Logging

Shadow Hawk

Shadow Hawk Busted Again

As many of you know, Shadow Hawk (a/k/a Shadow Hawk 1) had his home searched by agents of the FBI. . .

When he was tagged by the feds, he had been downloading software (in the form of C sources) from various AT&T systems. According to reports, these included the Bell Labs installations at Naperville, Illinois and Murray Hill, New Jersey.

—Phrack Issue 16, File 11, November 1987

How was Shadow Hawk Detected?

- He had broken into some Bell Labs machines
- He tried to use **`uucp`** — a dial-up file transfer/email system that came with Unix — to grab **`/etc/passwd`** files from other machines
- Uucp logged all file transfer requests
- Several people at Murray Hill had automated jobs that scanned the log files for anything suspicious
- (What did we look for?)

What Did We Look For?

- Requests denied for security reasons
- Anything mentioning **`/etc/`**
- Attempts to execute commands for other than email and netnews

Stalking the Wily Hacker

- An accounting file didn't balance — a username had been added without the proper bookkeeping entries
- Cliff Stoll noticed and tried to figure out what was going on
- Ultimately, it led to a KGB-controlled operation aimed at military secrets. . .
- (It also led to a CACM paper, a book, and a TV show for Cliff. . .)

What was the Common Thread?

- Log files of various sorts
- “Extraneous” information
- Log files can prevent problems, help you figure out how the system was penetrated, what was affected, and — if you’re lucky and persistent — who did it

Where Do Log Files Come From?

- Many different system components can produce logs
- Often, these aren’t enabled by default
- Should they be?

Web Logs

- Here's an entry from one of my web server logfiles:

```
aaa.bbb.ccc.ddd - - [20/Nov/2005:22:32:15 -0500] "GET
/1e/chap02.pdf HTTP/1.1" 200 215350
"http://www.wilyhacker.com/1e/" "Mozilla/4.0 (compatible;
MSIE 6.0; Windows NT 5.1; SV1; Tablet PC 1.7;
.NET CLR 1.0.3705; .NET CLR 1.1.4322; FDM)"
```

- Let's look at it piece by piece

Web Logs

- IP address and timestamp
- Hmm — what time zone?
- The actual command transmitted, plus the response
- Lots of information about the client
- Hmm — web clients tell a lot about themselves. . .

Detecting Problems Via Logfiles

The ”Code Red” worm activity can be identified on a machine by the presence of the following string in a web server log files:

```
/default.ida?NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNN%u9090%u6858%ucbd3%u7801%u9090%u6
u7801%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u00c3%u0003%
b%u53ff%u0078%u0000%u00=a
```

From `http://www.cert.org/advisories/CA-2001-19.html`

An Attempted Intrusion?

```
[Sun Nov 20 23:17:18 2005] [error] [client www.xxx.yyy.zzz]
File does not exist: /usr/pkg/share/httpd/htdocs/xmlrpc
[Sun Nov 20 23:17:28 2005] [error] [client www.xxx.yyy.zzz]
File does not exist: /usr/pkg/share/httpd/htdocs/phpgroupwar
```

(There were many more attempts from that IP address.) Both of these represent services with known security holes

It Still Happens

- Logs from the last few days show different attempts to find web holes
- Probes came from Germany, Indonesia, Japan, UK, Jamaica, and the US
- Targets probed include several PHP scripts, some mysql content, and what appear to be “shopping carts”

```
[error] [client aaa.bbb.ccc.ddd]
File does not exist:
/usr/local/www/fw-insec2/phpmyadmin
```

Problems with Log Files

- How did I spot those probes?
- Manual search through error_log
- Not very scalable. . .

Log File Scanners

- Need to automate scans
- Pick out “interesting” events
- Hmm — what’s interesting?

Log Files and Intrusion Detection

- Analyzing log files like that is a form of intrusion detection
- Can look for specific signatures, such as examples above
- Or — can look for anomalous patterns, such as too many misses or too-long URLs

Correlating Log Files

- Sometimes, the interesting information is spread among several log files
- Need accurate timestamps for correlation between machines
- Timestamps should generally be in UTC, rather than the local timezone

Types of Correlation

- Intra-machine — different forms of logfile
- Intra-site
- Inter-site
- Watch out for privacy issues!

