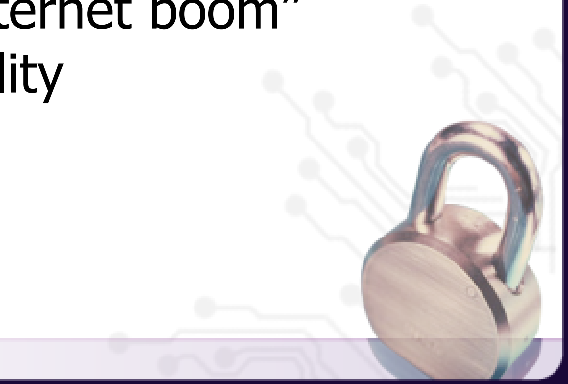
Is Your Firewall Enough?
Tools to Improve the Security of Your Site

Ned McClain
Trent R. Hein

Applied Trust Engineering

Why Now?

- Worldwide explosion of the Internet has produced an abundance of skilled hackers
- Down market slowed corporate investment in technology/infrastructure but not intrusion technology development
- Only strong companies will survive stormy market, security is required for strength
- Public awareness/concern for security and privacy has reached a threshold level
- Commerce technology developed during the “Internet boom” introduces new dimensions of security vulnerability



What is Security?

Security is…

- Vigilance
- Knowledge
- Risk management
- Methodology and policies
- Applied science / forensics
- Architecture
- Implementation
- Operations

Security Myths

Myth #1: “We aren’t a likely target of attack.”

Fact: 91% of CSI/FBI Computer Crime Survey respondents reported detecting a breach in the prior 12 months.

Security Myths

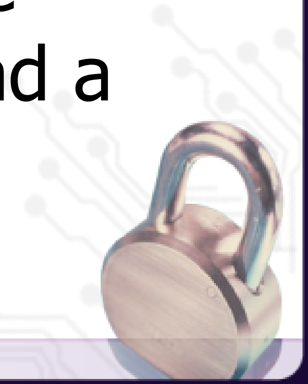
Myth #2: “70% of attacks involve insiders.”

Fact: Actually, this used to be true, but in the last 24 months the ratio has inverted. Today, only 30% of attacks involve insiders.

Security Myths

Myth #3: "We're secure because we have a firewall."

Fact: Hardly anything could be further from the truth. In the CSI/FBI Computer Crime survey, 95% of organizations surveyed had a standard commercial firewall in place.



Security Myths

Myth #4: “We haven’t been broken into, therefore we are secure.”

Fact: Most break-ins go undetected for more than 6 months.

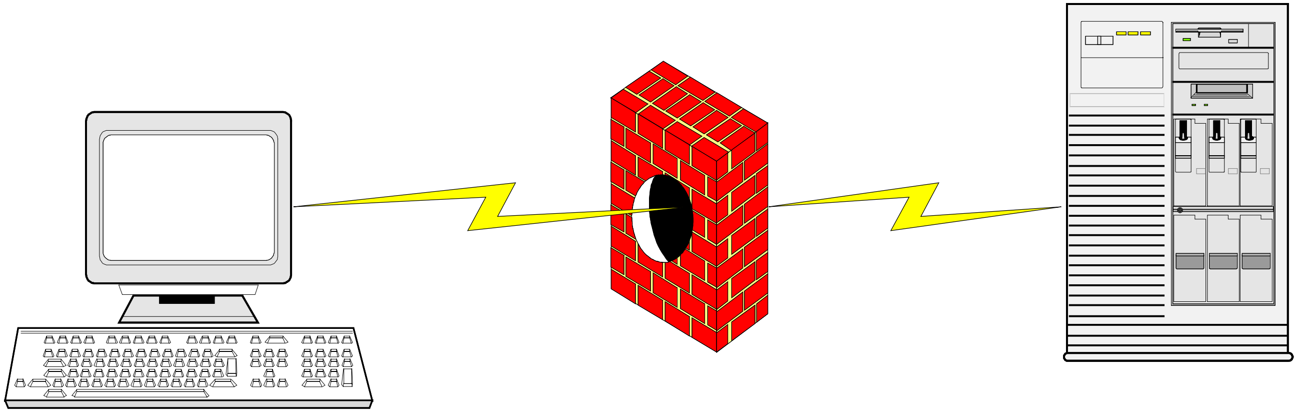
The Seven Common-Sense Rules of Rodent Infestation

