

1. List some logging and visualization tools available in the market with the preferred scenario to use one over other.

Some of the logging tools are:

ELK stack: It is the combination of Elasticsearch, Logstash and Kibana: a complete stack for logging and visualization which helps to process and analyze the logging data easily.

Elastic search helps users to find matches within datasets using a wide range of query languages. It is a high speed search and analytics engine which can be expanded into multiple clusters of server nodes and easily handles gigantic volumes of data. Kibana is a visualization tool which helps users to analyze their data and build reports with ease. Logstash is an open source, server side data processing pipeline that enables to ingest data from multiple sources simultaneously and enrich and transform it before indexed to Elasticsearch and used to aggregate and process data and send it to elastic search.

The reason to use this stack are:

It allows us to monitor apps built on open source installation, web servers and database logs.

It provides efficient log tracking and database management and visualization.

Graylog:

It is an open source centralized log management service that allows quick analyzing of logs. It is also easy to scale. It provides an easy to use interface and robust functionality.

Its built in fault tolerance can run multithreaded searches which helps us to analyze several potential threats together.

Nagios:

It simplifies data collection and information and makes it more accessible to system administrators.

It captures data in real time and feeds it to powerful search tools.

It can audit a range of network related events and helps to automate the distribution of alerts.

It can run predefined scripts if certain conditions are met which helps in automation in problem solving.

It can filter log data based on geographical location which helps to build dashboards with mapping technology which helps to understand flow of web traffic.

LOGalyze

It is designed to work as a massive pipeline in which multiple servers, apps and network devices can feed information using Simple Object Access Protocol (SOAP).

It provides a front end interface to monitor, collect and analyze data.

It allows the gathering of audit data in format required by regulatory acts.

Fluentd

It is a robust, open source tool for data collection which is compatible with most common technology tools available.

It decouples data sources from backend systems by providing a unified logging layer in between.

It can extend your logging data into other apps and drive better analysis from it with minimal effort.

Octopussy

It is an open source tool which helps to analyze logs from different networking devices (like routers, firewalls, load balancers etc) and all their applications and services supporting syslog protocol.

It sends alert messages via email and other open source instant messengers which helps to be updated to what is going on with the system.

It provides a free solution to prevent system outages, security threats and application errors.

2. Mention 10 best practises when logging. Why is log formatting necessary?

The best practices while logging are:

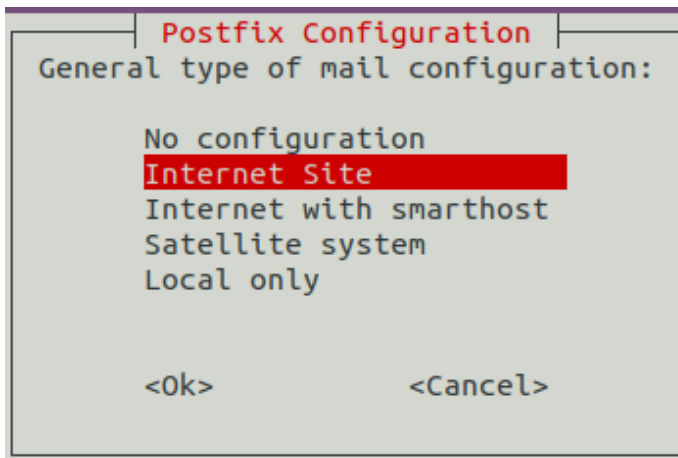
1. Choose specific goals: Choosing specific goals while logging helps to make efficient decisions and easy to extract information from generated logs.
2. Follow proper standards: Using standard log frameworks helps to control the amount of details included in logs, define levels of alert and implement log rotation policies.
3. Make accessible log routine: Using standard date and time format and providing appropriate log context helps to understand logs more easily.
4. Make sure that logs are helpful: Important log records should be highlighted if possible and meaningful log messages should be written. Like Operation Incomplete, Database Unreachable and so on.
5. Create logging standards and structure: The logging structure must be consistent and all logs should show the timestamp and name of host and logger.
6. Use proper error severity levels: The log levels should be properly mentioned like: FATAL, ERROR, WARN, INFO, DEBUG, TRACE ALL or OFF. It helps us to focus on what is important and what is not.
7. Provide appropriate details: The details in log messages must be appropriate, i.e. Not too much and no less. Saving too much log details will make it difficult to focus on what is important and Saving less details may miss some important information.
8. Use of tools to manage logs: Use of tools can help in automation and ease in managing logging information like managing log data, provides a way to search and filter log messages. Use of UI helps in visualization of data more easily.
9. Do not log sensitive information: Sensitive information like passwords, credit card informations and social security numbers must not be logged. It may expose certain vulnerabilities to the system.
10. Log in machine parsable format: Logging in machine parsable format like JSON helps us to easily evaluate the logged details and can be easily processed by existing tools.

