Logging

Portions courtesy Ellen Liu
Outline

• Introduction
• Finding log files
• Syslog: the system event logger
• Linux “logrotate” tool
• Condensing log files to useful information
• Logging policies
Who and Why

• System daemons, kernel, and utilities produce log data onto disks

• Most log data has limited useful life, needs to be summarized, compressed, archived, then removed

• Access and audit data are needed by government regulations and company policies

• Logs also reveal configuration problems
Logs

• A log event is captured as a single line of text
  – time and date, type and severity of the event, etc.,
    often separated by spaces, tabs, or punctuation

• Logs are plaintext files, can be processed by shell commands and shell scripts

• There are also log management tools that rotate, compress, and monitor log files daily or weekly
### syslog messages

![Syslog Watcher](image)

<table>
<thead>
<tr>
<th>Received</th>
<th>Source IP</th>
<th>Facility</th>
<th>Severity</th>
<th>Tag</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Info</td>
<td>kernel</td>
<td>Initializing IPsec netlink</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Debug</td>
<td>syslog</td>
<td>tftp server</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Info</td>
<td>kernel</td>
<td>TCP: Hash tables clean</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Debug</td>
<td>syslog</td>
<td>bftp server</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Info</td>
<td>kernel</td>
<td>IP: routing cache has changed</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Debug</td>
<td>kernel</td>
<td>PCI: Setting latency time</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Info</td>
<td>kernel</td>
<td>NET: Registered protocol</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Debug</td>
<td>syslog</td>
<td>snmp - s time.windows</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Info</td>
<td>kernel</td>
<td>NET: Registered protocol</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Info</td>
<td>kernel</td>
<td>Memory: 13948k/160m</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Notice</td>
<td>kernel</td>
<td>Etables v2.0 registered</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Critical</td>
<td>kernel</td>
<td>ADSL G.992 channel registered</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Critical</td>
<td>kernel</td>
<td>ADSL G.992 started</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Critical</td>
<td>kernel</td>
<td>ADSL G.994 training</td>
</tr>
<tr>
<td>03/20/2008 19:25:03</td>
<td>192.168.1.1</td>
<td>user-level</td>
<td>Debug</td>
<td>syslog</td>
<td>kill -9 521</td>
</tr>
</tbody>
</table>

**Interpretation**

This message indicates that the system was reset from a specified session. The message contains the number if the request is from a console session or an IP address if the request is from a Telnet session or SNMP.
IT Standards & Industry Regulations

• COBIT
  – A set of best practices framework for information technology (IT) management

• ISO 27002
  – Provides best practice recommendations on information security management

• Require sites to maintain a centralized, hardened, enterprise-wide repository for logs, with NTP time stamps and a strict retention schedule
## Finding Log Files

- **Names:** `maillog`, `ftp.log`, `lpderrs`, `console_log`, ...
- For Linux, by default most are found in `/var/log`, `/var/adm`
- Some common log files:

<table>
<thead>
<tr>
<th>File</th>
<th>Program</th>
<th>Where</th>
<th>Freq</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>acpid</td>
<td>acpid</td>
<td>F</td>
<td>64K</td>
<td>Power-related events</td>
</tr>
<tr>
<td>boot.log</td>
<td>rc scripts</td>
<td>F</td>
<td>M</td>
<td>Output of startup scripts</td>
</tr>
<tr>
<td>cron</td>
<td>cron</td>
<td>S</td>
<td>W</td>
<td>cron executions and errors</td>
</tr>
<tr>
<td>faillog</td>
<td>login</td>
<td>H</td>
<td>W</td>
<td>Unsuccessful login attempts</td>
</tr>
<tr>
<td>httpd/*</td>
<td>httpd</td>
<td>F</td>
<td>D</td>
<td>Apache logs</td>
</tr>
<tr>
<td>yum.log</td>
<td>yum</td>
<td>F</td>
<td>M</td>
<td>Package management log</td>
</tr>
</tbody>
</table>

**Where (filename source)** - S: syslog, H: hardwired, F: configuration file

**Freq (freq. of cleanup)** - D: Daily, W: Weekly, M: Monthly, Size-based
Log Permissions and Syslog

- Log files are normally owned by root
  - Occasionally by less privileged httpd, mysqld, etc.
- Sensitive logs need tight permissions. Others can be set to world-readable

- **Syslog**: an integrated system to concentrate logs
  - On UNIX/Linux systems
  - syslogd daemon
  - configuration file: /etc/syslog.conf
Log Files Management

- Log files can grow large quickly, especially with busy services, e.g., email, web, and DNS servers
- They may fill up the disk, degrading system performance
- Normally one uses a separate partition for busiest log files
  - On Linux, it is a good choice to have /var or /var/log occupy a separate partition on the disk
Logs *not* to manage

Logs are text files to which lines are written as interesting events occur. But some logs are different

- **wtmp**: records of users’ logins / logouts, system reboot and shutting down. Binary format. Use “`last`” command to decode
- **lastlog**: similar to above. Only records last login for each user.
- **utmp**: keeps a record of each user that is currently logged in. Maybe inaccurate if a shell is killed inappropriately

- You may read the man pages of each for more information
Vendor specific log file locations

• Vendors may have their log files all over the disk. Check daemons’ config files and syslog configuration files to find them.