1. Don’t leave food lying around
2. Plug the holes they use to get into the house
3. Don’t provide places that make good mouse “nests”
4. Set traps
5. Check traps daily
6. Don’t use bait-and-kill poison
7. Get a cat!

The Seven Common-Sense Rules of Security

1. Don’t provide online access to extraordinarily interesting files
2. Close holes that can be used to gain access to your system
3. Don’t provide “nests” for hackers to establish a base
4. Set traps to detect intrusions
5. Monitor reports generated by your security monitoring tools
6. Teach yourself about security
7. Vigilantly look for unusual activity

Network communication basics



Source Address: 172.16.30.1

Destination Address: 10.0.1.5

Source Port: 4302

Destination Port: 25

Protocol: TCP

Source Address: 10.0.1.5

Destination Address: 172.16.30.1

Source Port: 25

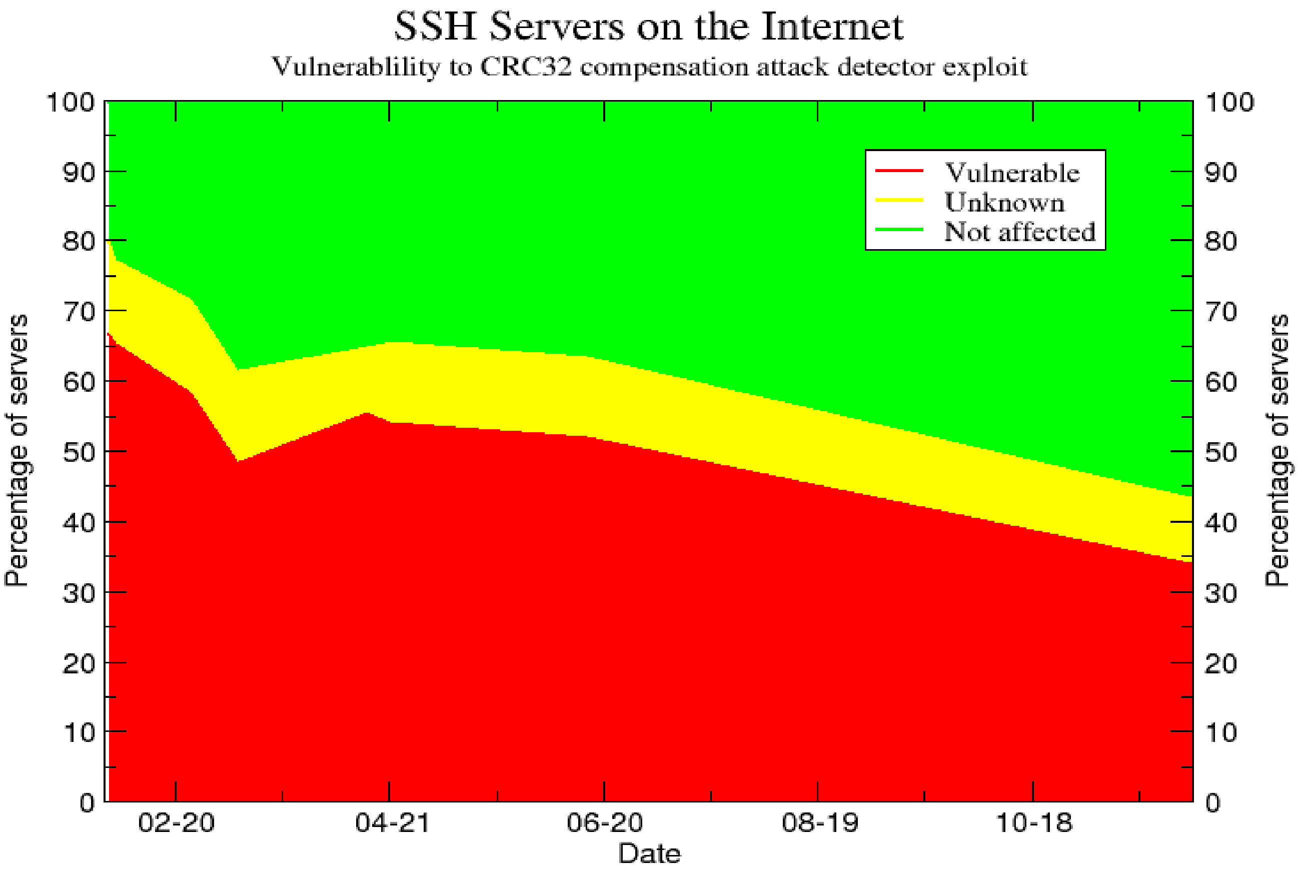
Destination Port: 4302

Protocol: TCP

UNIX security tools you should be aware of

- Nmap: The über-port scanner
- Ndiff: Track changes in network services
- Nessus: Remote vulnerability assessment
- The Coronor’s Toolkit: UNIX system forensics
- Syslog-ng: Centralized log management
- Checksyslog: Realistic log file management

SSH Server vulnerability statistics – 2001



(from http://www.citi.umich.edu/techreports/reports/citi-tr-01-13.pdf)

provos@citi.umich.edu

Nmap: the über-port scanner

- Identifies services and hosts on a network using ICMP ECHO (ping) sweeps and (connecting to TCP, UDP, and RPC ports)

- Provides several other cool network scanning features
