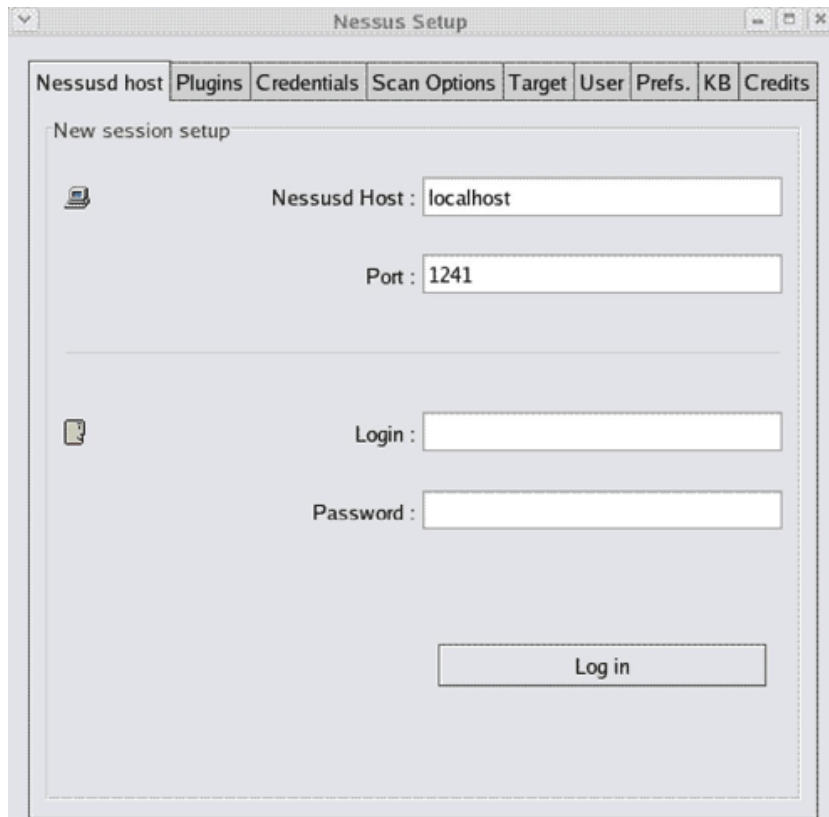
The preferred open-source tool for this is Nessus

Nessus is a powerful network scanner. It can also be configured to run a variety of attacks

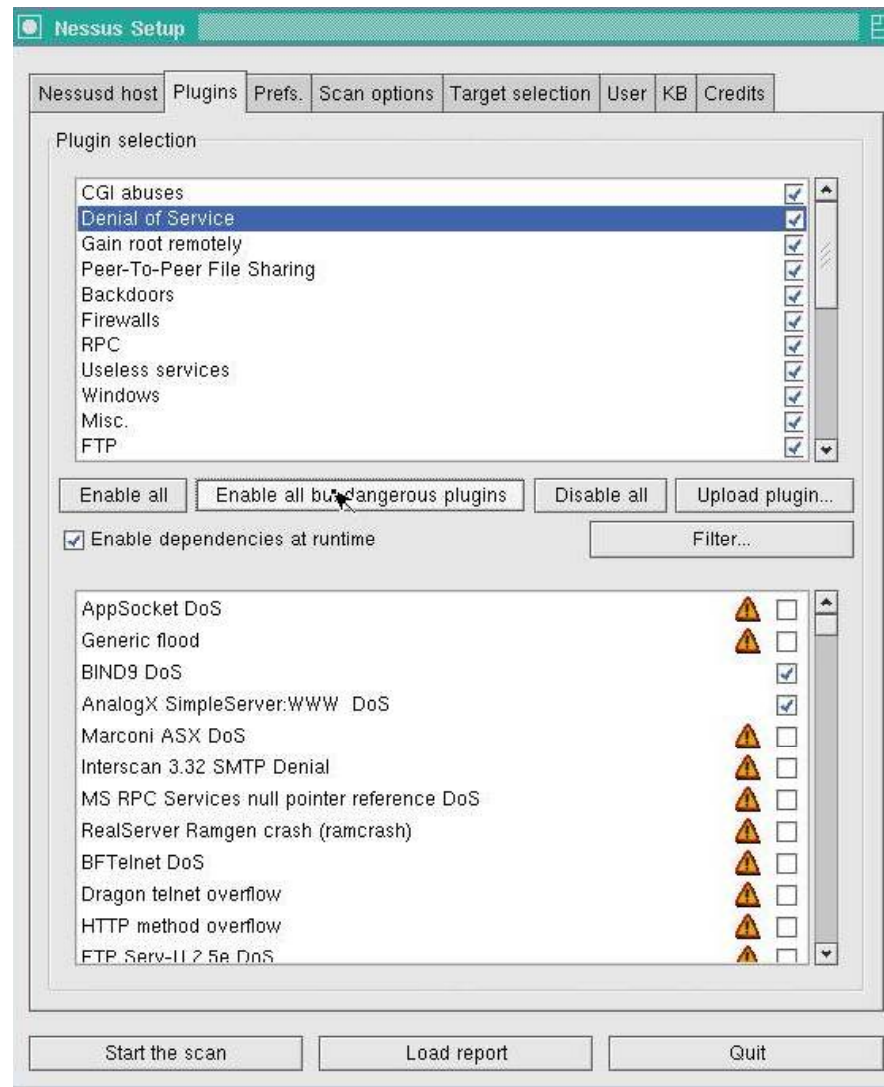




# Nessus: Screenshot 1



# Nessus: Screenshot 2



- ◉ **KLAXON**
- ◉ **Scanlogd**
  - <http://www.openwall.com/scanlogd/>
  - Detects and logs TCP port scans
- ◉ **Scanlogd**
  - only logs port scans
  - It does not prevent them
  - The user will only receive summarized information in the system's log
- ◉ **Psionic PortSentry**
  - <http://www.psionic.com/products/port Sentry/>
  - Portscan detection daemon, Portsentry, has the ability to detect port scans (including stealth scans) on the network interfaces of the user's server
  - Upon alarm, it can block the attacker via hosts.deny, dropped route, or firewall rule

# Password Cracking in Linux: Xcrack

Xcrack finds any passwords that match words in the dictionary file the user provides, but it would not apply any combinations or modifications of those words

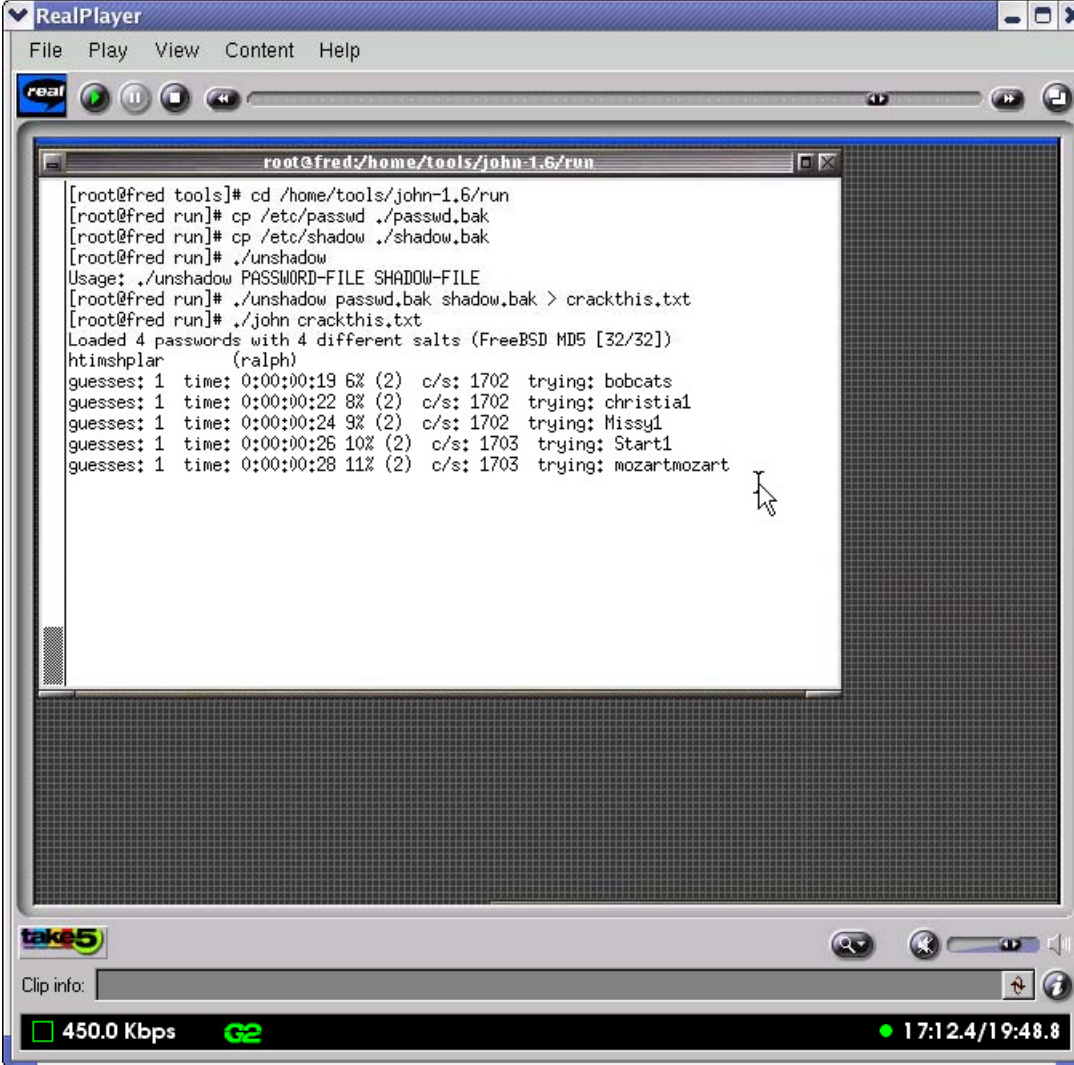
It is a comparatively fast tool

John the Ripper is another popular password cracking tool



Source: <http://packetstorm.linuxsecurity.com>

# Password Cracking in Linux: Screenshot



```
root@fred:/home/tools/john-1.6/run
[root@fred tools]# cd /home/tools/john-1.6/run
[root@fred run]# cp /etc/passwd ./passwd.bak
[root@fred run]# cp /etc/shadow ./shadow.bak
[root@fred run]# ./unshadow
Usage: ./unshadow PASSWORD-FILE SHADOW-FILE
[root@fred run]# ./unshadow passwd.bak shadow.bak > crackthis.txt
[root@fred run]# ./john crackthis.txt
Loaded 4 passwords with 4 different salts (FreeBSD MD5 [32/32])
htimshplar (ralph)
guesses: 1 time: 0:00:00:19 6% (2) c/s: 1702 trying: bobcats
guesses: 1 time: 0:00:00:22 8% (2) c/s: 1702 trying: christial
guesses: 1 time: 0:00:00:24 9% (2) c/s: 1702 trying: Missy1
guesses: 1 time: 0:00:00:26 10% (2) c/s: 1703 trying: Start1
guesses: 1 time: 0:00:00:28 11% (2) c/s: 1703 trying: mozartmozart
```

# Firewall in Linux: IPTables

IPTables is the replacement of userspace tool ipchains in the Linux kernel and beyond. It has many more features than IPChains

Connection tracking capability is the ability to do stateful packet inspection

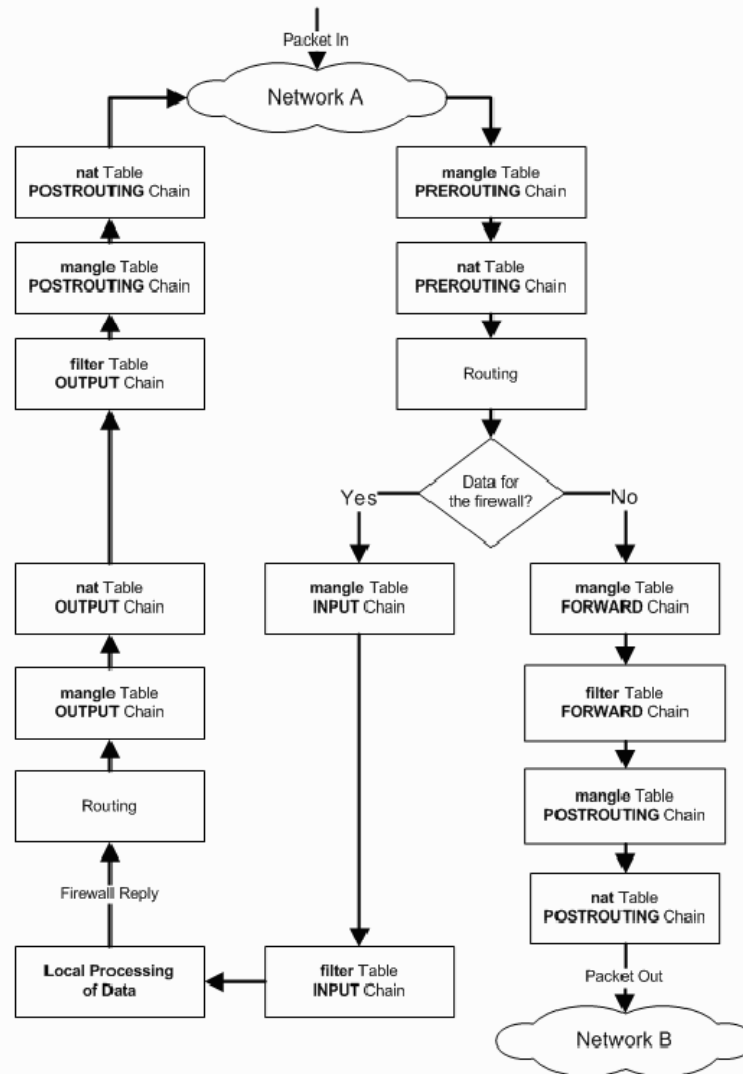
Simplified behavior of packets negotiating the built-in chains (INPUT, OUTPUT, and FORWARD)