3. Create a file in your system. Whenever someone performs some action(read, write, execute) on that file, the event should be logged somewhere.

I installed iwatch in my system using command

Sudo apt install iwatch

And accepting the agreement and using the watching process as local only, iwatch is installed in our system. We can see the status using **systemctl status iwatch**



And created a file test.txt which is to be monitored using command: **iwatch test.txt**

And the output of the file is appended to another text file using extension &>> and specifying another filename where data is to be appended as shown:

```
^C
bj@batman:~/log$ iwatch test.txt -e all_events &>> log.txt
```

When we open and close the file, log information is saved as shown in figure below:

```
GNU nano 4.8 log.txt
Wide character in print at /usr/bin/iwatch line 359.
Wide character in print at /usr/bin/iwatch line 373.
[28/नवम्बर/2021 16:07:11] * test.txt is opened
[28/नवम्बर/2021 16:09:23] * test.txt is closed
```

[I couldn't figure out how other actions are logged, even specifying the -e flag as all_events, it didn't record all events.]

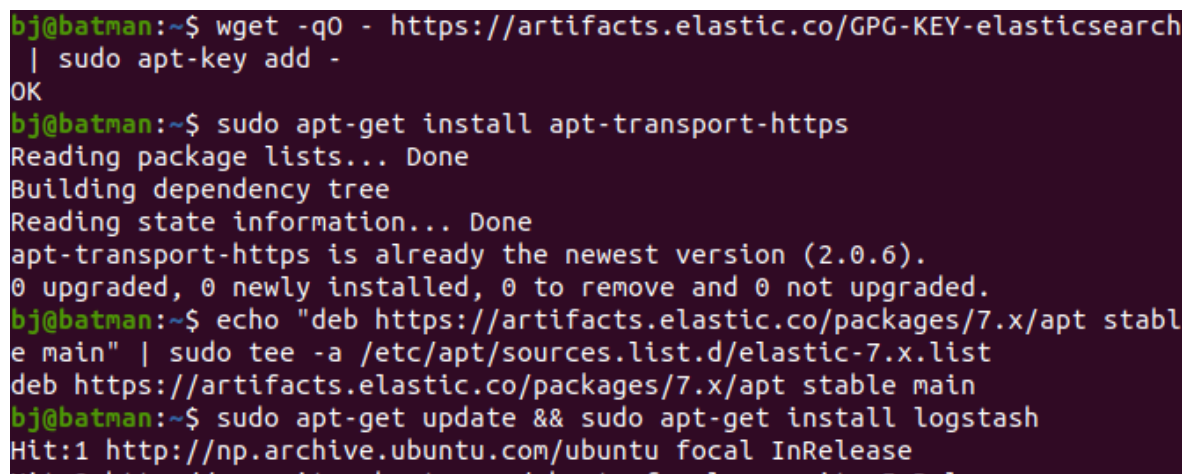
4. Install logstash in your system. download a sample nginx log from https://github.com/elastic/examples/blob/master/Common%20Data%20Formats/nginx_logs/nginx_logs , parse the logs using logstash. The parsed output must contain the geographical information like country, state etc. that the request is originating from. save the parsed output to a file in your system.

First of all, to install logstash, following commands are used:

wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -

echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-7.x.list

sudo apt-get update && sudo apt-get install logstash

A terminal window with a dark purple background and green text. The user 'bj@batman' is in the home directory. The commands and their outputs are as follows:
1. `wget -qO - https://artifacts.elastic.co/GPG-KEY-elasticsearch | sudo apt-key add -`
Output: OK
2. `sudo apt-get install apt-transport-https`
Output: Reading package lists... Done
Building dependency tree
Reading state information... Done
apt-transport-https is already the newest version (2.0.6).
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
3. `echo "deb https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee -a /etc/apt/sources.list.d/elastic-7.x.list`
Output: deb https://artifacts.elastic.co/packages/7.x/apt stable main
4. `sudo apt-get update && sudo apt-get install logstash`
Output: Hit:1 http://np.archive.ubuntu.com/ubuntu focal InRelease
Hit:2 https://artifacts.elastic.co/packages/7.x/apt stable InRelease