- Runs on almost every OS (even Win32)
- GUI front-ends available

- Download from: http://www.nmap.org

- Be a good neighbor. Note that even the simplest NIDS packages can detect nmap scans (see www.snort.org)

Nmap: simplest use

```
unix% nmap gerbil.atrust.com

Starting nmap V. 2.54BETA30 ( www.insecure.org/nmap/ )
Interesting ports on bull.atrust.com (192.168.1.1):
(The 1541 ports scanned but not shown below are in state: closed)
Port       State     Service
22/tcp     open      ssh
25/tcp     open      smtp
80/tcp     open      http
110/tcp    open      pop-3
139/tcp    open      netbios-ssn
143/tcp    open      imap2
515/tcp    open      printer
993/tcp    open      imaps
995/tcp    open      pop3s
Nmap run completed -- 1 IP address (1 host up) scanned in 2 seconds
```

Nmap: typical use

TCP SYN "stealth" scan

```
unix# nmap –sS –O –p1-65535 192.168.1.1-10

Starting nmap V. 2.54BETA30 ( www.insecure.org/nmap/ )
Interesting ports on bull.atrust.com (192.168.1.1):
(The 65525 ports scanned but not shown below are in state: closed)
Port       State     Service
22/tcp     open      ssh
25/tcp     open      smtp
80/tcp     open      http
110/tcp    open      pop-3
139/tcp    open      netbios-ssn
143/tcp    open      imap2
515/tcp    open      printer
993/tcp    open      imaps
995/tcp    open      pop3s
4000/tcp   open      unknown
Remote operating system guess: Linux Kernel 2.4.0 - 2.4.9 (X86)
Uptime 89.727 days (since Mon Nov 12 18:12:10 2001)
Nmap run completed -- 10 IP addresses (1 host up) scanned in 89 seconds
```