A clean separation of packet filtering and network address translation (NAT)

Rate-limited connection and logging capability

The ability to filter on tcp flag and tcp options and also MAC addresses

# Firewall in Linux: IPTables (cont'd)



# IPTables Command

```
⊙ iptables -A INPUT -s 0/0 -i eth0  
-d 192.168.1.1 -p TCP -j ACCEPT
```

- iptables is being configured to allow the firewall to accept TCP packets coming in on interface eth0 from any IP address destined for the firewall's IP address of 192.168.1.1

```
⊙ iptables -A OUTPUT -p icmp --  
icmp-type echo-request -j ACCEPT  
iptables -A INPUT -p icmp --  
icmp-type echo-reply -j ACCEPT
```

- iptables is being configured to allow the firewall to send ICMP echo-requests (pings) and in turn, accept the expected ICMP echo-replies





# Basic Linux Operating System Defense

Linux operating system has a number of built-in protection mechanisms that you should activate by modifying the system kernel parameters in the `/proc` filesystem via the `/etc/sysctl.conf` file

Change the `/etc/sysctl.conf` to modify kernel parameters



# Basic Linux Operating System Defense (cont'd)

```
# File: /etc/sysctl.conf
#-----
# Disable routing triangulation. Respond to queries out
# the same interface, not another. Helps to maintain state
# Also protects against IP spoofing
#-----
net/ipv4/conf/all/rp_filter = 1
#-----
# Enable logging of packets with malformed IP addresses
#-----
net/ipv4/conf/all/log_martians = 1
#-----
# Disable redirects
#-----
net/ipv4/conf/all/send_redirects = 0
#-----
# Disable source routed packets
#-----
net/ipv4/conf/all/accept_source_route = 0
```

# Basic Linux Operating System Defense (cont'd)

```
#-----  
# Disable acceptance of ICMP redirects  
#-----  
net/ipv4/conf/all/accept_redirects = 0  
#-----  
# Turn on protection from Denial of Service (DOS) attacks  
#-----  
net/ipv4/tcp_syncookies = 1  
#-----  
# Disable responding to ping broadcasts  
#-----  
net/ipv4/icmp_echo_ignore_broadcasts = 1  
#-----  
# Enable IP routing. Required if your firewall is protecting a  
# network, NAT included  
#-----  
net/ipv4/ip_forward = 1
```

# SARA (Security Auditor's Research Assistant)

The Security Auditor's Research Assistant (SARA) is a third generation Unix-based security analysis tool that supports the FBI Top 20 Consensus on Security

SARA operates on most Unix-type platforms including Linux and Mac OS X

SARA is the upgrade of SATAN tool

Getting SARA up and running is a straightforward compilation process and the rest is done via a browser



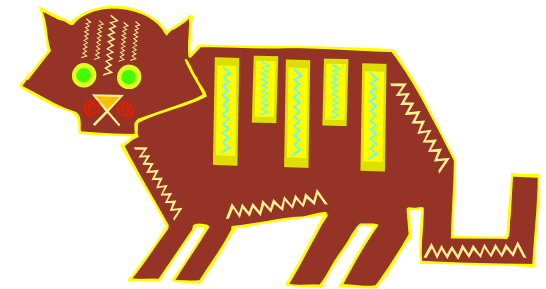
Source: <http://www-arc.com>

# Linux Tool: Netcat

TCP/IP swiss army knife is a simple Unix utility that reads and writes data across network connections using TCP or UDP protocol

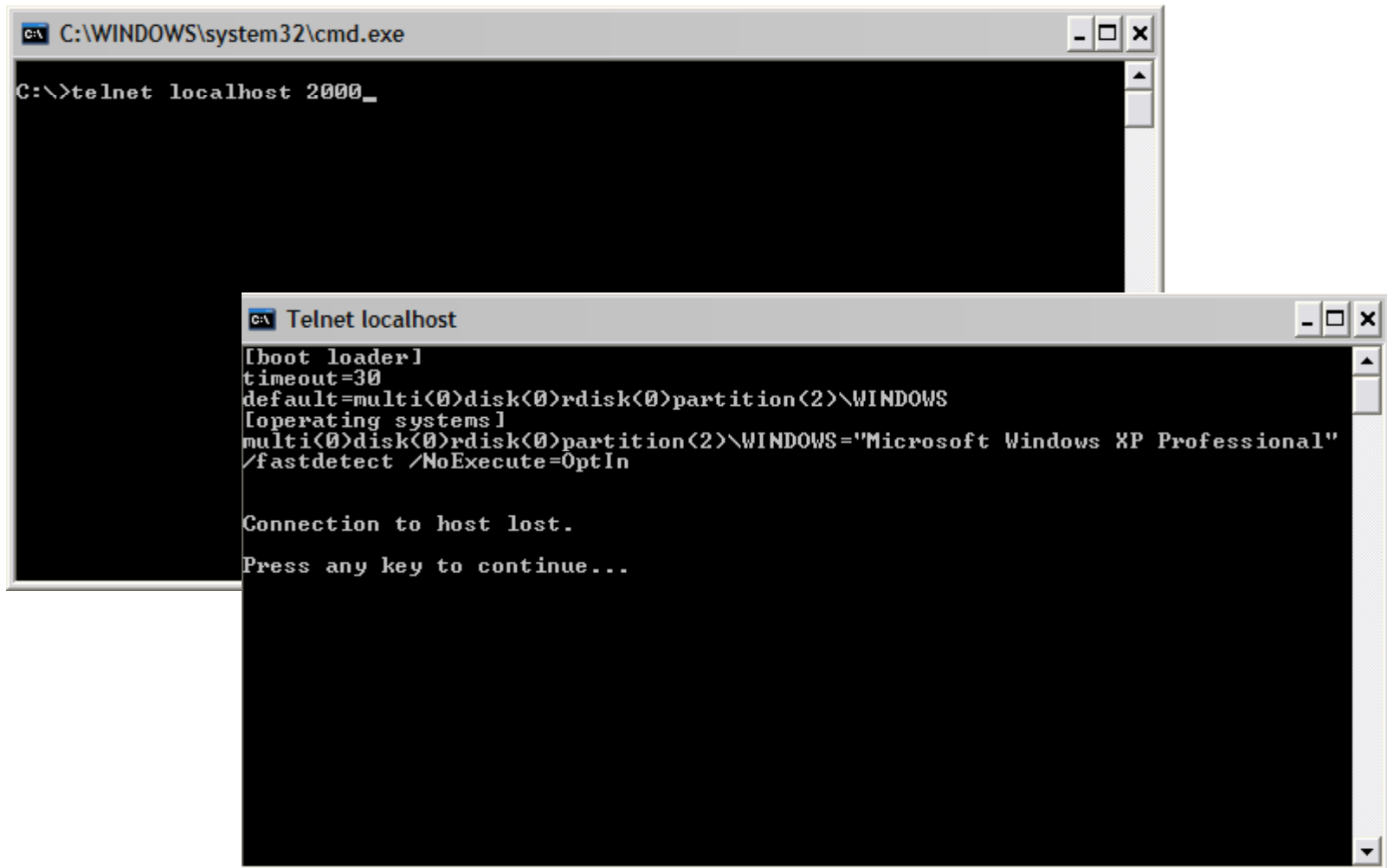
It is designed to be a reliable "back-end" tool that can be used directly or easily driven by other programs and scripts

It can create almost any kind of connection you would need and has several interesting built-in capabilities



Source: <http://www.atstake.com>

# Netcat: Screenshot 1



# Netcat: Screenshot 2

```
C:\>netcat /?
netcat 1.0.0.0 - Copyright c Mike's Consulting 2006

Usage: netcat [options] [host]
Options:
-? -help           Show this help list
   -help2         Show an additional help list
   -if:PARAM      Input file
-i -input          Standard Input
-l -listen         Listen
   -of:PARAM      Output file
-o -output         Standard Output
-p -port:PARAM    Port (required)
   -ssl           SSL
   -usage         Show usage
-v -verbose        Verbose
-U -version        Display version

C:\>
```

```
C:\>netcat -if boot.ini -l -p 2000
C:\>_
```

# Linux Tool: tcpdump

A powerful tool for network monitoring and data acquisition which allows you to dump the traffic on a network

It can be used to print out the headers of packets on a network interface that match a given expression

You can use this tool to track down network problems, to detect "ping attacks," or to monitor the network activities



