# Logging and auditing

Peter Honeyman
CITI
University of Michigan
Ann Arbor

# Introduction

- Log: generic or application-specific file that records noteworthy events
- Audit: process log files to monitor system behavior

# Summary

- Logging mechanisms used in UNIX
- External logging mechanisms

# Who am I?

- Ph.D. in database theory
- Three years Bell Labs
- Three years Professor @ Princeton
- Ten years Research Scientist @ CITI
- Research manager in middleware

# Who are you?

- Managers?
- Techies?
- Groupies?

# Agenda

- Topics to be covered:
  - UNIX logging facilities
  - The arms race
  - Defensive mechanisms
  - Prophylactic mechanisms
- Times allotted to each:
  - TBD

# Overview

- Log files and audit trails
- Essential for understanding and recovering from attacks
- Extremely vulnerable
- Log files themselves are subject to attack
- Alternative: external auditing

# Log files

- Application specific
- Generic
- Slight differences among UNIX versions
- Found in various places in UNIX, often in `/var/adm/`

# Application-specific logs

- last login
- aculog
- utmp and wtmp
- su log
- shell histories
- ftp xferlog
- httpd access_log

# last login

```
Last login: Tue May 27 15:50:47 on console
```

- Can flag suspicious behavior
- Overwritten at each login

# aculog

- Logs a record each time the “tip” command is used to place a phone call

# sulog

- Logs a record for each use of “su”
- `'su root' failed for honey on /dev/ttyp9`
- Sometimes logs to generic facility

# utmp and wtmp

- utmp is touched on each login/ logout event
  - Tells who is logged in
- wtmp is updated on each logout
  - Tells who has used the system

# Reading utmp with “who”

```
citi:; who
ted      ttyp0   May 27 09:19   (zeitgeist.citi.u)
ekl      ttyp1   May 27 17:20   (biloxi.citi.umic)
sarr     ttyp2   May 27 09:24   (sinshan.engin.um)
jej      ttyp3   May 27 09:27   (dopey.citi.umich)
honey    ttyp4   May 27 09:28   (vroom.citi.umich)
nigel    ttyp5   May 27 09:58   (heffalump.eecs.u)
honey    ttyp8   May 27 10:27   (doom.citi.umich.)
honey    ttyp9   May 27 18:35   (morelia.citi.umi)
admutil  ttypa   May 27 15:08   (excelsior.citi.u)
```

# Reading wtmp with “last”

```
citi:; last|sed 10q
honey     ttyp7    screwem.citi.umi Tue May 27 19:01 - 19:01  (00:00)
honey     ttyp9    morelia.citi.umi Tue May 27 18:35   still logged in
ekl       ttyp1    biloxi.citi.umic Tue May 27 17:20   still logged in
honey     console                   Tue May 27 15:50 - 16:11  (00:21)
admutil   ttypa    excelsior.citi.u Tue May 27 15:08   still logged in
johnpar   ttyp9    boyne.citi.umich Tue May 27 13:13 - 17:27  (04:14)
drh       ttyp7    dig.ifs.umich.ed Tue May 27 10:40 - 18:36  (07:56)
honey     ttyp8    doom.citi.umich. Tue May 27 10:27   still logged in
mts       ttyp7    206.252.4.86     Tue May 27 10:20 - 10:29  (00:08)
jbwl      ttyp8    raiden.us.itd.um Tue May 27 10:18 - 10:19  (00:00)
```

# Shell histories

- Many shells log commands
  - Per user
- Shell accounting

# xferlog

```
citi:; sed 10q xferlog
Tue Sep 14 16:23:56 1993 1 watson.citi.umich.edu 905 /u/lhuston/recl.c a _ o r l
huston ftp 0 *
Tue Sep 14 16:25:36 1993 26 watson.citi.umich.edu 1850397 /afs/umich.edu/group/i
td/citi/public/techreports/AUTO/citi-tr-92-3.ps b _ o a lhuston@citi.umich.edu f
tp 0 *
Tue Sep 14 16:26:41 1993 1 watson.citi.umich.edu 12314 /tmp/realp.ps a _ i r lhu
ston ftp 0 *
Tue Sep 14 17:18:52 1993 7 michael.centerline.com 21637 /afs/umich.edu/group/itd
/citi/public/techreports/ABSTRACTS b _ o a WWWuser@michael ftp 0 *
Tue Sep 14 17:21:09 1993 2 michael.centerline.com 7218 /afs/umich.edu/group/itd/
citi/public/techreports/INDEX b _ o a WWWuser@michael ftp 0 *
Tue Sep 14 17:29:58 1993 8 michael.centerline.com 21637 /afs/umich.edu/group/itd
/citi/public/techreports/ABSTRACTS b _ o a WWWuser@michael ftp 0 *
Tue Sep 14 17:31:58 1993 5 michael.centerline.com 28886 /afs/umich.edu/group/itd
/citi/public/techreports/PS.Z/citi-tr-93-4.ps.Z b _ o a WWWuser@michael ftp 0 *
Tue Sep 14 18:50:58 1993 1 watson.citi.umich.edu 321 /u/lhuston/foo3/1 a _ i r l
huston ftp 0 *
Tue Sep 14 18:50:59 1993 1 watson.citi.umich.edu 757 /u/lhuston/foo3/2 a _ i r l
huston ftp 0 *
Tue Sep 14 18:52:26 1993 1 watson.citi.umich.edu 321 /u/lhuston/foo3/1 a _ i r l
huston ftp 0 *
```