Nmap: additional features

- Specify ranges of IPs to scan: 192.168.1.0/24 or 192.168.1-4.*
- Verbose runtime messages (-v), extra verbosity (-vv)
- UDP port scan (-sU)
- Higher-level protocol scans: RPC (-sR), Ident (-sI)
- Disable pinging hosts before scanning them (-P0)
- Don't do DNS resolution (-n)
- Alternate output formats: XML (-oX *filename*), machine-parsable (-oM), grepable (-oG *filename*), human readable (-oN *filename*)
- Several malicious features: forge decoy source addresses (-D*fakeIP*), various scan speeds (-T Sneaky, -T Aggressive), various alternate scanning methods (-sX, -sF, -sN)

Ndiff: managing Nmap information

- Calculate the difference between two Nmap scans
  - New hosts
  - Missing hosts
  - Changed hosts (TCP/UDP ports that are opened or closed)

- Includes three Perl scripts
  - Ndiff: Compare two Nmap files
  - Ngen: Create baseline from user definition or Nmap file
  - Nrun: Run nmap and ndiff in a scalable, manageable manner

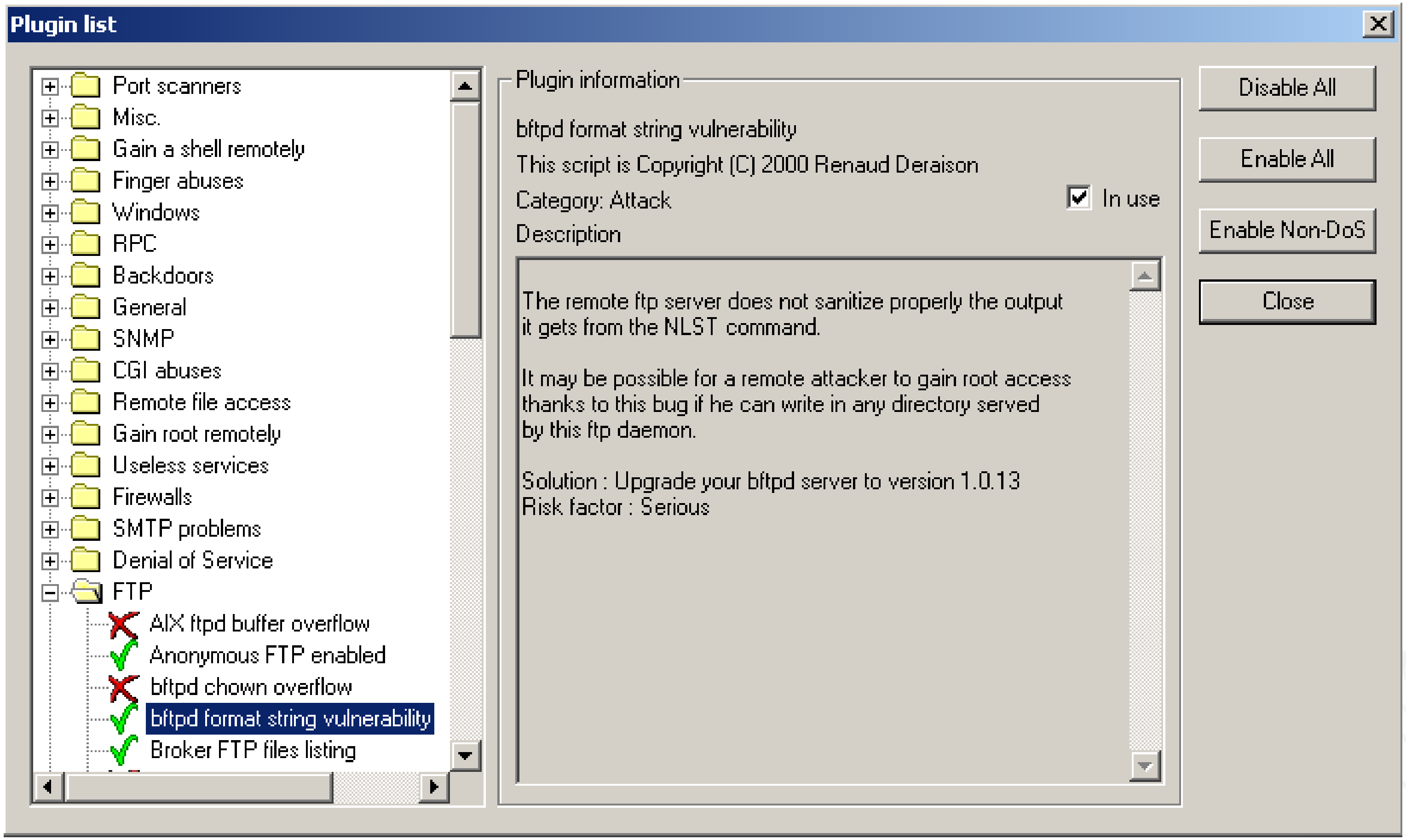
- It’s really effective to start nrun out of cron regularly