Source: <http://www.tcpdump.org>



# tcpdump: Screenshot 1

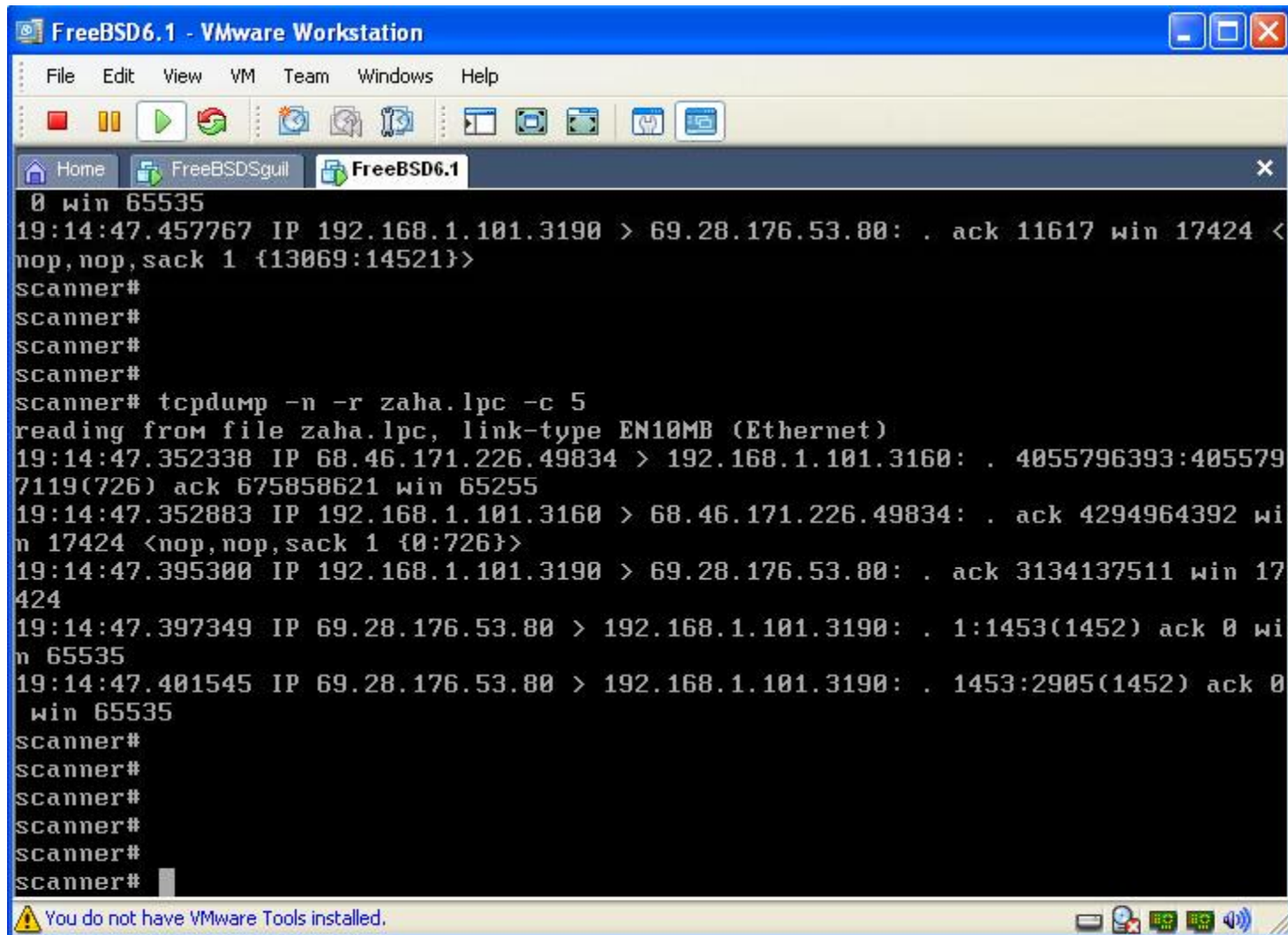
```
4dos
16:27:55.327570 137.133.57.68.1040 > 137.133.24.8.1352: tcp 0 <DF>
16:27:55.330774 209.1.224.18.www-http > 137.133.57.68.1255: tcp 592
16:27:55.333843 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:55.336912 209.1.224.18.www-http > 137.133.57.68.1255: tcp 1160
16:27:55.340174 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:55.343677 209.1.224.18.www-http > 137.133.57.68.1255: tcp 568
16:27:55.437553 209.1.224.18.www-http > 137.133.57.68.1255: tcp 48
16:27:55.440488 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:55.444033 209.1.224.18.www-http > 137.133.57.68.1255: tcp 1160
16:27:55.447007 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:55.450144 209.1.224.18.www-http > 137.133.57.68.1255: tcp 1160
16:27:55.456636 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:55.657583 209.1.224.18.www-http > 137.133.57.68.1255: tcp 1160
16:27:55.660751 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:55.877792 137.133.16.54.32793 > 137.133.63.36.1730: tcp 147 <DF>
16:27:55.881254 209.1.224.18.www-http > 137.133.57.68.1255: tcp 592
16:27:55.884685 209.1.224.18.www-http > 137.133.57.68.1255: tcp 616
16:27:55.887807 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:55.890866 137.133.63.36.1730 > 137.133.16.54.32793: tcp 0 <DF>
16:27:55.893980 209.1.224.18.www-http > 137.133.57.68.1255: tcp 1160
16:27:55.987888 209.1.224.18.www-http > 137.133.57.68.1255: tcp 1160
16:27:55.991367 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:55.994454 209.1.224.18.www-http > 137.133.57.68.1255: tcp 1112
16:27:55.997915 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:56.000966 137.133.63.36.33272 > 137.133.16.54.1730: tcp 147 <DF>
16:27:56.004058 137.133.63.36.32789 > 137.133.16.53.1730: tcp 147 <DF>
16:27:56.207953 nera-x.1035 > nera-y.loc-serv: udp 204
16:27:56.210769 nera-y.loc-serv > nera-x.1035: udp 172
16:27:56.213192 209.1.224.18.www-http > 137.133.57.68.1255: tcp 664
16:27:56.216609 209.1.224.18.www-http > 137.133.57.68.1255: tcp 834
16:27:56.219818 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:56.222877 209.1.224.18.www-http > 137.133.57.68.1255: tcp 326
16:27:56.318051 209.1.224.18.www-http > 137.133.57.68.1255: tcp 1160
16:27:56.321065 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:56.324244 209.1.224.18.www-http > 137.133.57.68.1255: tcp 1106
16:27:56.327222 209.1.224.18.www-http > 137.133.57.68.1255: tcp 670
16:27:56.330206 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:56.333267 209.1.224.18.www-http > 137.133.57.68.1255: tcp 700
16:27:56.428080 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:56.538072 209.1.224.18.www-http > 137.133.57.68.1255: tcp 460
16:27:56.541090 209.1.224.18.www-http > 137.133.57.68.1255: tcp 1160
16:27:56.544893 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:56.648239 209.1.224.18.www-http > 137.133.57.68.1255: tcp 1160
16:27:56.651262 209.1.224.18.www-http > 137.133.57.68.1255: tcp 616
16:27:56.654247 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>
16:27:56.657292 209.1.224.18.www-http > 137.133.57.68.1255: tcp 1160
16:27:56.660425 209.1.224.18.www-http > 137.133.57.68.1255: tcp 911
16:27:56.663376 137.133.57.68.1255 > 209.1.224.18.www-http: tcp 0 <DF>

tcpdump 3.5 Eth: 139 <139>
```

# tcpdump: Screenshot 2

```
root@localhost:~  
[root@localhost ~]# tcpdump -X -i eth0 port 80  
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode  
listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes  
01:09:10.196603 IP 192.168.11.2.gxte!md > 192.168.11.4.http: S 3053066467:3053066467(0) win 16384 <mss 1460,  
  0x0000: 4500 0030 601c 4000 8006 0355 c0a8 0b02 E..0`.0....U....  
  0x0010: c0a8 0b04 0934 0050 b5fa 18e3 0000 0000 .....4.P.....  
  0x0020: 7002 4000 d366 0000 0204 05b4 0101 0402 p.0..f.....  
01:09:10.239801 IP 192.168.11.4.http > 192.168.11.2.gxte!md: S 2904952068:2904952068(0) ack 3053066468 win 5  
p,nop,sackOK>  
  0x0000: 4500 0030 0000 4000 4006 a371 c0a8 0b04 E..0..0.0..q....  
  0x0010: c0a8 0b02 0050 0934 ad26 0d04 b5fa 18e4 .....P.4.&.....  
  0x0020: 7012 16d0 425b 0000 0204 05b4 0101 0402 p...B[.....  
01:09:10.197422 IP 192.168.11.2.gxte!md > 192.168.11.4.http: . ack 1 win 17520  
  0x0000: 4500 0028 601e 4000 8006 035b c0a8 0b02 E..(`.0....[....  
  0x0010: c0a8 0b04 0934 0050 b5fa 18e4 ad26 0d05 .....4.P.....&..  
  0x0020: 5010 4470 417f 0000 0000 0000 0000 0000 P.DpA.....  
01:09:10.197873 IP 192.168.11.2.gxte!md > 192.168.11.4.http: P 1:339(338) ack 1 win 17520  
  0x0000: 4500 017a 601f 4000 8006 0208 c0a8 0b02 E..z`.0.....  
  0x0010: c0a8 0b04 0934 0050 b5fa 18e4 ad26 0d05 .....4.P.....&..  
  0x0020: 5018 4470 c9d4 0000 4745 5420 2f70 6870 P.Dp....GET./php  
  0x0030: 696e 666f 2e70 6870 2048 5454 502f 312e info.php.HTTP/1.  
  0x0040: 310d 0a41 6363 6570 743a 2069 6d61 6765 1..Accept:.image  
  0x0050: 2f67 /g  
01:09:10.197978 IP 192.168.11.4.http > 192.168.11.2.gxte!md: . ack 339 win 6432  
  0x0000: 4500 0028 0aa4 4000 4006 98d5 c0a8 0b04 E..(..0.0.....  
  0x0010: c0a8 0b02 0050 0934 ad26 0d05 b5fa 1a36 .....P.4.&.....6  
  0x0020: 5010 1920 6b7d 0000 P...k}..  
01:09:10.203618 IP 192.168.11.4.http > 192.168.11.2.gxte!md: . 1:1461(1460) ack 339 win 6432  
  0x0000: 4500 05dc 0aa5 4000 4006 9320 c0a8 0b04 E.....0.0.....  
  0x0010: c0a8 0b02 0050 0934 ad26 0d05 b5fa 1a36 .....P.4.&.....6  
  0x0020: 5010 1920 3a91 0000 4854 5450 2f31 2e31 P.....HTTP/1.1  
  0x0030: 2032 3030 204f 4b0d 0a44 6174 653a 2053 .200.OK..Date:.S  
  0x0040: 6174 2c20 3038 2044 6563 2032 3030 3720 at,.08.Dec.2007.  
  0x0050: 3136 16  
01:09:10.203668 IP 192.168.11.4.http > 192.168.11.2.gxte!md: . 1461:2921(1460) ack 339 win 6432  
  0x0000: 4500 05dc 0aa6 4000 4006 931f c0a8 0b04 E.....0.0.....  
  0x0010: c0a8 0b02 0050 0934 ad26 12b9 b5fa 1a36 .....P.4.&.....6  
  0x0020: 5010 1920 6e04 0000 7769 6474 683d 2236 P...n...width="6  
  0x0030: 3030 223e 0a3c 7472 2063 6c61 7373 3d22 00">.<tr.class="
```

# tcpdump: Screenshot 3



```
FreeBSD6.1 - VMware Workstation
File Edit View VM Team Windows Help
Home FreeBSGSguil FreeBSD6.1
0 win 65535
19:14:47.457767 IP 192.168.1.101.3190 > 69.28.176.53.80: . ack 11617 win 17424 <
nop,nop,sack 1 {13069:14521}>
scanner#
scanner#
scanner#
scanner#
scanner# tcpdump -n -r zaha.lpc -c 5
reading from file zaha.lpc, link-type EN10MB (Ethernet)
19:14:47.352338 IP 68.46.171.226.49834 > 192.168.1.101.3160: . 4055796393:405579
7119(726) ack 675858621 win 65255
19:14:47.352883 IP 192.168.1.101.3160 > 68.46.171.226.49834: . ack 4294964392 wi
n 17424 <nop,nop,sack 1 {0:726}>
19:14:47.395300 IP 192.168.1.101.3190 > 69.28.176.53.80: . ack 3134137511 win 17
424
19:14:47.397349 IP 69.28.176.53.80 > 192.168.1.101.3190: . 1:1453(1452) ack 0 wi
n 65535
19:14:47.401545 IP 69.28.176.53.80 > 192.168.1.101.3190: . 1453:2905(1452) ack 0
win 65535
scanner#
scanner#
scanner#
scanner#
scanner#
scanner#
```

You do not have VMware Tools installed.

# Linux Tool: Snort

Flexible packet sniffer/logger that detects attacks, Snort is a libpcap-based packet sniffer/logger that can be used as a lightweight network intrusion detection system

Snort has a real-time alerting capability, with alerts being sent to syslog, a separate "alert" file, or even to a Windows computer via Samba

```
F:\Snort>snort

---= Initializing Snort =---

-*> Snort! <*-
Version 1.7-WIN32
By Martin Roesch <roesch@clark.net, www.snort.org>
WIN32 Port By Michael Davis <mike@datanerds.net, www.datanerds.net/~nike>
USAGE: snort [-options] <filter options>
Options:
  -A          Set alert mode: fast, full, or none <alert file alerts only>
              "unsock" enables UNIX socket logging <experimental>. *
  -a          Display ARP packets
  -b          Log packets in tcpdump format <much faster!>
  -c <rules>  Use Rules File <rules>
  -C          Print out payloads with character data only <no hex>
  -d          Dump the Application Layer
  -D          Run Snort in background <daemon> mode
  -e          Display the second layer header info
  -E          Log alert messages to NT Eventlog.
  -F <bpf>    Read BPF filters from file <bpf>
  -g <gname>  Run snort gid as 'gname' user or uid after initialization *
  -h <hn>     Home network = <hn>
  -i <if>     Listen on interface <if>
  -I          Add Interface name to alert output
  -l <ld>     Log to directory <ld>
  -n <cnt>    Exit after receiving <cnt> packets
  -N          Turn off logging <alerts still work>
  -o          Change the rule testing order to Pass!Alert!Log
  -O          Obfuscate the logged IP addresses
  -p          Disable promiscuous mode sniffing
  -P <snap>   set explicit snaplen of packet <default: 1514>
  -q          Quiet. Don't show banner and status report
  -r <tf>     Read and process tcpdump file <tf>
  -s <server:port> Log alert messages to syslog server <default port: 514>
  -S <n=v>    Set rules file variable n equal to value v
  -t <dir>    Chroots process to <dir> after initialization
  -u <uname>  Run snort uid as <uname> user <or uid> after initialization
              Use UTC for timestamps
  -v          Be verbose
  -W          Lists available interfaces.
  -V          Show version number
  -X          Dump the raw packet data starting at the link layer
  -?          Show this information
<Filter Options> are standard BPF options, as seen in TCPDump

* denotes an option that is NOT SUPPORTED in this WIN32 port of snort.

Uh, you need to tell me to do something....

: Invalid argument

F:\Snort>
```



# Snort: Screenshot

Snort IDS Console - Microsoft Internet Explorer

Address: https://...

