LSG

UNIT 6 - Replay

Last Week ① Repos ② Patching
This Week ① Storing logs - How do we set them, where do we set
② Parsing logs - Look through for patterns

A log is an immutable record of what occured or what attempted
We use logs to formulate response to incidents & etc.
↳ log should say (I) when did it happen (II) Severity (III) what process ran
Storage of logs (A) Centralized locations (B) Rotation for space & compression
(C) How do we get logs off the servers in real time
*If the servers go down THEN we can parse the logs while host is still down

Review Logs: • Troubleshooting ⇒ logs provide clues about what, why, when
• Security Auditing ⇒ logs track user activity, login attempts, system changes
Checkpointing
• Performance Monitoring: provide insights into performance
• Debugging Applications: Devs use app logs to identify bugs, etc

LOG FORMATS (A) BSD (RFC3164) (B) IETF (RFC5424)
(I) LOGS MUST BE TIMESTAMPED (II) LOGS MUST BE CATEGORIZED & SEVERITY
BSD SYSLOG (1) omits the year IETF (1) includes the year
SYSTEMD JOURNAL (modern) - stored in a binary format
(not readable by humans) - viewed with journalctl command
LOG ROTATION - essential: rotates, compresses & deletes oldest using rules
#logrotate • /etc/logrotate.conf • /etc/logrotate.d
1. RSYSLOG FORWARDING & COLLECTION (1 centralize collector: use ufw ports, etc.)
2. LOG AGGREGATION W/ AGENTS & A CENTRALIZED PLATFORM (eg ELK, splunk)
3. MESSAGE QUEUES (eg kafka, RabbitMQ)
4. Cloud-Native Logging Services

RSYSLOG
- can handle up to 100Gs, but doesn't have redundancy, etc.
- low cost way to get logs off the system
-

LOG AGGREGATORS
- key is that you put agents on all your machines
- the agents send the logs/data back to the centralized one
  ELK stack, Splunk, Graylog, Loki ("Grafana Promtail")
-

MESSAGE QUEUES (aka EVENT BUS)
- YOU CAN REPLAY all the events of the day
- Apache Kafka has 80% of the market

CLOUD-NATIVE LOGGING
- EVENTBRIDGE? VPC FLOW LOGS (9 Reddit)?

Sending logs off the server
- Scale
- Reliability Requirements
- Real-time analysis options
- Budget
- Expertise
- Compliance Requirements
- Existing Infrastructure

Parsing logs with text tools (effective tips)
- Understand the log format: delimiters, order of fields, timestamp
- Start simple: grep, awk, sed
- Regex w/ sed & grep, etc.
- Iterate & Refine: Initial command → then refine & experiment
- Save useful commands;

Parsing logs with special tools
SIEM - Security Information & Event Management solutions
- Splunk
- logQL
- ELK (Elasticsearch, Logstash, Kibana)
- Graylog

Libraries in Programming Languages
- Python libraries (re for regex, json, csv, pyparsing)
- Go libraries (regexp, encoding/json, encoding/csv) (grok-cli)

When to use special tools
- Large log volumes
- Complex log formats
- Need for real-time analysis & alerting
- Requirement for visualization and dashboards
- Security Analysis
- Compliance Requirements

UNIT 6 contd minute 41

What is a log?

Immutable record of an action that occured or tried to occur

We need tools to help non-tech visualize what's going on

Latency - time it takes to service a request

Traffic - measure of the demand placed on your system

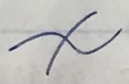
Errors - Rate of requests that, either explicitly (500's), implicitly (HTP 200), or by policy

Saturation - How full your service is... Constraints, Utilization

/etc/rsyslog.d/ ##

↳ these are read after /etc/rsyslog.conf

Raft replaces Zookeeper with Kafka



LSG

UNIT 7

Last week ① Storing logs ② Parsing logs
Immutable logs • RFC formats • Central/red storage • Rotation
Get logs off the server in real time • Parsing logs • Splunk/Loki
This week ① What do we monitor ② How do we decide what to alert
Telemetry - metrics at a distance (track metrics)
big 4: (compute, memory, disk, network
Trace - mapping a user/process as it flows through a system
front end web svr - middleware - database, ...
Alerting - severity (warn vs. critical)
Monitoring: logs - A record of what's happening within your system
never change the actual/original log: immutable
Monitoring: Metrics - A numerical assessment of system/app performance
Utilization ⇒ 0 to 100%, Saturation ⇒ queue wait times, Errors ⇒ error rates
Monitoring: Traces ⇒ how data are moving through your system
System: Utilization, Saturation, Errors ⇒ tells us how healthy the system is
The sysadmin cares about whether the system will finish w/o crashing.
Percentage of Utilization • Length of time/Delays in Saturation • Errors
The security engineer is looking for indicators of compromise
? Is the system performing out of normal operating ranges?
Is something happening to our system that shouldn't be happening
e.g. Internally by accident vs Internally malicious vs Externally malicious
e.g. high CPU may indicate bitcoin running
? Are users logging into areas they don't normally log into

SLO - Service Level Objectives
- Looks at Service level indicators, that are less than (in an upper bound) or greater than a lower bound

SLA - Service Level Agreements

Service Level Indicators
↳ what tells of the system service level (ex: working, etc.
*This are quantitative measurments of numbers
- Request Latency - Error

200 HTTP response mean OK
400 HTTP responses mean the client did something weird
500 HPP response mean there is something wrong with servs

Standardize Indicators:
- Frequency: Every 10 seconds
- Data access latency: Time to last byte of a read (baseline)
- Aggregation intervals: Averaged over 1 minute & looking over time

Why are we monitoring? Because time is of the essence
In the middle of an incident we need the data
But during forensics & after action reviews, speed is less important

DESCRIPTIVE ANALYTICS → what happened in the PAST
DIAGNOSTIC ANALYTICS → why did it happen in the PAST
PREDICTIVE ANALYTICS → what may happen in the future
PRESCRIPTIVE ANALYTICS → How can we make it happen

UNIT 7 contd

Monitoring - Push vs Pull

Push - rsyslog will catch the data,

Pull - preferred model, Prometheus
- scrapers data into it's database (nodeexporter, cephexporter)

Advantages of pull
- can easily tell if instances are down
- manually go to target & inspect health with the web
- easily add & remove

How do we store data that we monitor
TSDB

Alerting - how we send info from a closed system out for others
- discrete actions • Past • Current • Predictive

When do we alert? ① Security Breaches ② System failures ③ Performance

Wazuh - Open Source SIEM

LAB

Add entry to /etc/fail2ban/jail.conf

```
[sshd]
...
```

Delete existing entries

[sshd] ⇒ port=ssh, logpath=~, backend=~

#fail2ban-client get sshd banned
↳ lists ip

#fail2ban-client set sshd unbanip ....

VMT 7 lab section contd

#vi /etc/telegraf/telegraf.conf
*uncomment lines
[[outputs.influxdb_v2]]
urls = ["http://127.0.0.1:8086]
token = *copy key from influx db ⇒ generated by influxdb
organization = "lab" ⇒ as input into influxdb
bucket = "test" ⇒ as input into influxdb
[[inputs.fail2ban]]
use-sudo = "true"

#vi /etc/sudoers.d/telegraf
*insert lines