# access_log

- Web server logs
- Summarized with "getstats"

# Generic logs

- messages
- syslog
- tcp wrapper logs

# messages

- Copy of all console messages

# syslog

- syslogd service provided to kernel and applications
- Numerous classes of logs
  - facility.level
    - facility is name of subsystem sending message
    - level is severity of message

# syslog table configuration

- facility.level destination
- destination may be
  - file
  - device
  - remote host
  - user

# syslog facilities

- kern
- mail
- lpr
- daemon
- auth
- see syslog(3)

# syslog levels

- emergency
- alert
- critical
- warning
- notice
- info
- debug

# syslog config example

```
*.notice /var/log/notice
*.crit   /var/log/critical
kern.*   /dev/console
kern.err @logroll.citi.umich.edu
*.emerg  *
*.alert  root
*.alert  /var/log/alert
```

# tcp wrapper logs

```
in.telnetd : ALL : /usr/local/etc/tcpdlog
                             %d %h >> /var/adm/inetd.log
```

```
citi:; cat /usr/local/etc/tcpdlog
#!/bin/sh
# usage tpcdlog service name
# e.g., tpcdlog in.telnetd eecs.umich.edu
#
# this script exists solely to clean up
# hosts.allow and hosts.deny a little
#
/bin/echo $1 from $2 at "`/bin/date`"
```

# Log handling

- Always back up logs
- Search logs for suspicious behavior
  - E.g., logins from outside the domain
  - E.g., failed login attempts

# External logging

- syslog remote facility
- Promiscuous snooping on broadcast network
- Mitnick vs. Shimomura
- “The vault”

# Vault goals

- Rapid response to intrusion incident
- Continuous oversight of subnet traffic

# Approach

- Capture and process network packets
  - Initially all packets on 10 Mbps Ethernet
- Store long term
- Cryptographic sealing of packet contents

# Requirements

- Collector must sustain 10 Mbps packet input rate
- Archiver must sustain 270 KB/s to CD-R
  - ISO 9660 image created on magnetic disk
  - Image written to CD-R
  - Loss of data rate creates unusable CD

# Requirements, cont’d

- Commodity components
- Satisfy university, government, law enforcement, and individual needs