Snort IDS Console [Unfilter](#) Refresh every 30 secs. View alerts since 6 AM or on

Alert Information			Sensors			Top Sources			Top Targets			Top Target Ports			
	#	%	Sensor	Sigs	Alerts	IP Address	Sigs	Alerts	IP Address	Sigs	Alerts	TCP	#	UDP	#
Signatures:	62			19	482		6	186		6	186	80	513	1434	1,259
TCP Alerts <a href="#">View</a> :	1,126	42%		13	177		5	5		5	5	139	186	53	242
UDP Alerts <a href="#">View</a> :	1,523	57%		11	240		3	21		3	24	443	122	177	9
ICMP Alerts <a href="#">View</a> :	0	0%		11	131		2	108		2	352	1433	23	111	6
Total Alerts <a href="#">View</a> :	2,649	100%		9	298		2	92		2	92	3389	19	69	2

### Alert Overview by Signature

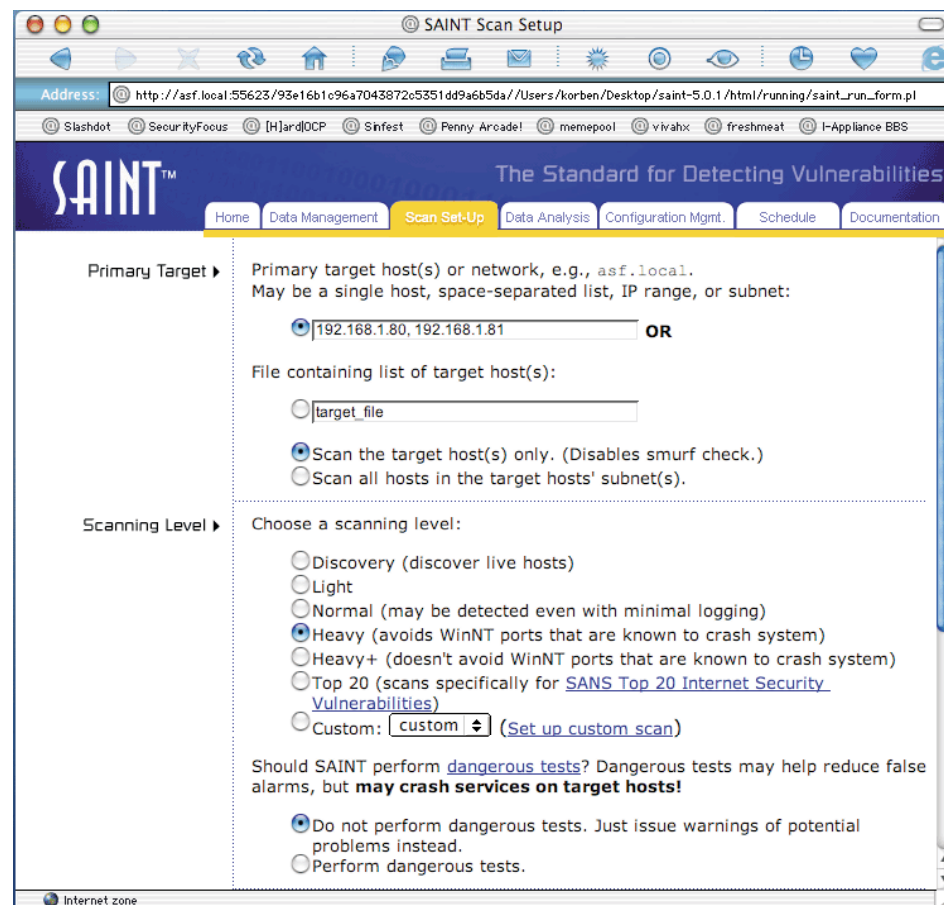
Earliest Alert: 2004-12-29 06:01:03  
Latest Alert: 2004-12-29 13:37:12

Signatures					
Prio	Signature	# Sensors	# Alerts	# Srcs	# Dests
1	<a href="#">WEB-MISC cross site scripting attempt [sid 1497]</a>	2	353	2	2
1	<a href="#">P2P Fastrack kazaa/morpheus traffic [sid 1699]</a>	2	145	3	49
1	<a href="#">MS-SQL/SMB raiserror possible buffer overflow [sid 1386]</a>	2	117	1	1
1	<a href="#">WEB-MISC NetObserve authentication bypass attempt [sid 2441]</a>	1	110	1	1
1	<a href="#">MS-SQL/SMB xp_cmdshell program execution [sid 681]</a>	2	33	1	1
1	<a href="#">WEB-MISC PCT Client Hello overflow attempt [sid 2515]</a>	2	25	1	8
1	<a href="#">MS-SQL xp_cmdshell - program execution [sid 687]</a>	1	17	2	1
1	<a href="#">MS-SQL/SMB xp_reg* registry access [sid 689]</a>	2	12	1	1
1	<a href="#">MS-SQL/SMB sp_password password change [sid 677]</a>	2	10	1	1
1	<a href="#">MS-SQL/SMB sp_delete alert log file deletion [sid 678]</a>	2	10	1	1
1	<a href="#">MS-SQL sp_start_job - program execution [sid 673]</a>	2	6	1	1
1	<a href="#">MS-SQL sa login failed [sid 688]</a>	1	5	1	1

# Linux Tool: SAINT

SAINT (Security Administrator's Integrated Network Tool) is a security assessment tool based on SATAN

Features include scanning through a firewall, updated security checks from CERT & CIAC bulletins, 4 levels of severity (red, yellow, brown, & green) and a feature rich HTML interface



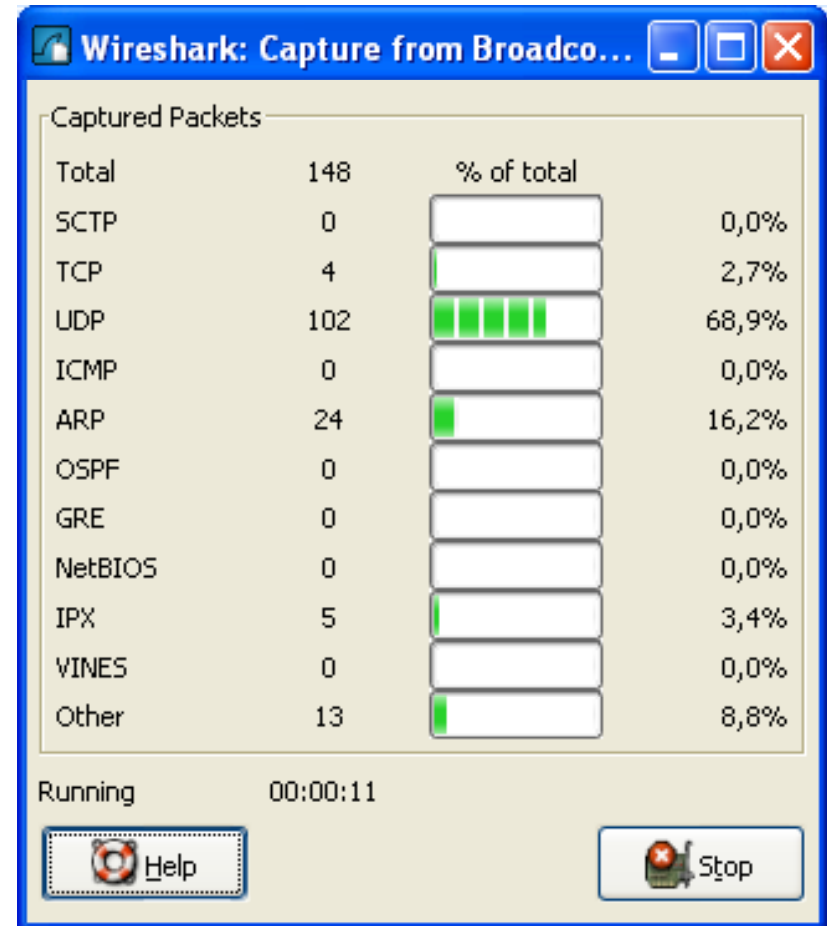
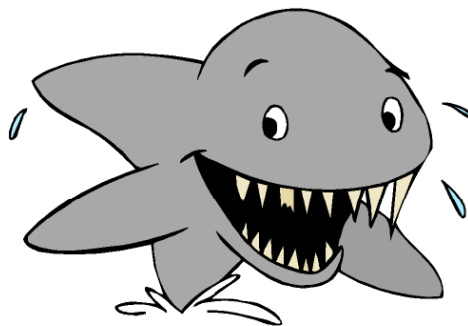
Source: <http://www.saintcorporation.com>

Copyright © by **EC-Council**  
All Rights Reserved. Reproduction is Strictly Prohibited

# Linux Tool: Wireshark

Network traffic analyzer wireshark is a network traffic analyzer, or "sniffer," for Unix and Unix-like operating systems

It uses GTK+, a graphical user interface library, and libpcap, a packet capture and filtering library



Source: <http://www.wireshark.org/>

(Untitled) - Wireshark

File Edit View Go Capture Analyze Statistics Help

Filter:  + Expression... Clear Apply

No. -	Time	Source	Destination	Protocol	Info
504	152.158290	192.168.12.21	66.187.224.210	DNS	Standard query A www.redhat.com
505	152.24944	66.187.224.210	192.168.12.21	DNS	Standard query response A 209.132.177.50
506	152.25091	192.168.12.21	209.132.177.50	TCP	48890 > http [SYN] Seq=0 Len=0 MSS=1460 TSV=1535
507	152.31125	209.132.177.50	192.168.12.21	TCP	http > 48890 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0
508	152.31132	192.168.12.21	209.132.177.50	TCP	48890 > http [ACK] Seq=1 Ack=1 Win=5840 Len=0 TS
509	152.31154	192.168.12.21	209.132.177.50	HTTP	GET / HTTP/1.1
510	152.38737	209.132.177.50	192.168.12.21	TCP	http > 48890 [ACK] Seq=1 Ack=498 Win=6864 Len=0
511	152.40516	209.132.177.50	192.168.12.21	TCP	[TCP segment of a reassembled PDU]
512	152.40520	192.168.12.21	209.132.177.50	TCP	48890 > http [ACK] Seq=498 Ack=1369 Win=8576 Len=0
513	152.41351	209.132.177.50	192.168.12.21	TCP	[TCP segment of a reassembled PDU]
514	152.41356	192.168.12.21	209.132.177.50	TCP	48890 > http [ACK] Seq=498 Ack=2737 Win=11312 Len=0
515	152.45058	192.168.12.21	209.132.177.50	TCP	48891 > http [SYN] Seq=0 Len=0 MSS=1460 TSV=1535
516	152.47685	209.132.177.50	192.168.12.21	TCP	[TCP segment of a reassembled PDU]
517	152.47690	192.168.12.21	209.132.177.50	TCP	48890 > http [ACK] Seq=498 Ack=4105 Win=14048 Len=0

▶ Frame 507 (74 bytes on wire, 74 bytes captured)  
 ▶ Ethernet II, Src: Amit\_04:ae:54 (00:50:18:04:ae:54), Dst: Intel\_e3:01:f5 (00:0c:f1:e3:01:f5)  
 ▶ Internet Protocol, Src: 209.132.177.50 (209.132.177.50), Dst: 192.168.12.21 (192.168.12.21)  
 ▾ Transmission Control Protocol, Src Port: http (80), Dst Port: 48890 (48890), Seq: 0, Ack: 1, Len: 0

Source port: http (80)  
 Destination port: 48890 (48890)  
 Sequence number: 0 (relative sequence number)  
 Acknowledgement number: 1 (relative ack number)  
 Header length: 40 bytes  
 ▶ Flags: 0x12 (SYN, ACK)  
 Window size: 5792  
 Checksum: 0x99db [correct]  
 ▶ Options: (20 bytes)  
 ▶ [SEQ/ACK analysis]

```

0000 00 0c f1 e3 01 f5 00 50 18 04 ae 54 08 00 45 00 .....P...T..E.
0010 00 3c 00 00 40 00 35 06 f6 47 d1 84 b1 32 c0 a8 <...@.5. .G...2..
0020 0c 15 00 50 be fa b5 36 ce 18 e0 bb b5 58 a0 12 ..P...6.....X..
0030 16 a0 99 db 00 00 02 04 05 64 04 02 08 0a 10 1d .....d.....
0040 ee de 5b 81 15 29 01 03 03 02 ..[...].
  
```

Source Port (tcp.srcport), 2 P: 1096 D: 1096 M: 0 Drops: 0



### Wireshark: Capture Options

**Capture**

Interface: Broadcom NetXtreme Gigabit Ethernet Driver: {Device}\NPF\_{4C4DB8EB-AC95-4B46-9}

IP address: 157.163.15.28

Link-layer header type: Ethernet Buffer size: 1 megabyte(s)

Capture packets in promiscuous mode

Limit each packet to 68 bytes

Capture Filter:

**Capture File(s)**

File: Browse...