- http://www.vinecorp.com/ndiff

Nessus: remote vulnerability assessment

- Identifies By Renaud Deraison and Jordan Hrycaj
- Requires both a server and a client
  - Server runs on most UNIX-like OSes
  - UNIX Command Line, X11, Java, and Win32 clients
- Over eight hundred vulnerability plugins
  - Easy to upgrade: #./nessus-update-plugins
  - Exploit database categorized: Gain a shell remotely, CGI abuses, Backdoors, Remote file access, Denial of Service, Useless services, NIS, Finger abuses, Firewalls, Misc., FTP, Gain root remotely, SMTP problems, Port scanners, RPC
- Provides encryption between client and server (PEKS or SSL)
- Customizable reports in text, HTML, or PDF
  - Support for false positives
  - Schedule regular, reoccurring network scans
- http://www.nessus.org/

Nessus plugin configuration



- Port scanners
- Misc.
- Gain a shell remotely
- Finger abuses
- Windows
- RPC
- Backdoors
- General
- SNMP
- CGI abuses
- Remote file access
- Gain root remotely
- Useless services
- Firewalls
- SMTP problems
- Denial of Service
- FTP
  - ✗ AIX ftpd buffer overflow
  - ✓ Anonymous FTP enabled
  - ✗ bftpd chown overflow
  - ✓ bftpd format string vulnerability
  - ✓ Broker FTP files listing

