Network and Log Monitoring

Linux Systems Security

Learning Objectives

- Understand the importance of network and log monitoring
- Learn about tools available in Ubuntu 24.04 for monitoring
- Set up a basic monitoring environment
- Conduct practical lab exercises

Why Monitor Networks and Logs?

- **Prevent Security Threats**: Identify and mitigate vulnerabilities
- **Analyze System Behavior**: Understand resource usage and network traffic
- **Compliance Requirements**: Meet organizational and regulatory standards
- **Debugging and Troubleshooting**: Detect and resolve issues effectively

Common Tools

Network Monitoring

- iftop
  - top-style CLI monitor for network traffic volumes by IP/hostname
- nethogs
  - top-style CLI monitor for network traffic by process
- netstat/ss
  - CLI tool for examining current network end point states
- nmap
  - diagnostic network traffic generator
- tcpdump
  - CLI packet capture and trivial analysis tool
- Wireshark
  - GUI packet capture and analysis tool

Log Monitoring

- journalctl
  - CLI systemd log viewer
- logrotate
  - Automated log aging tool
- logwatch
  - CLI log analysis and summary reporting tool
- rsyslog
  - Tool for processing logs and protocol for transferring them

Setting Up For Monitoring

Software Installation

- journalctl, rsyslog, and logrotate are typically already present and have functional default configurations in most Linux systems
- iftop, nethogs, logwatch, nmap, tcpdump, wireshark are usually manually added as desired
- wireshark requires a GUI, tshark is a CLI version of wireshark for remote use
- network and log monitoring tools are often integrated into web apps such as webmin and cockpit

rsyslog Configuration

- /etc/rsyslog.conf is the main config file and sets global parameters
- /etc/rsyslog.d/*.conf are additional service specific configuration files (e.g. ufw, postfix)
- There are man pages and rsyslog.com has many sample configs
- The config file language is not intuitive for most people

Systemd Log Viewing

- The journalctl command is intended to make logs written by systemd-initiated programs easier to find and understand
- Typically run with the -x option, it not only display log entries but tries to look up descriptive text for messages it recognizes to make the entries more useful
- The -e option tells it to jump to the end of the logs so that the most recent log entries are shown by default as entries are displayed in ascending time order
- It has many other options to filter or format the logs entries
- It is only useful for viewing logs on the local machine written by things that are started by systemd

Manual Log Viewing

- Logs are by default kept in plain text file in the /var/log directory
- You can use typical text manipulation tools to view them and find things in them
- It is common to use grep to find specific error message text, process id numbers, port numbers, service names, process names, and user names
- If you use grep for this, it is often helpful to include context (-A, -B, -C) for the log entries found (the lines right before and after)
- It is also often helpful to use specific word searches instead of just pattern matches (-w)
- Remember there can be multiple log files written into for any particular event
- You may wish to search archived log files as well as the current log file

Log Management

- Logs by their nature are data repositories that continuously grow
- Without bounds, they would grow to consume the entire storage space they reside in
- Some programs can use a circular log buffer, such as the kernel message buffer (view with dmesg)
- For standard system log files, the logrotate program is designed to move older messages along a virtual conveyor belt on their way to the great bit bucket in the sky
- logrotate use a configuration file to specify log files to be automatically aged, /etc/logrotate.conf
  - it runs from cron, periodically doing its work, but can be run interactively on the command line
  - can use additional config files to split out configs for different logs, and compresses older logs
  - can handle many logs types, schedules, and has lots of other parameters to control how aging is done

Logwatch

- Logwatch parses log files and extracts summary reports, based on config files in /etc/logwatch/conf/ and /usr/share/logwatch
- The default is text format and output on the terminal
- Commonly implemented as automated daily reports sending email summaries
- Shows software changes, hardware changes, user changes, sudo usage, services access, kernel errors, storage usage, and whatever else you add config files for
- Lots of default config files are in /usr/share/logwatch/default.conf

Web-based Admin Tools

- There are many web-based applications that can run on a Linux system that include log viewing and management capabilities
- cockpit is a Redhat webapp that listens on port 9090 and provides a way to manage a Linux systems derived from Redhat or Debian
- It allows log viewing with simple filtering and date selection, but is not suitable for anything other than taking a look for something fairly specific
- webmin is an OSS webapp that listens on port 10000 and runs on most Linux distros
- It allows full and highly granular control over logging as well as the ability to view, filter, sort, and search logs

Packet Capture tools

- tcpdump and tshark provide the ability to capture packets and either live display them in multiple formats or save them to pcap files for later analysis
- wireshark does the same thing but with a GUI and adds point and click analysis and reporting tools for captured packets
- they all can use filters to be selective in what is captured and all require root access to properly capture things
- nmap can be used to do reconnaissance, test service responses or firewall effectiveness, or generate traffic

Network End Point State Examination

- ss and the older netstat provide multiple views of network end points on a system
- both listening and active end points can be revealed along with statistics for them
- combined with firewall state, it is possible to see what services a system is providing to remote clients
- ss can generate reports, show detailed internal connection states and flow states, and kill network connections