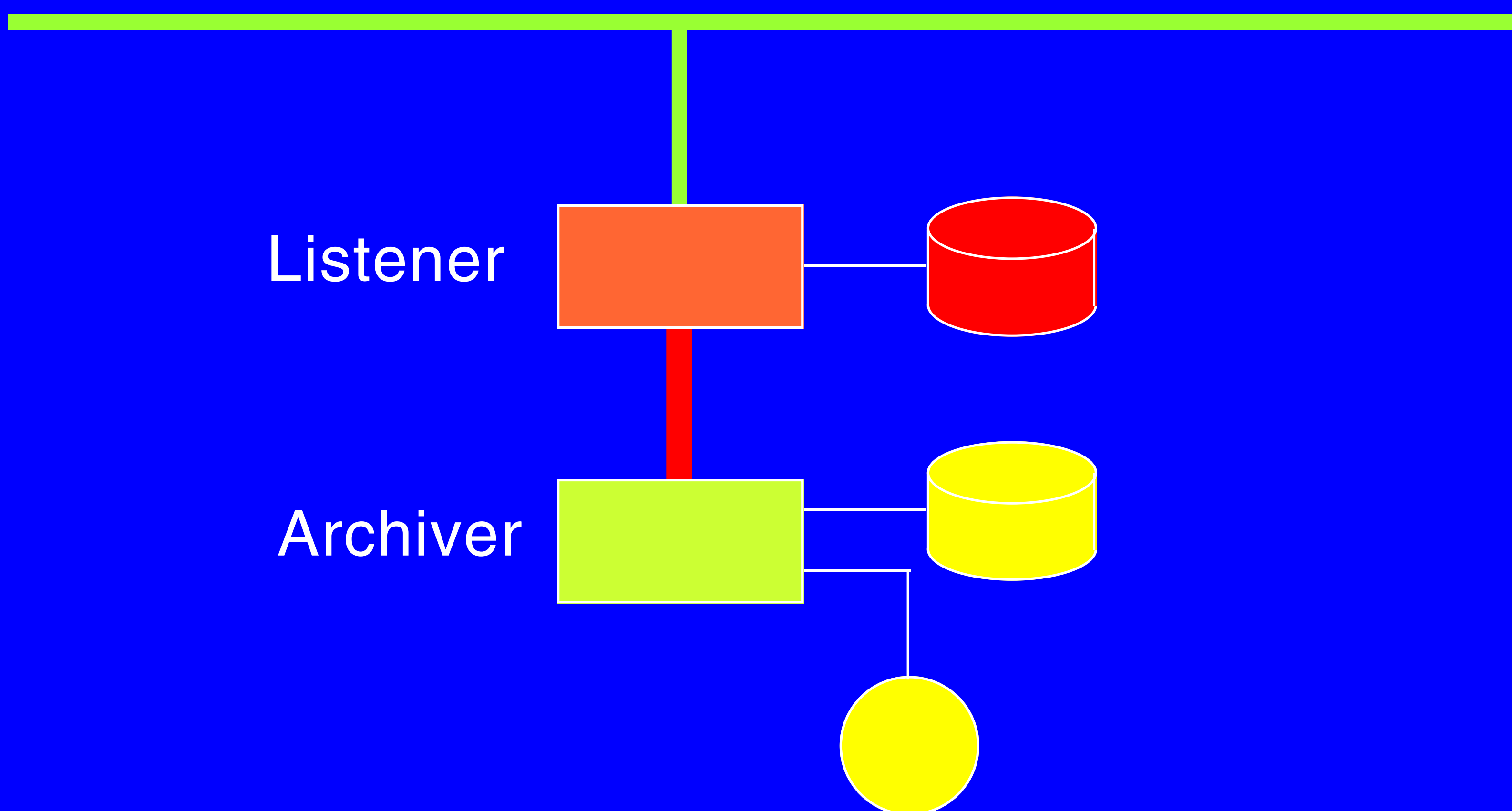
# Policy issues

- Privacy/First Amendment
- Search and Seizure/Fourth Amendment
- Discovery/Evidence
- Ownership/Copyright
- Student Information/FERPA
- Right to Know/FOIA
- Carrier-Transport/ECPA
- Human Subject Guidelines
- Pending legislation and legislative trends

# Policy issues

- Is storing encrypted data equivalent to storing unencrypted data?
  - We don’t know!
  - Little direct precedent
  - Currently under study.
- We are proceeding carefully

# Architecture



# Architecture

- Dual commodity Pentiums
- Listener accumulates packets from network onto staging disk
  - Continuous operation
- Archiver stages and transfers to archival storage
  - Batch operation

# Vault hardware

- Collector
  - 133 MHz Pentium
  - 128 MB RAM
  - IDE disks
- Archiver
  - 133 MHz Pentium
  - SCSI disks
  - Yamaha CD-R
- Private 100 Mbps network

# Collector software

- OpenBSD
  - Network, MFS
- User-level processes to capture packets
  - `tcpdump` format
- Scripts for post-processing

# Collector software, cont'd

- BPF delivers raw packets
- Packets accumulated in MFS files
- Post-processing
  - Host/port mapping
  - Cryptographic sealing
  - Transfer to archiver

# Archiver software

- Linux
  - PCI, CD-R
- Scripts for post-processing
  - Create ISO filesystem image
  - Write to CD-ROM

# Cryptographic requirements

- No direct identification of source and destination packet addresses
- Per-volume keying
- Per-conversation payload keying

# Cryptographic organization

- Source/destination addresses obscured via translation table
- Payloads encrypted with payload key
- Payload key derived from volume payload key and packet header

# Cryptographic organization

translation table symmetric key

Regents' public key

volume payload symmetric key

Regents' public key

translation table

translation table key

translated header | packet payload

payload key

■ ■ ■

# Cryptographic organization

- Per-volume key: Kv
- Per-conversation payload key, Kc
- Kp = DES(Kc | TSA | TDA, constant)
  - TSA: translated source address
  - TDA: translated destination address
- |Kp| = 192

  2 x 64 bits for DESX whitening

  64 bit DES key

# Other issues

- Storage policy
  - How many packets could the packet vault drop if the packet vault had to drop packets?
  - Investigating packet triage methods
    - drop “known harmless” conversations
    - you had better be sure!
- Packaging
  - “Single box” solution attractive
  - Investigating ways to shrink prototype

# Vault status

- Collector running to MFS
- Archiver writing CDs
  - Not archiving any data yet!

# Vault work in progress

- Improving performance on private net
- Studying existing tools for intrusion detection
- Studying policy issues, report being prepared
- Studying packaging and storage policy issues

# Summary

- UNIX has myriad logging and auditing tools
- Probably too many
- Unified through syslog to a degree
- Logs are vulnerable
- External logging can be valuable

# More information

- Practical UNIX & Internet Security (Second Edition), Simson Garfinkel and Gene Spafford, O'Reilly & Assoc., Inc., Sebastopol, 1996.
- UNIX System Administration Handbook (Second Edition), Evi Nemeth, Garth Snyder, Scott Seebass and Trent R. Hein, Prentice-Hall, Englewood Cliffs, 1995.

# How was it?

- Too long? short? thin? heavy?

# Any questions?

**`http://citi.umich.edu/`**