Use multiple files

Next file every 1 megabyte(s)

Next file every 1 minute(s)

Ring buffer with 2 files

Stop capture after 1 file(s)

**Stop Capture ...**

... after 1 packet(s)

... after 1 megabyte(s)

... after 1 minute(s)

**Display Options**

Update list of packets in real time

Automatic scrolling in live capture

Hide capture info dialog

**Name Resolution**

Enable MAC name resolution

Enable network name resolution

Enable transport name resolution

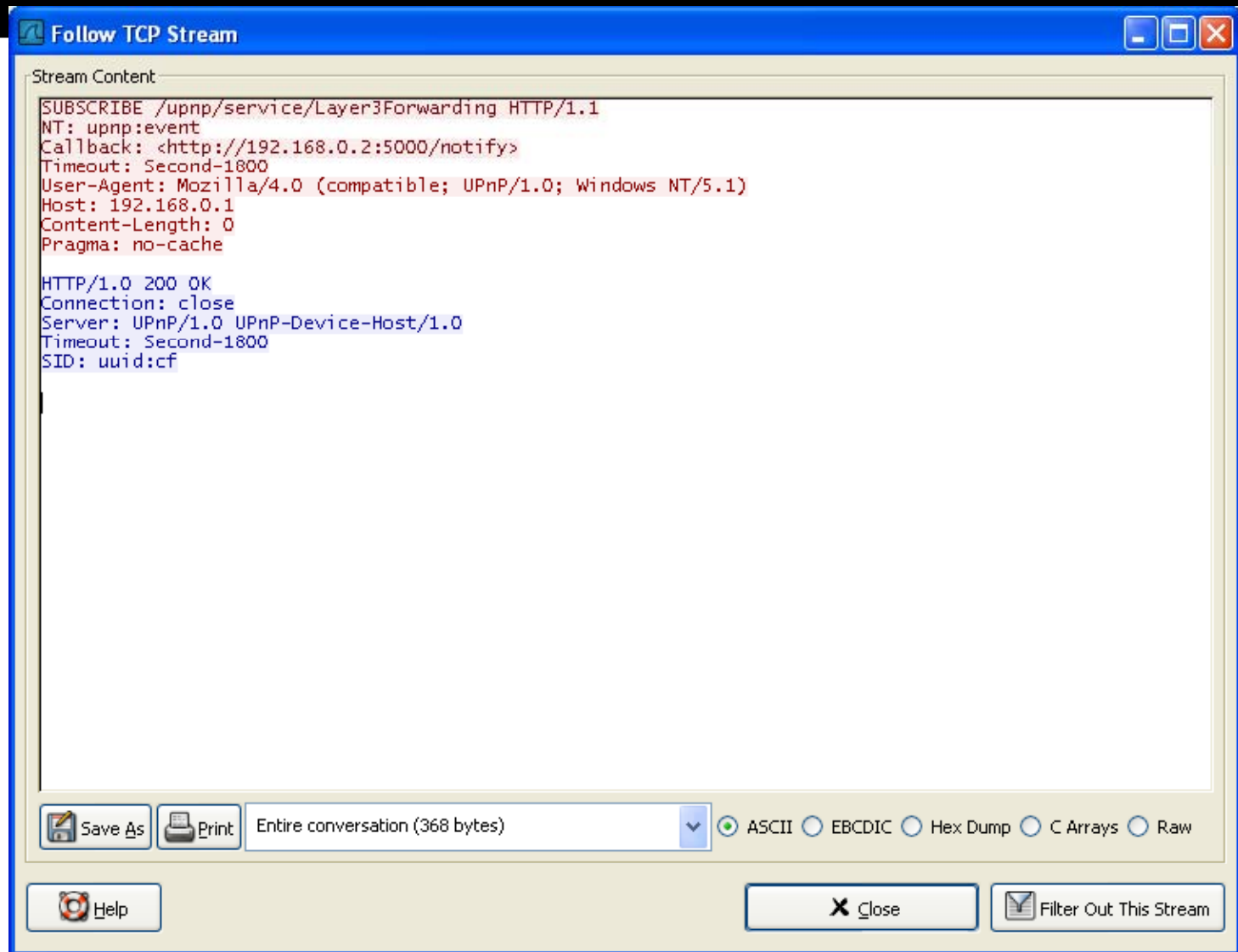
Buttons: Help, Start, Cancel

### Wireshark: Capture Interfaces

Description	IP	Packets	Packets/s	Stop
Adapter for generic dialup and VPN capture	unknown	0	0	Start Options Details
Broadcom NetXtreme Gigabit Ethernet Driver	157.163.15.28	110	1	Start Options Details

Buttons: Help, Close

# Wireshark: Screenshot



# Linux Tool: Abacus Port Sentry

Portscan detection daemon Port Sentry has the ability to detect portscans (including stealth scans) on the network interfaces of your machine

Upon alarm, it can block the attacker via `hosts.deny`, dropped route, or firewall rule

It is a part of the Abacus program suite



Source: <http://www.psionic.com>

# Abacus Port Sentry: Screenshot

```
[root@clubcm portsentry-1.1]# make linux
SYSTYPE=linux
Making
cc -O -Wall -DLINUX -DSUPPORT_STEALTH -o ./portsentry ./portsentry.c \
    ./portsentry_io.c ./portsentry_util.c
[root@clubcm portsentry-1.1]#
[root@clubcm portsentry-1.1]#
[root@clubcm portsentry-1.1]# make install
Creating psionic directory /usr/local/psionic
Setting directory permissions
Creating portsentry directory /usr/local/psionic/portsentry
Setting directory permissions
chmod 700 /usr/local/psionic/portsentry
Copying files
cp ./portsentry.conf /usr/local/psionic/portsentry
cp ./portsentry.ignore /usr/local/psionic/portsentry
cp ./portsentry /usr/local/psionic/portsentry
Setting permissions
chmod 600 /usr/local/psionic/portsentry/portsentry.ignore
chmod 600 /usr/local/psionic/portsentry/portsentry.conf
chmod 700 /usr/local/psionic/portsentry/portsentry

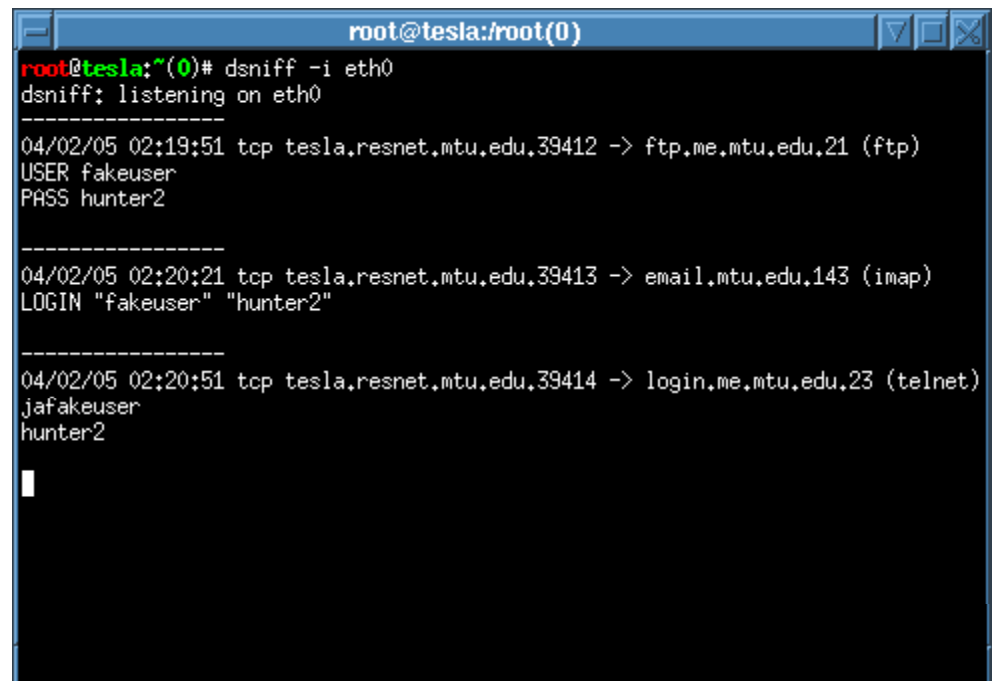
Edit /usr/local/psionic/portsentry/portsentry.conf and change
your settings if you haven't already. (route, etc)

WARNING: This version and above now use a new
directory structure for storing the program
and config files (/usr/local/psionic/portsentry).
Please make sure you delete the old files when
the testing of this install is complete.

[root@clubcm portsentry-1.1]#
[root@clubcm portsentry-1.1]#
```

A suite of powerful tools for sniffing networks for passwords and other information

Includes sophisticated techniques for defeating the "protection" of network switchers

A terminal window titled 'root@tesla:/root(0)' showing the execution of the 'dsniff' tool. The user runs 'dsniff -i eth0', and the tool starts listening on eth0. It captures three network sessions: 1) An FTP session to ftp.me.mtu.edu.21 with user 'fakeuser' and password 'hunter2'. 2) An IMAP session to email.mtu.edu.143 with user 'fakeuser' and password 'hunter2'. 3) A Telnet session to login.me.mtu.edu.23 with user 'jafakeuser' and password 'hunter2'.

```
root@tesla:/root(0)
root@tesla:~(0)# dsniff -i eth0
dsniff: listening on eth0

04/02/05 02:19:51 tcp tesla.resnet.mtu.edu.39412 -> ftp.me.mtu.edu.21 (ftp)
USER fakeuser
PASS hunter2

-----

04/02/05 02:20:21 tcp tesla.resnet.mtu.edu.39413 -> email.mtu.edu.143 (imap)
LOGIN "fakeuser" "hunter2"

-----

04/02/05 02:20:51 tcp tesla.resnet.mtu.edu.39414 -> login.me.mtu.edu.23 (telnet)
jafakeuser
hunter2
```

# Linux Tool: Hping2

hping2 is a network tool which sends custom ICMP/UDP/TCP packets and displays target replies like ping does with ICMP replies

It handles fragmentation and arbitrary packet body and size and can be used to transfer files under supported protocols

Using hping2, you can test firewall rules

```
# hping2 -n 172.16.240.241 -p 21 -S -c 4
eth0 default routing interface selected (according to /proc)
HPING 172.16.240.241 (eth0 172.16.240.241): S set, 40 headers + 0 data bytes
46 bytes from 172.16.240.241: flags=SA seq=0 ttl=63 id=0 win=5840 rtt=13.7 ms
46 bytes from 172.16.240.241: flags=SA seq=1 ttl=63 id=0 win=5840 rtt=3.7 ms
46 bytes from 172.16.240.241: flags=SA seq=2 ttl=63 id=0 win=5840 rtt=3.7 ms
46 bytes from 172.16.240.241: flags=SA seq=3 ttl=63 id=0 win=5840 rtt=3.5 ms

--- 172.16.240.241 hping statistic ---
4 packets trammitted, 4 packets received, 0% packet loss
round-trip min/avg/max = 3.5/6.1/13.7 ms
```



# Linux Tool: Sniffit

Sniffit is a packet sniffer and monitoring tool. It is a packet sniffer for TCP/UDP/ICMP packets

It is able to give you detailed technical info on these packets (SEC, ACK, TTL, Window, etc.) but also packet contents in different formats (hex or plain text, etc )

```

Sniffit 0.3.7 Beta
62.242.181.127 29400 -> 24.173.162.174 4662 : Unknown
24.173.162.174 4662 -> 62.242.181.127 29400 : Unknown
24.173.162.174 4662 -> 66.131.205.237 2107 : Unknown
24.173.162.174 4662 -> 69.167.231.28 1927 : Unknown
24.173.162.174 56553 -> 139.168.241.73 4662 : Unknown
24.173.162.174 4662 -> 195.241.94.48 7775 : Unknown
195.241.94.48 7775 -> 24.173.162.174 4662 : Unknown
82.45.102.74 1619 -> 24.173.162.174 4662 : Unknown
212.87.111.137 3165 -> 24.173.162.174 4662 : Unknown
69.167.231.28 1927 -> 24.173.162.174 4662 : Unknown
139.168.241.73 4662 -> 24.173.162.174 56553 : Unknown
24.173.162.174 4662 -> 212.87.111.137 3165 : Unknown
82.38.160.245 4662 -> 24.173.162.174 51398 : Unknown
66.56.118.204 3264 -> 24.173.162.174 4662 : Unknown
81.86.251.251 4884 -> 24.173.162.174 4662 : Unknown
84.137.222.17 63563 -> 24.173.162.174 4662 : Unknown
69.5.6.117 6994 -> 24.173.162.174 56333 : Unknown
24.173.162.174 56333 -> 69.5.6.117 6994 : Unknown
66.131.205.237 2107 -> 24.173.162.174 4662 : Unknown
24.173.162.174 54731 -> 81.86.190.11 5200 : Unknown
81.86.190.11 5200 -> 24.173.162.174 54731 : Unknown
80.53.230.66 36163 -> 24.173.162.174 4662 : Unknown
24.173.162.174 4662 -> 80.53.230.66 36163 : Unknown
24.173.162.174 58566 -> 65.32.5.121 119 : Unknown
-Sniffit 0.3.7 Beta 162.174 58566 : Unknown
IP packets/sec. : 104 221.122 4664 : Unknown
TCP packets/sec. : 82 162.174 50602 : Unknown
ICMP packets/sec. : 0 162.174 4662 : Unknown
UDP packets/sec. : 22 1.166.3 6667 : IRC
bytes/sec. (TCP) : 48741 162.174 56421 : IRC
bytes/sec. (UDP) : 1534

-Sniffit 0.3.7 Beta
Source IP : All Source PORT : All
Destination IP: All Destination PORT: All

Masks: F1-Source IP F2-Dest. IP F3-Source Port F4-Dest. Port
    
```

