Log Analysis

Monitoring and Log Management

Analysis Goals

- Logs typically contain too many records to review directly
- Analysis should make the data ready for human consumption, highlighting the interesting events and drawing attention to unusual records
- Useful summaries need to have data from as many sources as possible to find correlations or begin to research event initiation for non-trivial events
- Critical events will ideally invoke real-time responses from support systems and staff

Obstacles

- The syslog standard is a message transport mechanism, not a logging mechanism - programs make their own weird and wonderful logging choices and produce free-form output, messages are just a 32-bit number followed by a block of text
- Different versions of the same program often produce different message formats
- The same version of a program may produce different messages on different distros
- Successful attacks often leave little or nothing in the logs, depending on the attack

Challenges

- Log messages can exceed several GB per day on a departmental or corporate server, and be many GB on web-facing public servers
- Most of the data is useful only in support of trend analysis with respect to usage, capacity planning, etc.
- Finding interesting entries is a needle in a haystack problem, log messages are intended to be human readable which makes machine parsing more difficult
- Message handling on busy systems can be a significant load, some places disable it or seriously limit it especially on costly infrastructure devices
- Where do you look to see how logging is broken?

Message Value

- Message detail, quality, and integrity varies depending on the log data source and verbosity
  - where do you accept messages from?
  - are your clocks synchronized reliably?
  - do you store ip address of the source, or hostnames?
  - did the intruder send false messages to the log to throw you off (bad timestamps etc.)?
  - did the intruder go to the trouble of trying to crash your log daemon?
  - did the intruder have the ability to alter or remove messages, or even the whole logs?
- Message parsing is the holy grail of log analysis
- Message contents should be viewed with skepticism on a compromised host

Defining Interesting

- Interesting can be defined as being anything you don't typically see in the logs
- The first step is to pare down the incoming log data to only be the types of events you care about, so you can more readily see when you get something out of the ordinary
- Presuming your capacity planning is adequate, you are usually looking for resource consumption or exhaustion, system or service restarts/reboots, unplanned software configuration changes, access failures or account changes
- If you use file sharing and run native applications on desktops, desktop logs become useful, otherwise they tend to be voluminous data of questionable value

Interesting Router Events

- Access events, such as entering enable mode on a cisco
- Configuration changes
- Interface enabling/disabling or state changes
- Firmware changes
- Failed login attempts
- Unreachable routes

Interesting Firewall Events

- Access events, particularly from external networks
- Administrative logins, configuration changes
- Traffic flow data
- Filtered traffic data
- fwanalog and fwlogwatch are examples of tools to get log summaries, configuration improvements or customization of reporting may be needed for new attacks

Interesting System Events

- Reboots, panics, shutdowns to single-user mode
- Hardware resource exhaustion e.g. full filesystems, memory shortage (RAM and VM), i/o bandwidth overrun, processor adequacy
- Service crashes and restarts
- Unplanned configuration changes, unexpected ports listening for connections
- Access failures
- Unusual access volume or source
- logwatch is a good starting point for this type of information

Interesting Database Events

- Access events, particularly administrative access
- Interactive access as opposed to normal backend service provision
- Configuration changes and queries, especially access control
- Schema queries and invalid queries, probing for db structures
- Changes to scripts and software that access the db

Interesting Web Server Events

- Permissions issues and access failures
- Errors accessing backend services such as databases
- Unusual resource requests (e.g. php config files, my.cnf, dot files, etc.)
- Web service process death or injury
- Malicious URL signatures
- Unexpected configuration (config files as well as modules) or content changes including via symlinks
- Traffic volume spikes
- analog is an example of a tool for summarizing web activity

Strange Creatures

- Non-printing characters or nulls in a log message can be indicative of buffer overflows or overflow attempts
- Pay attention to hostnames in logs, they can be forgeries or designed to cause log message truncation
- Process names (tags) can be helpful in finding successful intrusions, watch for unusual commands and program names

Identifying Configuration Changes

- There is software to help with this (e.g. Tripwire, Lynis), most can only tell you if you have configured your machine, not if that has been altered from what you intentionally modified
- Scripts run by cron are your best friend in a well-managed environment, tailored to your needs (e.g. use hashes and compare the current hashes to the values from the last time you intentionally modified anything)
- The key is to have a well-managed environment, which means not having unnecessary software installed
- It also means actively configuring any installed software
- When you know what you have configured, it is much easier to use script-based checking if any of it has changed without your involvement