Plugin information

bftpd format string vulnerability

This script is Copyright (C) 2000 Renaud Deraison

Category: Attack ☑ In use

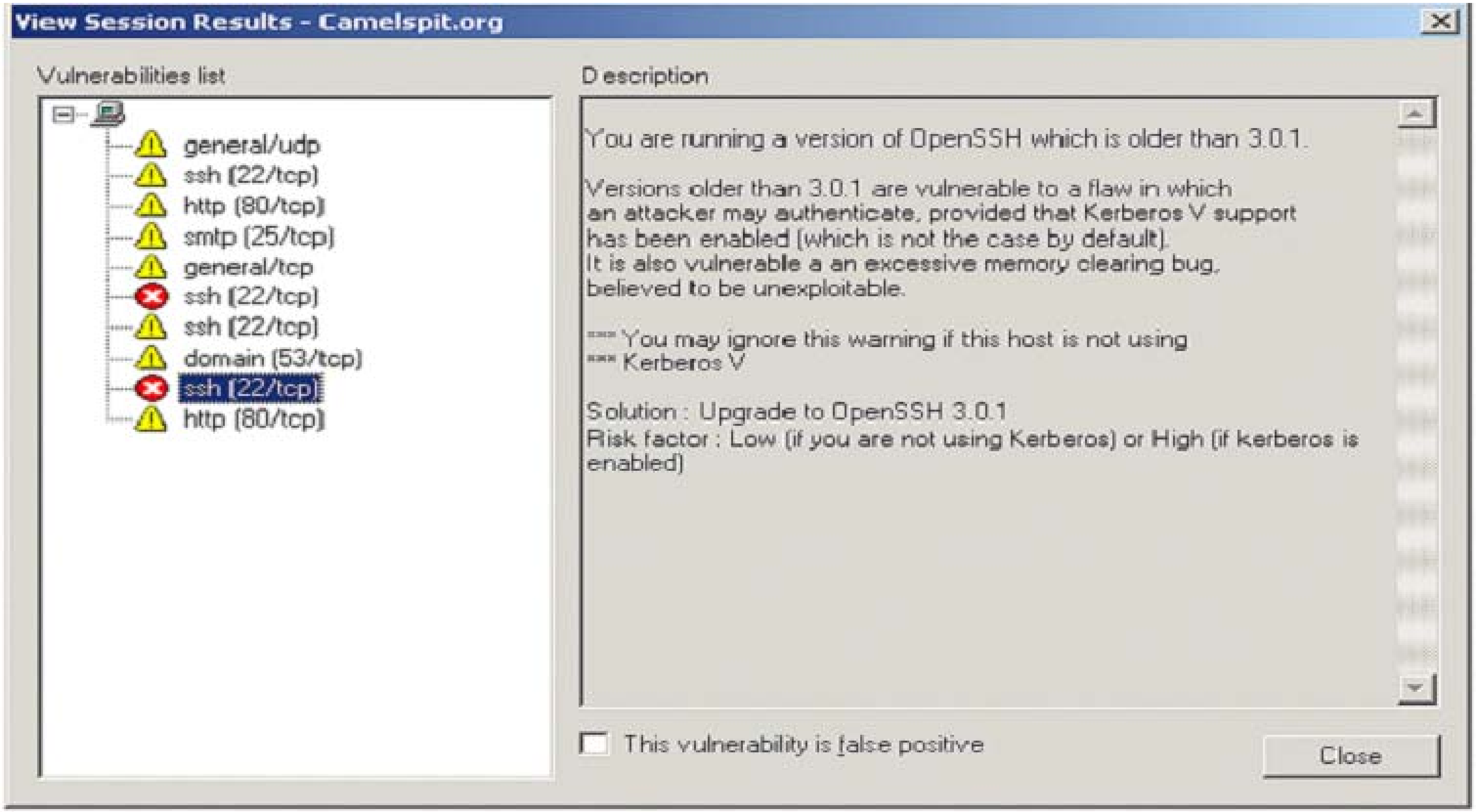
Description

The remote ftp server does not sanitize properly the output
it gets from the NLST command.

It may be possible for a remote attacker to gain root access
thanks to this bug if he can write in any directory served
by this ftp daemon.

Solution : Upgrade your bftpd server to version 1.0.13
Risk factor : Serious

Sample Nessus output



- ⚠ general/udp
- ⚠ ssh (22/tcp)
- ⚠ http (80/tcp)
- ⚠ smtp (25/tcp)
- ⚠ general/tcp
- ☒ ssh (22/tcp)
- ⚠ ssh (22/tcp)
- ⚠ domain (53/tcp)
- ☒ ssh (22/tcp)
- ⚠ http (80/tcp)

```
You are running a version of OpenSSH which is older than 3.0.1.

Versions older than 3.0.1 are vulnerable to a flaw in which
an attacker may authenticate, provided that Kerberos V support
has been enabled (which is not the case by default).
It is also vulnerable a an excessive memory clearing bug,
believed to be unexploitable.

*** You may ignore this warning if this host is not using
*** Kerberos V

Solution : Upgrade to OpenSSH 3.0.1
Risk factor : Low (if you are not using Kerberos) or High (if kerberos is
enabled)
```

□ This vulnerability is false positive

The Coroner's Toolkit (TCT)