Linux “logrotate” tool

• Linux logs are usually clearly named and consistently stored in /var/log

• Linux distributions also include a log management tool “logrotate”. It rotates, truncates, manages logs

• New software can add a config file to /etc/logrotate.d directory, to set up a management strategy for their logs, as part of their installation procedure.
Syslog: the system event logger

• Liberate programmers from tedious mechanics of writing log files

• Put administrators in control of logging rather than letting every program make up its own logging policy, such as what information to keep and where it is stored

• Let you sort messages by importance and source, also route messages to a variety of destinations: log files, users’ terminals, other machines’ syslogd
  – The last one can centralize logging on a network
Syslog Architecture

Three parts:

- **syslogd**: the logging daemon, its config file `/etc/syslog.conf`
- **openlog** et al., library routines that submit msgs to syslogd
- **logger**: a user command that submits log entries from the shell

- Syslogd is started at boot time and runs continuously
- Programs write log entries using the library calls
- One can submit an entry using command “logger”
  - `logger -p local7.warning “a warning message”`
Configuring Syslogd

- `/etc/syslog.conf` file, called `/etc/rsyslog.conf` in CentOS 6
  - It is a text file with simple format
  - ‘#’ starts comment lines, which are ignored
  - The basic format: `Selector<tab>action`
  - Can have one or more tabs
  - E.g., “`mail.info<tab>/var/log/maillog`”
    - causes messages from the email system to be saved in `/var/log/maillog` file
Syslog Selectors

• Selectors identify the program sending the log message, and the message’s severity level,

• Selectors syntax *facility.level*
  
  – Both facility names and severity levels must be from a short list of defined values
  
  – Facilities are defined for the kernel, for common utilities, for locally written program, and for others named “user”
  
  – Also use special keywords: * means all, none means nothing, comma to separate multiple facilities, ; to separate multiple selectors
  
  – Facility names: auth, cron, daemon, ftp, kern, local0-7, lpr, mail, news, ...
  
  Severity levels (descending severity): emerg, alert, crit, err, warning, notice, info, debug
Syslog Actions

Syslog produces time stamp messages.

- **Filename**: appends the message to a file on the local machine
- **@hostname**: forwards the message to the syslogd on hostname
- **| fifoname**: writes the message to the named pipe
- **User1, user2**: write the message to the screens of users if they are logged in
- *****: write the message to all users currently logged in
- **-** means no filesystem syncing after writing each log entry, this helps with performance, may miss some log upon crash
Linux “logrotate” tool

• “logrotate” rotates, truncates, manages logs
• The logrotate config file is /etc/logrotate.conf
• logrotate is normally run out of cron once a day
• Example logrotate options:
  – Compress all noncurrent versions of the log files
  – Rotate log files daily, weekly, or monthly
  – Emails error notification to a specified email address
  – Specify script to run after log is rotated
  – Include $n$ versions of log
Condensing Logs

• Syslog great for sorting and routing log messages, at the end a bunch of log files are created

• Tools can scan log entries, match against a database of patterns of log messages, and find the important messages

• Example log postprocessor tools: swatch, logcheck, Splunk, SEC (Simple Event Correlator) etc.
  – swatch: ‘simple watchdog’ to monitor log files from syslog and others
Important Checking

Always check for important items, including:

• Most security-related messages need prompt review
  – Failed login, su, sudo attempts. Someone may forget passwords, but also want to prevent potential break-ins

• Messages about disks that have filled up
  – Full disks often bring useful work to a standstill

• Events that repeated many times
Logging Policies

• Logs are critical to security incident handling

• Ask the following when designing logging policies
  – How many systems and apps will be included?
  – What type of storage infrastructure is available?
  – How long must logs be retained?
  – What types of events are important?

• Record the following:
  – user name or ID, event success or failure, source address, data and time, sensitive data changed, event details
Log Centralization

• If site has >20 servers, consider centralized log collection and analysis. Reasons:
  – Simplified storage, automated analysis and alerting, improves attack visibility

• Storage strategy:
  – E.g., 30 days on RAID array, 1 year on SAN, and 3 years on tape archives

• Access only to high-level sysadmins, access to central logs should be logged

• Small sites: rotate logs, regular archives