Types of Logs

- Application-specific: web, email, etc.
- Generic OS logs
- Network element logs

Sliced Another Way

- Routine processing
- Error messages
- Authentication events and/or errors
- Access control events and/or errors

Processing Logs

- Primary rule: retain raw data as long as possible
- Be suspicious — log files often contain enemy-supplied data
- Be *especially* careful if you use a web browser to look at log file data: loggable data has included things like Javascript-based pop-ups
- Crunch log down to manageable size
- Pick out interesting items

Network Logs

- Can you do full-traffic logging?
- A DS-3 line is 45M bits/sec, or 5.6M bytes/sec.
- Assume you run the link at 20% capacity == 1.125M bytes/sec
- On a 250G drive, you can store 222,222 seconds, or 61 hours
- That looks feasible
- But — can you extract any meaning from that much data?

Suppose Your Site is Penetrated

- A full-traffic log can show you what was done
- Well, maybe it can't — did the bad guys use encryption?
- Such logs may or may not be helpful
- They're a *tremendous* privacy risk

Network Connection Logs

- Many routers can produce “flow logs”
- A flow is (roughly speaking) a TCP connection
- This sort of traffic analysis can reveal many types of attacks

Limits of Full-Traffic Logs

- Capturing every packet is hard
- If you want, say, URLs accessed from your site, it's easier to use a web proxy log
- Must prevent direct web access — block ports 80 and 443

Security and Logfiles

- What sorts of security do logfiles need?
- Confidentiality? Integrity? Availability?
- All of them!

Confidentiality

- Logfiles can contain sensitive data
- Again, watch out for personal privacy
- Besides, you don’t want to tip off the attacker about what you know

Integrity

- Make sure the enemy can’t tamper with your logs
- Prime target for many hackers!
- Absence of log file entries is not evidence that nothing happened

Availability

- Attack: fill up log area with innocent garbage
- When the log file is full, launch the real attack
- Some systems will lock up if the log files are full — easy denial of service attack against the whole system!
- Which is better — no logs, or no processing?

Storing Log Files

- Local machine — easy, but vulnerable to attackers
- Use a log server
- Note: log server must authenticate log requests

Secure Logs

- Scope: assume no high-bandwidth connection to log server
- Protocols exist for secure logging
- Limit attacker’s ability to read or corrupt log files

Creating Good Logfiles

- What should be in a log file?
- How is functionality divided?
- Remember — logfiles are parts of *systems*

What’s in a Logfile?

- Timestamp — when did it happen?
- What happened? To what resource or resources?
- Who did it?
- What “session” or network connection did it come from?

Logfiles and "Sessions"

- The accounting log on my desktop says:

```
sh          -S      root    __          0.61 secs
```

  (timestamp omitted)

- That is — **`root`** ran a shell, which in turn performed some operation that only root can do
- Was it me, in a root window? A system daemon? An attacker?
- If it was an attacker, was there a network connection involved? From what IP address?
- In this case, the **`--`** is supposed to be the pseudo-tty, so it's not from a window — but the process can set it to null, so it's not trustworthy.

Things to Log

- Desired log level varies; tradeoff between comprehensiveness and space or performance
- Always log security events: permission denied, `su`, use of privileges, authentication failures, etc.
- Always log session start/end
- Log accesses? To what? By which users?

What if You Can't Log Everything?

- Application processes may not know enough to create proper logs
- Example: a Unix application running in remote login window doesn't, in general, know the IP address of the client
- Solution: provide linkage information
- The remote login program logs the connection information and tty name; the application logs the tty name.
- Correlation done by the log file analyzer

Logfiles and Auditing

- Logfiles can act as a deterrent
- If people know the logfiles show what they did, maybe they won't do bad things
- Example: the original purpose of a cash register was to produce the manager's copy of all receipts
- How do we do such audits?

Follow Sample Transactions

- Find all log records pertaining to a transaction
- Note: implies ability to correlate
- May require extra tag fields in log messages: tag actions with “session” information
- Manually check each step
- Supplement with phone calls, phone logs, etc

The Sad Reality

- Lots of folks have very good logs
- Few of these logs are ever examined
- Many warnings are missed

The Wikileaks Data

“Weak servers, weak logging, weak physical security, weak counterintelligence, inattentive signal analysis . . . a perfect storm.’

Pfc. Bradley Manning, prime suspect in the leaks

What Went Wrong?

- Caveat: we don't really know yet; I'm speculating based on that quote
- He brought in CDs — regulations prohibit "personally owned computers and associated media" in secure facilities
- No one objected
- No one checked the logs for how much he was downloading
- Possibly bad personnel security — should he have received a clearance? Were there changes in his behavior that should have been noticed?