Log Maintenance

- logrotate is used to age the log files
- It is run out of cron
- The oldest log files are simply discarded
- If your business policy includes keeping historical data, you should modify the logrotate job to archive old logs instead of removing them
- You may also create your own scripts to archive old logs, and perhaps replicate them using version control to prevent loss/damage in old archives

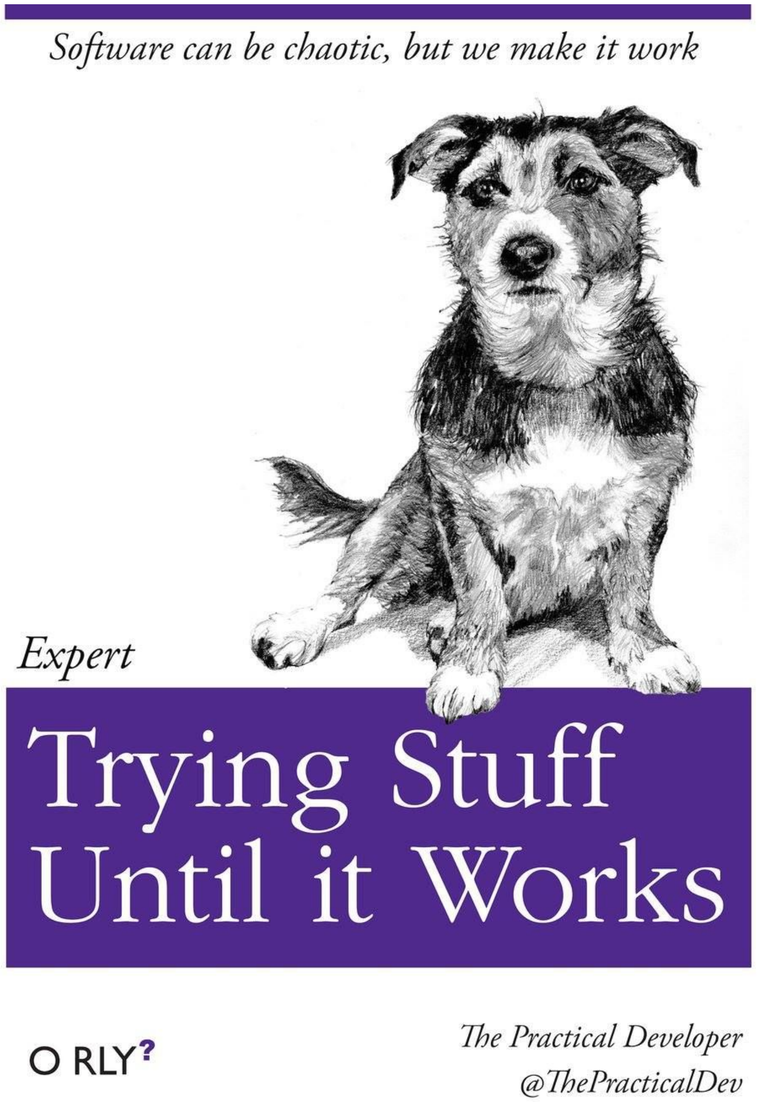
Logwatch

- Logwatch is a tool that can perform basic summary reporting of text log files
- It is a configurable tool which can be run by cron
- It provides an overview of important events during the log report period
- If you require more detail, you can use grep, sed, awk, sort, uniq to pull and summarize the actual log entries from text log files

Loganalyzer

- Loganalyzer is a mature web application for viewing and searching log files stored in any of a number of locations, remote or local
- It can run under apache2 and other platforms with a little tweaking
- It is free to download and run, but is not very modern in its interface

http://yallalabs.com/linux/how-to-setup-loganalyzer-with-rsyslog-on-ubuntu-16-04-lts-ubuntu-18-04-lts/



Log Analysis Lab

- logwatch, fwlogwatch, analog, fwanalog
- email of reports
- loganalyzer webapp