Source: <http://reptile.rug.ac.be>

The Nemesis Project is designed to be a command line-based and portable human IP stack for UNIX/Linux

The suite is broken down by protocol and should allow for useful scripting of injected packet streams from simple shell scripts

```
dolavimus:src/projects/nemesis/nemesis-1.4beta2/src
(dolavimus):nemesis/nemesis-1.4beta2/src$ 78: ./nemesis tcp help 16:48:36

TCP Packet Injection -- The NEMESIS Project Version 1.4beta2 (Build 14)

TCP usage:
  tcp [-v (verbose)] [options]

TCP options:
  -x <Source port>
  -y <Destination port>
  -f <TCP flags>
    -fS (SYN), -fA (ACK), -fR (RST), -fP (PSH), -fF (FIN), -fU (URG)
  -w <Window size>
  -s <SEQ number>
  -a <ACK number>
  -u <Urgent pointer offset>
  -o <TCP options file>
  -P <Payload file>

IP options:
  -S <Source IP address>
  -D <Destination IP address>
  -I <IP ID>
  -T <IP TTL>
  -t <IP TOS>
  -F <IP fragmentation offset>
  -O <IP options file>

Data Link Options:
  -d <Ethernet device>
  -H <Source MAC address>
  -M <Destination MAC address>

(dolavimus):nemesis/nemesis-1.4beta2/src$ 79: 16:51:11
```

Source: <http://jeff.wwti.com>



# Linux Tool: LSOF

List open files. Lsof is a Unix-specific diagnostic tool

Its name stands for LiSt Open Files and it does just that

It lists information about any files that are open by processes currently running on the system

```
File Edit View Terminal Tabs Help
baart@localhost:~/download/...  baart@localhost:~
baart@localhost ~ $ /usr/sbin/lsof /dev/null
COMMAND      PID  USER  FD  TYPE DEVICE SIZE NODE NAME
gnome-ses    8053 baart   0r  CHR  1,3   1355 /dev/null
dbus-laun    8074 baart   0r  CHR  1,3   1355 /dev/null
dbus-laun    8074 baart   1u  CHR  1,3   1355 /dev/null
dbus-laun    8074 baart   2u  CHR  1,3   1355 /dev/null
dbus-laun    8074 baart   3u  CHR  1,3   1355 /dev/null
dbus-daem    8075 baart   0u  CHR  1,3   1355 /dev/null
dbus-daem    8075 baart   1u  CHR  1,3   1355 /dev/null
dbus-daem    8075 baart   2u  CHR  1,3   1355 /dev/null
dbus-daem    8075 baart   4u  CHR  1,3   1355 /dev/null
gconfd-2     8080 baart   0u  CHR  1,3   1355 /dev/null
gconfd-2     8080 baart   1u  CHR  1,3   1355 /dev/null
gconfd-2     8080 baart   2u  CHR  1,3   1355 /dev/null
gconfd-2     8080 baart   3u  CHR  1,3   1355 /dev/null
bonobo-ac    8086 baart   0u  CHR  1,3   1355 /dev/null
bonobo-ac    8086 baart   1u  CHR  1,3   1355 /dev/null
bonobo-ac    8086 baart   2u  CHR  1,3   1355 /dev/null
at-spi-re    8088 baart   0u  CHR  1,3   1355 /dev/null
at-spi-re    8088 baart   1u  CHR  1,3   1355 /dev/null
at-spi-re    8088 baart   2u  CHR  1,3   1355 /dev/null
gnome-key    8090 baart   0r  CHR  1,3   1355 /dev/null
gnome-key    8090 baart   1w  CHR  1,3   1355 /dev/null
gnome-set    8097 baart   0u  CHR  1,3   1355 /dev/null
```

Source: <ftp://vic.cc.purdue.edu>

Copyright © by **EC-Council**

All Rights Reserved. Reproduction is Strictly Prohibited

# Linux Tool: IPTraf

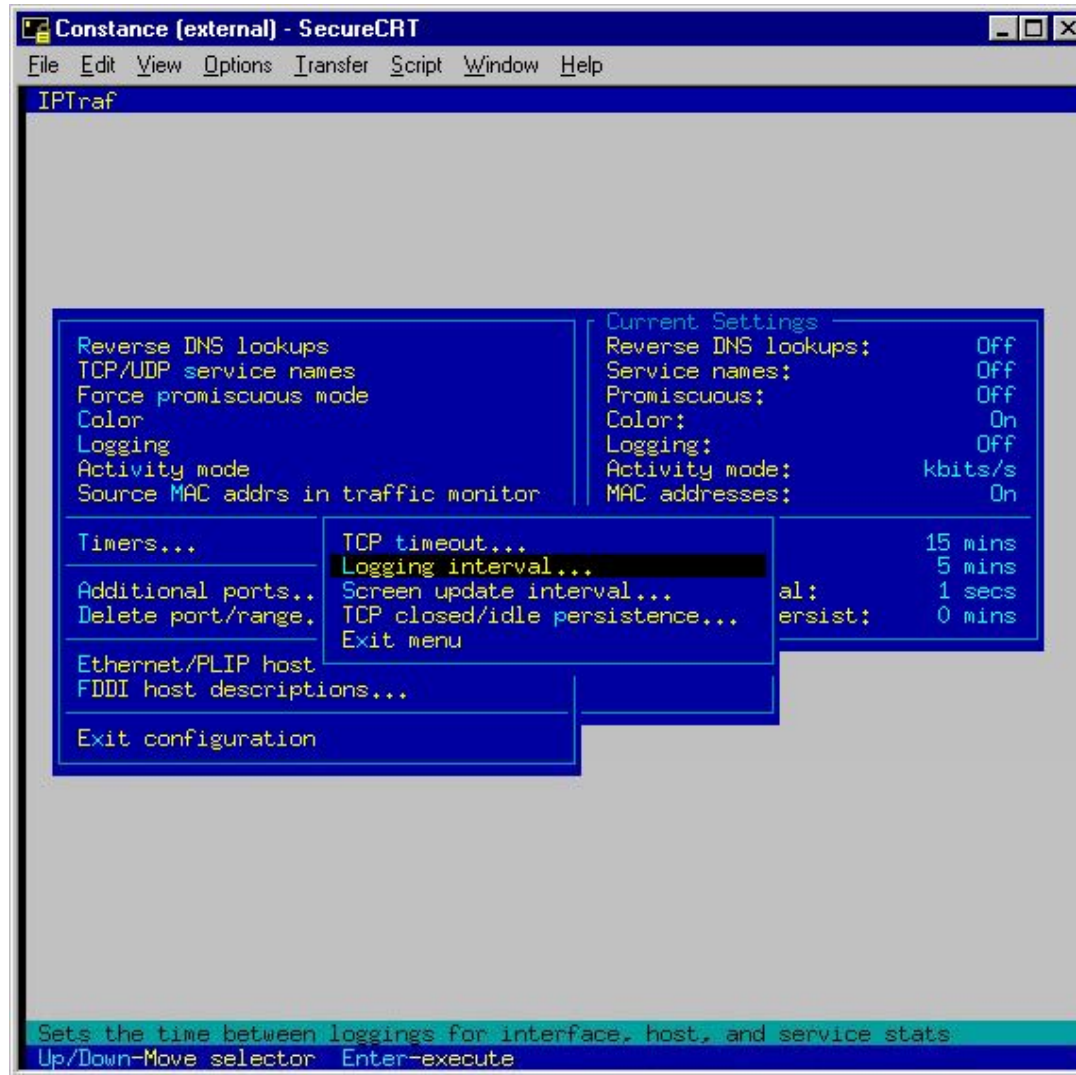
Interactive Colorful IP LAN Monitor, IPTraf is an ncurses-based IP LAN monitor that generates various network statistics including TCP info, UDP counts, ICMP and OSPF information, Ethernet load info, node stats, IP checksum errors, and others

```

IPTraf
Proto/Port ----- Pkts --- Bytes --- PktsTo - BytesTo PktsFrom BytesFrom
TCP/www          6064  1960227  3490  387688  2574  1572539
TCP/8088         1328  411655  647   71848  681   339807
TCP/webcache     545   209710  269   21707  276   188003
TCP/pop3        508   169510  220   8952   288   160558
TCP/smtp        177   86150   88    79197  89    6953
UDP/domain      352   40643  192   13357  160   27286
TCP/netbios-ss  160   22112  86    9408   74    12704
UDP/netbios-ns  164   15530  130   10337  34    5193
TCP/https       22    7533   12    1553   10    5980
TCP/telnet      45    4649   25    2052   20    2597
TCP/ftp         25    1269   13    746    12    523
UDP/netbios-dg  5     1177   3     703    2     474
TCP/rntp        7     578    4     213    3     365
TCP/74          6     564    6     564    0     0
TCP/40          9     540    9     540    0     0
UDP/bootps     1     328    1     328    0     0
UDP/bootpc     1     328    0     0      1     328
UDP/ntp        8     608    4     304    4     304
TCP/81         7     332    5     252    2     80
TCP/tproxy     9     508    9     508    0     0
26 entries ----- Elapsed time: 0:00
Protocol data rates (kbits/s): 165.25 in 537.00 out 702.25 total
Up/Down/PgUp/PgDn-scroll window S-sort X-exit
    
```

Source: <http://cebu.mozcom.com>

# IPTraf: Screenshot 1



# IPTraf: Screenshot 2

```
192.168.0.2 - PuTTY
IPTraf
Statistics for eth0
-----

```

	Total Packets	Total Bytes	Incoming Packets	Incoming Bytes	Outgoing Packets	Outgoing Bytes
Total:	645	93258	218	22764	427	70494
IP:	645	84216	218	19700	427	64516
TCP:	642	82488	215	17972	427	64516
UDP:	3	1728	3	1728	0	0
ICMP:	0	0	0	0	0	0
Other IP:	0	0	0	0	0	0
Non-IP:	0	0	0	0	0	0

```

Total rates:          36.8 kbits/sec      Broadcast packets:      3
                   33.6 packets/sec      Broadcast bytes:       1770

Incoming rates:       9.7 kbits/sec
                   11.4 packets/sec

Outgoing rates:       27.1 kbits/sec
                   22.2 packets/sec

Elapsed time: 0:00
X-exit