- Digital forensics is useful for determining:
  - How a break-in occurred
  - A timeline of the incident – duration of exposure
  - What files and resources may have been exposed
  - Information regarding the attacker’s origin
- Two main strategies for digital forensics: system and network
- TCT provides three tools for UNIX system forensics:
  - grave-robber: data collection framework
  - unrm and lazarus: recover deleted files
  - mactime: checks file modify, access, and change times
- Works on almost all UNIX systems
- By Dan Farmer and Wietse Venema
- http://www.porcupine.org/forensics/tct.html

TCT usage

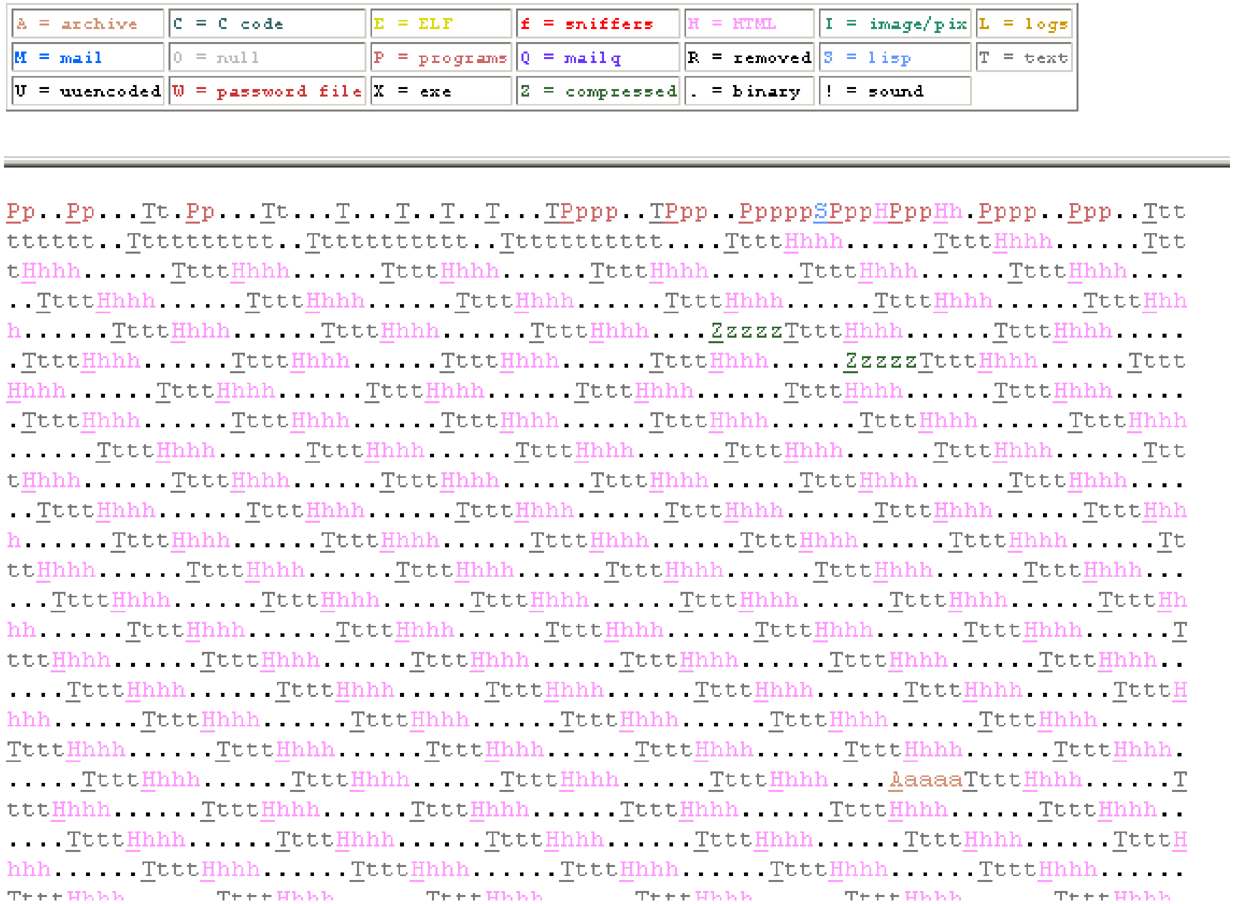
- grave-robber captures forensic information about a UNIX system according to the “Order of Volatility”
  - Roughly: memory, network state, running processes, disk, removable/fixed media