And logstash is not started by default so, it is started using command:

Sudo systemctl start logstash

To check the status,

Sudo systemctl status logstash

```

bj@batman:/etc/logstash/conf.d$ sudo systemctl status logstash
● logstash.service - logstash
   Loaded: loaded (/etc/systemd/system/logstash.service; disabled; vend
   Active: active (running) since Sun 2021-11-28 13:49:46 +0545; 13s ago
   Main PID: 15601 (java)
     Tasks: 19 (limit: 9110)
    Memory: 545.1M
    CGroup: /system.slice/logstash.service
            └─15601 /usr/share/logstash/jdk/bin/java -Xms1g -Xmx1g -XX:+>

Nov 28 13:49:46 batman systemd[1]: Started logstash.
Nov 28 13:49:46 batman logstash[15601]: Using bundled JDK: /usr/share/log>
Nov 28 13:49:47 batman logstash[15601]: OpenJDK 64-Bit Server VM warning:>
lines 1-12/12 (END)
^C
bj@batman:/etc/logstash/conf.d$ cd /etc/logstash/conf.d/
bj@batman:/etc/logstash/conf.d$ ls
logstash.conf
bj@batman:/etc/logstash/conf.d$ nano logstash.conf
bj@batman:/etc/logstash/conf.d$ sudo apt install default-jre
Reading package lists... Done
Building dependency tree
Reading state information... Done

```

Now, java runtime environment should be installed which is installed using command:

Sudo apt install default-jre

In the directory /etc/logstash/conf.d a file is made, *logstash.conf* with content shown in figure:

Cd /etc/logstash/conf.d

Nano logstash.conf

```

input {
    file {
        type => "nginx-logs"
        path => "/home/bj/logtest/log/nginx_logs"
        start_position => "beginning"
        sincedb_path => "/dev/null"
    }
}

filter {
    grok {
        match => {
            "message" => "%{IP:remote_ip}"
        }
    }

    geoip {
        source => "remote_ip"
    }
}

output {
    file { path => "/home/bj/logtest/log/logstashlogs.log" }
}

```

Now we run a logstash script from within the /usr/share/logstash/bin directory:

Cd /usr/share/logstash/bin

Sudo ./logstash --path.settings /home/bj/logtest/log --path.data sensor39 -f /etc/logstash/conf.d/

```

[INFO ] 2021-11-28 14:52:35.111 [LogStash::Runner] runner - Logstash shut down.
bj@batman:/usr/share/logstash/bin$ sudo ./logstash --path.settings /home/bj/log
test/log --path.data sensor39 -f /etc/logstash/conf.d/
Using bundled JDK: /usr/share/logstash/jdk
OpenJDK 64-Bit Server VM warning: Option UseConcMarkSweepGC was deprecated in v
ersion 9.0 and will likely be removed in a future release.
WARNING: Could not find logstash.yml which is typically located in $LS_HOME/con
fig or /etc/logstash. You can specify the path using --path.settings. Continuin
g using the defaults
Could not find log4j2 configuration at path /home/bj/logtest/log/log4j2.properties

```

After this process is completed: we can use grep to view the longitude information in the logstashlogs.log file, which is specified in the above configuration file.

Here, we can see country code, ip address, location (with longitude and latitude) and other geo information for that specific IP.

```

bj@batman:~/logtest/log$ grep longitude logstashlogs.log
{"@timestamp":"2021-11-28T09:04:07.755Z","host":"batman","type":"nginx-logs","remote_ip":"7.59.4.07",
"geoip":{"timezone":"America/Chicago","country_code3":"US","longitude":-97.822,"latitude":37.751,"ip"
:"7.59.4.7","country_code2":"US","location":{"lon":-97.822,"lat":37.751},"country_name":"United State
s","continent_code":"NA"},"@version":"1","path":"/home/bj/logtest/log/nginx_logs","message":"<pat
h fill-rule=\"evenodd\" d=\"M8 0C3.58 0 0 3.58 0 8c0 3.54 2.29 6.53 5.47 7.59.4.07.55-.17.55-.38 0-.1

```