```

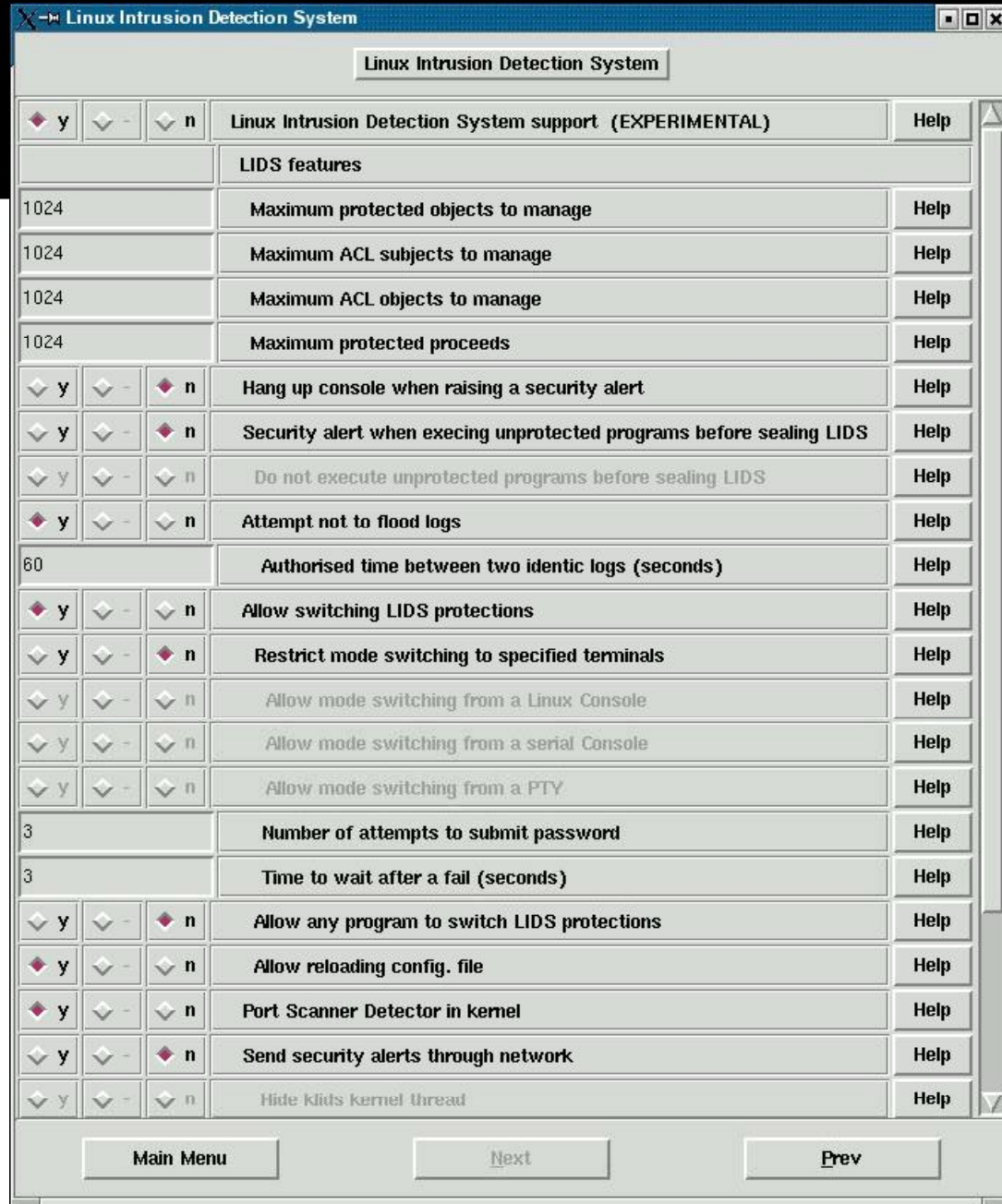
# Linux Tool: LIDS

The LIDS is an intrusion detection/defense system in the Linux kernel

The goal is to protect Linux systems against root intrusions by disabling some system calls in the kernel itself



Source: <http://www.lids.org/>



# Hacking Tool: Hunt

Hunt is a session hijacking tool

One of Hunt's advantages over other session hijacking tools is that it uses techniques to avoid ACK storms

Hunt avoids the ACK storm, and the dropping of the connection, by using ARP spoofing to establish the attacker's machine as a relay between the source and the destination

Now, the attacker uses Hunt to sniff the packets to the source and destination to send over this connection

The attacker can choose to act as a relay and forward these packets to their intended destinations, or he can hijack the session

The attacker can type in commands that are forwarded to a destination but that the source cannot see

Source: <http://lin.fsid.cvut.cz/~kra/index.html>

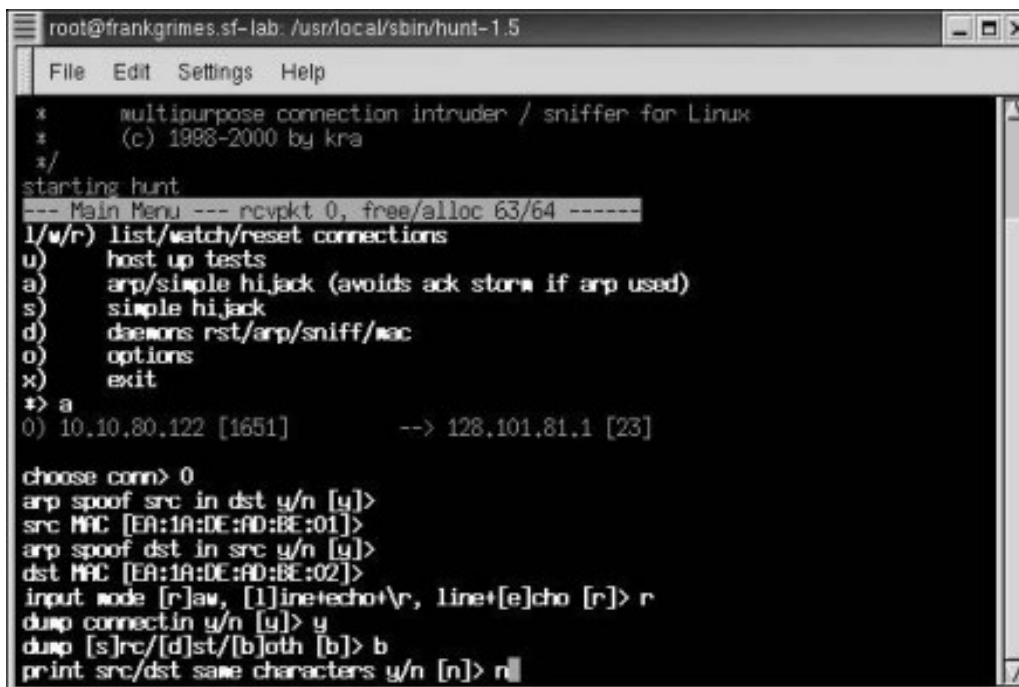
Copyright © by **EC-Council**



# Hunt: Screenshot

Any commands the source types in can be seen on the attacker's screen, but they are not sent to the destination

Then Hunt allows the attacker to restore the connection back to the source when he/she is done with it



```
root@frankgrimes.sf-lab: /usr/local/sbin/hunt-1.5
File Edit Settings Help
* multipurpose connection intruder / sniffer for Linux
* (c) 1998-2000 by kra
*/
starting hunt
--- Main Menu --- rcvpkt 0, free/alloc 63/64 -----
l/w/r) list/watch/reset connections
u) host up tests
a) arp/simple hijack (avoids ack storm if arp used)
s) simple hijack
d) daemons rst/arp/sniff/mac
o) options
x) exit
*-> a
0) 10.10.80.122 [1651] --> 128.101.81.1 [23]

choose conn> 0
arp spoof src in dst y/n [y]>
src MAC [EA:1A:DE:AD:BE:01]>
arp spoof dst in src y/n [y]>
dst MAC [EA:1A:DE:AD:BE:02]>
input mode [r]aw, [l]ine+echo+v, line+[e]cho [r]> r
dump connectin y/n [y]> y
dump [s]rc/[d]st/[b]oth [b]> b
print src/dst same characters y/n [n]> nll
```



# Tool: TCP Wrappers

TCP Wrappers allow the user to monitor/filter incoming requests for SYSTAT, FINGER, FTP, TELNET, R-Commands, TFTP, TALK, and other network services

It provides access control to restrict what systems connect with which network daemons

It provides some protection from host spoofing

It has 4 components:

- Tcpsd—the actual wrapper program
- Tcpsmatch, tcpschk—ACL testing programs
- Try-from—tests host lookup function
- Safe-finger—a better version of finger



# Linux Loadable Kernel Modules

LKMs are Loadable Kernel Modules used by the Linux kernel to expand its functionality

The advantage of those LKMs: *They can be loaded dynamically*; there must be no recompilation of the whole kernel. Because of these features, they are often used for specific device drivers (or filesystems) such as soundcards

This command forces the system to do :

- Load the objectfile (here module.o)
- Call `create_module` syscall (for systemcalls -> see I.2) for relocation of memory
- Unresolved references are resolved by Kernel-Symbols with the syscall `get_kernel_syms`
- After this, the `init_module` syscall is used for the LKM initialization -> executing `int init_module(void)` and so on

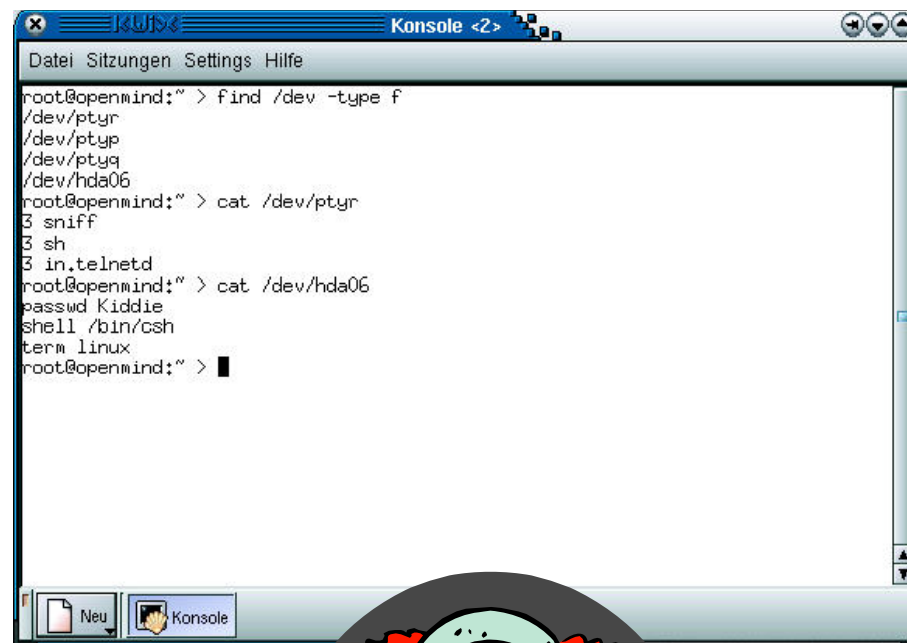


# Hacking Tool: Linux Rootkits

By installing a rootkit, an intruder can maintain access to a compromised system

A rootkit contains a set of tools and replacement executables for many of the operating system's critical components, used to hide evidence of the attacker's presence and to give the attacker backdoor access to the system

Rootkits require root access to install, but once set up, the attacker can get root access back at any time



```
root@openmind:~$ find /dev -type f
/dev/ptyr
/dev/ptyr
/dev/ptyq
/dev/hda06
root@openmind:~$ cat /dev/ptyr
3 sniff
3 sh
3 in.telnetd
root@openmind:~$ cat /dev/hda06
passwd Kiddie
shell /bin/csh
term linux
root@openmind:~$
```



# Rootkits: Knark & Torn

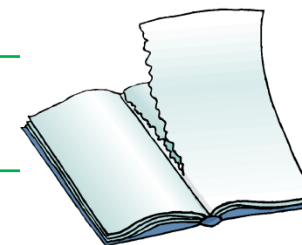


## Knark:

- The following is the list of files that comes along with Knark:
  - Makefile, apache.c, Apache.cgi, backup, Bj.c, caine, Clearmail, dmsg, Dmsg, ered, Exec, fix, Fixtext, ftpt, Gib, gib.c, Hds0, hidef, Inc.h, init, Lesa, login Lpdx, lpx.c, Make-ssh-host-key, make-ssh-known-hosts, Module, nethide, Pgr, removeme, Rexec, rkhelp, sl2 Sl2.c, snap, Ssh\_config, sshd\_config, Ssht, statdx2 , Sysmod.o, sz, T666, unhidef, Wugod, zap
- KNARK comes with a few good exploits as well, such as Lpdx, T666, Wugod

## Torn:

- First rootkit of its kind that is precompiled and yet allows the user to define a password; the password is stored in an external encrypted file



# Rootkits: Tuxit, Adore, Ramen

## Tuxit

- Written by a Dutch group called Tuxtendo
- There are six files in the tuxkit, which include a README, an installation script, and four tarred/zipped files

## Adore

- Adore is a worm that was originally known as the Red Worm
- LPRng is installed by default on Red Hat 7.0 systems. From the reports so far, Adore started to spread from April 1, 2001

## Ramen

- It is a Linux-based Internet worm named after the popular noodle soup
- It has been seen in the wild affecting systems that run Red Hat Inc.'s 6.2 or 7.0 versions of the open-source OS



# Rootkit: Beastkit

Beastkit replaces common binaries that can be used to monitor system operations (like ps) and the list of programs included in the rootkit (bin.tgz)

The timestamp does not change, because the rootkit uses touch -acmr to transmit the timestamp to the rootkit files

Beastkit contains some tools (bktools) (placed at /lib/ldd.so/bktools):

- bkget - SynScan Daemon (by psychoid/tCl)
- bkp - hdlp2
- bks - Sniffer
- bkspb - "sauber"-Script (see duarawkz-rootkit), cleans up some of the intruder's traces
- bkscan - SynScan (by psychoid/tCl)
- bktd
- patch - SSHd-Patchscript (update to ssh-1.2.32 using ftp)
- prl - SSHd-Patchscript (update to ssh-1.2.32 using http)
- prw - SSHd-Patchscript (update to ssh-1.2.32)