- Actually a bunch of tiny programs that do tasks like:
  - Gather network, host configuration, and user info
  - Suck in information from lsof, ps, and the memory of all processes
  - Save the executable of running programs which have been deleted from disk
  - Gather MAC information for files (see mactime)
  - Save important individual files
  - Make MD5 signatures of collected data

TCT usage

- unrm pulls unused blocks from a disk device
  - Outputs all unallocated space in one big stream
  - Only supports ext2fs on Linux and ufs on Solaris or BSD
- Lazarus sorts through this huge stream of data and identifies blocks of intelligible data
  - Output formatted as text or HTML
  - Requires at least double the disk space that you are trying to analyze
  - Assists in classifying blocks of data by file type

Sample Lazarus output



A = archive	C = C code	E = ELF	f = sniffers	H = HTML	I = image/pix	L = logs
M = mail	0 = null	P = programs	Q = mailq	R = removed	S = lisp	T = text
U = uuencoded	W = password file	X = exe	Z = compressed	. = binary	! = sound	

```
Pp..Pp...Tt.Pp...Tt...T...T..T..T...TPppp..TPpp..PppppSPppHPppHh.Pppp..Ppp..Ttt
tttttt..Tttttttttt..Ttttttttttt..Ttttttttttt....TtttHhhh......TtttHhhh......Ttt
tHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh....
..TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhh
h......TtttHhhh......TtttHhhh......TtttHhhh....ZzzzzTtttHhhh......TtttHhhh.....
.TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh.....ZzzzzTtttHhhh......Tttt
Hhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh.....
.TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh
......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......Ttt
tHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh....
..TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhh
h......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......Tt
ttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh...
...TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHh
hh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......T
tttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh..
....TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttH
hhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......
TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh.
.....TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh....AaaaaTtttHhhh......T
tttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh..
....TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttH
hhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......TtttHhhh......
TtttHhhh      TtttHhhh      TtttHhhh      TtttHhhh      TtttHhhh      TtttHhhh
```

mactime

- Modified, Accessed, Created time
- Easy to use – can be used independently of other TCT tools

  unix% mactime 2/9/2002
  - Returns all files with MAC changes since that date
- Can run against live filesystem or grave-robber data
- Will produce colored HTML output
  - With SUID/SGID files highlighted
- Find out what files are touched/run during a system boot
- Determining activity during a day or slice of time
- Finding out how much complexity (in terms of files) an application adds

Security: A Guide for Busy People

- Make sure you have a packet filtering firewall.
- Turn off unnecessary services on your systems.
- Apply security-related patches.
- Perform regular backups.
- Choose good passwords (and every account must have a password). Passwords sent across a network must be encrypted.
- Regularly monitor the health of your systems.

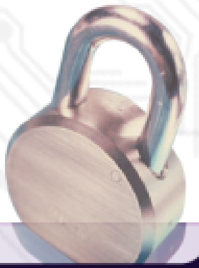
Deep thoughts…

… from special assistant to the President for cyberspace security Richard Clarke (in an interview with *Wired,* February, 2002):

"CEOs need to understand two things:

1. You have to have a multi-layered defense, and
2. You can't buy a security product and say you're done – you have to worry everyday."

"Most Fortune 500 companies spent .0025 percent of revenue on IT security -- less than on coffee. Now if you spent .0025 percent, you deserve to be hacked. And by the way, you will be."



Contacting us…

Ned McClain
ned@atrust.com
303-245-4505

Trent R. Hein
trent@atrust.com
303-245-2524