`chkrootkit` is a tool to locally check for signs of a rootkit

It contains `chkrootkit`, a shell script that checks system binaries for rootkit modification

```
Searching for Ramen Worm files and dirs... nothing found
Searching for Maniac files and dirs... nothing found
Searching for RK17 files and dirs... nothing found
Searching for Ducoci rootkit... nothing found
Searching for Adore Worm... nothing found
Searching for ShitC Worm... nothing found
Searching for Omega Worm... nothing found
Searching for Sadmin/IIS Worm... nothing found
Searching for MonKit... nothing found
Searching for Showtee... nothing found
Searching for OpticKit... nothing found
Searching for T.R.K... nothing found
Searching for Mithra... nothing found
Searching for LOC rootkit ... nothing found
Searching for Romanian rootkit ... nothing found
Searching for Suckit rootkit ... nothing found
Searching for Volc rootkit ... nothing found
Searching for Gold2 rootkit ... nothing found
Searching for TC2 Worm default files and dirs... nothing found
Searching for Anonoying rootkit default files and dirs... nothing found
Searching for ZK rootkit default files and dirs... nothing found
Searching for ShKit rootkit default files and dirs... nothing found
Searching for anomalies in shell history files... nothing found
Checking 'asp'... not infected
Checking 'bindshell'... not infected
Checking 'lkm'... You have 1 process hidden for readdir command
You have 1 process hidden for ps command
Warning: Possible LKM Trojan installed
Checking 'rexedcs'... not found
Checking 'sniffer'... Checking 'w55808'... not infected
Checking 'wted'... nothing deleted
Checking 'scalper'... not infected
Checking 'slapper'... not infected
Checking 'z2'... nothing deleted
root@sys: ~/chkrootkit-0.42b#
```

Source: <http://www.chkrootkit.org>

# 'chkrootkit' detects the following Rootkits

1. Irc3, Irc4, Irc5, Irc6 (and some variants);
2. Solaris rootkit;
3. FreeBSD rootkit;
4. t0rn (including some variants and t0rn v8);
5. Ambient's Rootkit for Linux (ARK);
6. Ramen Worm;
7. rh[67]-shaper;
8. RSHA;
9. Romanian rootkit;
10. RK17; Lion Worm;
11. Adore Worm;
12. LPD Worm;
13. kenny-rc;
14. Adore LKM;
15. ShitC Worm;
16. Omega Worm;
17. Wormkit Worm;
18. Maniac-RK;
19. |dec-rootkit;
20. Duccoi rootkit;
21. zc Worm;
22. BST.b trojan;
23. duserwitz;
24. kmark LKM;
25. Monkkit;
26. Hidrootkit; Bobkit;
27. Pindakit;
28. t0rn (v8.0 variant);
29. Showtee;
30. Optickit;
31. T.R.K.;
32. MchRa's Rootkit;
33. George;
34. SuckIT;
35. Scalper (FreeBSD/Apache chunked encoding worm);
36. Slapper A, B, C and D;
37. (Linux/Apache mod\_ssl Worm);
38. OpenBSD rk v1;
39. Illogic rootkit;
40. SK rootkit;
41. sobek LKM;
42. Romanian rootkit;
43. LOC rootkit;





## Rain.Forest.Puppy's

- CGI vulnerability scanner

Source: <http://www.dwheeler.com>

## Flawfinder:

- It is a Python program that searches through the source code for potential security flaws, listing them sorted by risk, with the most potentially dangerous flaws shown first. The risk level depends not only on the function, but also on the values of the parameters of the function

Source: <http://www.wiretrip.net>

## StackGuard

- A compiler that emits programs hardened against "stack smashing" attacks. Stack smashing attacks are a common form of penetration attack. Programs that have been compiled with StackGuard are largely immune to stack smashing attack. Protection requires no source code changes at all

Source: <http://www.immunix.org>

## Libsafe

- It is generally accepted that the best solution to buffer overflow and format string attacks is to fix the defective programs

Source: <http://www.avayalabs.com>

# Advanced Intrusion Detection Environment (AIDE)

AIDE (Advanced Intrusion Detection Environment) is a free replacement for Tripwire

It creates a database from the regular expression rules that it finds from the config file

Once this database is initialized, it can be used to verify the integrity of the files

This first AIDE database is a snapshot of the system in its normal state and the yardstick by which all subsequent updates and changes will be measured

**NMap** (<http://www.insecure.org/nmap>)

- Premier network auditing and testing tool

**LSOF** (<ftp://vic.cc.purdue.edu/pub/tools/unix/lsof>)

- LSOF lists open files for running Unix/Linux processes

**Netcat** (<http://www.atstake.com/research/tools/index.html>)

- Netcat is a simple Unix utility that reads and writes data across network connections, using TCP or UDP protocol

**Hping2** (<http://www.kyuzz.org/antirez/hping/>)

- Hping2 is a network tool to send custom ICMP/UDP/TCP packets and to display target replies like ping does with ICMP replies

**Nemesis** (<http://www.packetninja.net/nemesis/>)

- The Nemesis Project is designed to be a command-line based, portable human IP stack for Unix/Linux

## Stunnel (<http://www.stunnel.org>)

- Stunnel is a program that allows the user to encrypt arbitrary TCP connections inside SSL (Secure Sockets Layer) available on both Unix and Windows. Stunnel can allow the user to secure non-SSL aware daemons and protocols (like POP, IMAP, NNTP, LDAP, etc.) by having Stunnel provide the encryption, requiring no changes to the daemon's code

## OpenSSH /SSH (<http://www.openssh.com/>)

- SSH (Secure Shell) is a program for logging into a remote machine and for executing commands on a remote machine. It provides secure encrypted communications between two untrusted hosts over an insecure network

## GnuPG (<http://www.gnupg.org>)

- GnuPG is a complete and free replacement for PGP. Since it does not use the patented IDEA algorithm, it can be used without any restrictions

## MRTG (<http://www.mrtg.org>)

- The Multi-Router Traffic Grapher (MRTG) is a tool to monitor the traffic load on network-links

## Swatch (<http://www.stanford.edu/~atkins/swatch/>)

- Swatch, the simple watch daemon, is a program for Unix system logging

## Timbersee (<http://www.fastcoder.net/~thumper/software/sysadmin/timbersee/>)

- Timbersee is a program similar to the Swatch program

## Logsurf (<http://www.cert.dfn.de/eng/logsurf/>)

- The program log surfer was designed to monitor any text-based logfiles on the system in real time

# Linux Tools: Log and Traffic Monitors (cont'd)

## TCP Wrappers (<ftp://ftp.prcupine.org/pub/security/index.html>)

- Wietse Venema's network logger, also known as TCPD or LOG\_TCP. These programs log the client hostname of incoming telnet, ftp, rsh, rlogin, finger, etc. requests

## IPLog (<http://ojnk.sourceforge.net/>)

- iplog is a TCP/IP traffic logger. Currently, it is capable of logging TCP, UDP, and ICMP traffic

## IPTraff (<http://cebu.mozcom.com/riker/iptraff/>)

- IPTraff is an ncurses-based IP LAN monitor that generates various network statistics including TCP info, UDP counts, ICMP, OSPF information, Ethernet load info, node stats, IP checksum errors, and others

## Ntop (<http://www.ntop.org>)

- ntop is a Unix/Linux tool that shows network usage, similar to what the popular "top" Unix/Linux command does

# Linux Security Auditing Tool (LSAT)

LSAT is a post install security auditor for Linux and Unix

It checks for system configurations and local network settings on the system for common security/config errors and for packages that are not needed

LSAT consists:

- `checkcfg`, `checkdotfiles`, `checkfiles`,  
`checkftpusers`, `checkhostsfiles`,  
`checkinetd`, `checkinittab`,  
`checkissue`, `checkkbd`, `checklimits`,  
`checklogging`, `checkmodules`,  
`checkmd5`, `checknet`, `checknetforward`,  
and `checkset`, to name a few





## Physical Security:

lock your computer physical in a secure place.

## Password Security:

Do not assign easy-to-guess password.

Do not share your account with other person.

Check user account with null passwd (without passwd) in /etc/shadow:

## Network Security:

Close the door first by denying access from network by default.

```
$ cat "ALL:ALL" >> /etc/hosts.deny
```

Stop all unused services such as sendmail, NFS.

```
$ chkconfig --list
```

```
$ chkconfig --del sendmail
```

```
$ chkconfig --del nfslock
```

```
$ chkconfig --del rpc
```

Check system logs in /var/log regularly especially /var/log/secure.

## Update your Linux system regularly.

Checking the errata (bug fixes) in

<http://www.redhat.com/support/errata>

The update packages can be found in <ftp://updates.redhat.com>

# Steps for Hardening Linux

Minimizing installed software

Patching the system

Securing filesystem permissions and S\*ID binaries

Improving login and user security

Setting some physical and boot security controls

Securing the daemons via network access controls

Increasing logging and audit information

Configuring vendor supplied security software (IDS, host firewall)



# What Happened Next

Immediately after this attack, Jason, an Ethical Hacker, was called in to troubleshoot the problem.

After hours of tracing the systems of top-shopper.com, Jason found that access rights for the users and groups were set to default which hackers exploited to attack the systems.

Jason suggested the following measures to top-shopper.com:

- Install the operating system properly
- Set up and enable iptables
- Configure security related kernel parameters
- Disable unnecessary daemons and network services
- Change default passwords and create regular users
- Disable remote root logins over ssh

Linux is gaining in popularity and is fast becoming a stable industry strength OS

Once the IP address of a target system is known, an attacker can begin port scanning, looking for holes in the system for gaining access; Nmap being a popular tool

Password cracking tools are available for Linux as well

Sniffers, as well as packet assembly/analyzing tools for Linux, provide attackers with the edge that they have when dealing with other OSs

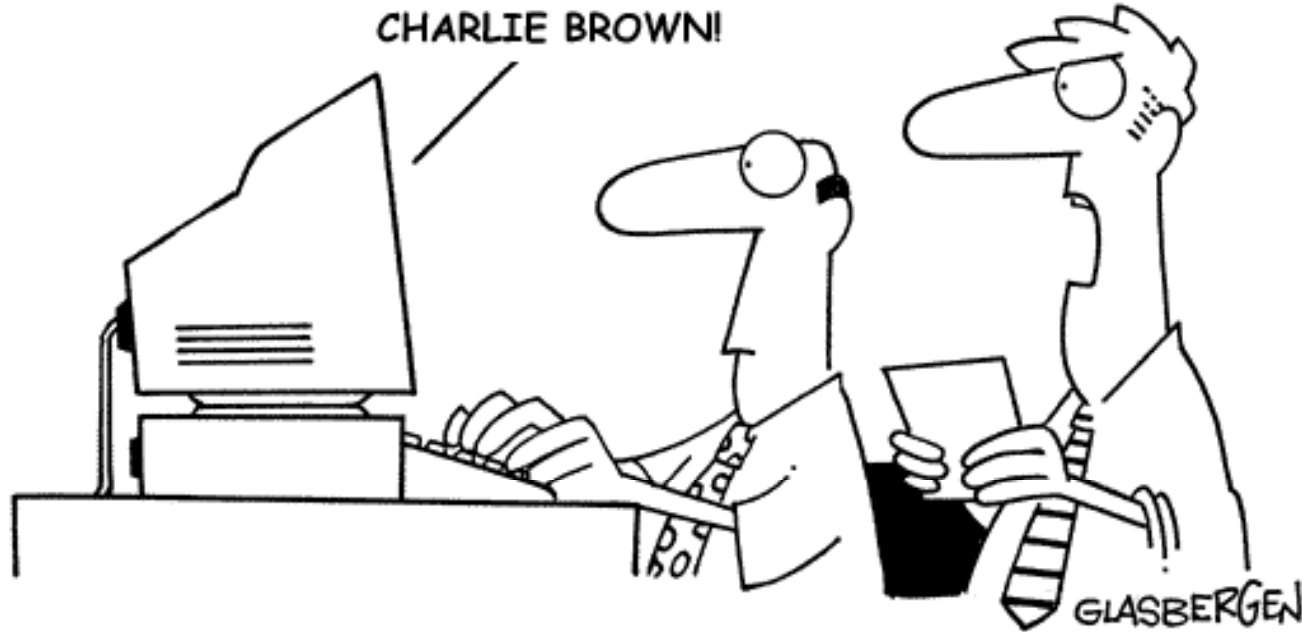
Attackers with root privileges can engage in session hijacking as well

Trojans, backdoors, and worms are also prevalent in the Linux environment

As with any other system, a well-developed integrated procedure is to be put in place to counter the threats that exist

© 2000 Randy Glasbergen. [www.glasbergen.com](http://www.glasbergen.com)

YOU'RE A  
BLOCKHEAD,  
CHARLIE BROWN!



**“It’s right here in your own handwriting.  
You didn’t ask for Linux, you asked for Linus!”**

© 1999 Randy Glasbergen. [www.glasbergen.com](http://www.glasbergen.com)



**"It's the latest innovation in office safety.  
When your computer crashes, an air bag is activated  
so you won't bang your head in frustration."**

Copyright 2003 by Randy Glasbergen.  
[www.glasbergen.com](http://www.glasbergen.com)



**“I’ve been using the same computer since 1980.  
They can’t replace it without violating the  
company’s age discrimination policies.”**