**Monitoring for Enterprise Applications**

**Monitoring Solutions for Jenkins –**

1. Jenkins Monitoring with **Prometheus and Grafana (Opensource)**
2. Jenkins Monitoring with **Datadog (Licensed)**
3. Jenkins Server Monitoring with **AWS CloudWatch (If Jenkins Server is hosted in AWS)**
4. Jenkins Log monitoring with **ELK (Opensource and can be hosted in AWS)**
5. Other Solutions with **Zabbix and Nagios (Opensource)**
6. **Splunk** with Jenkins

**Considerations –**

* Cost
* Securing data
* Implementing a failproof reliable system
* Performance
* Effort involved in building and to Operate
* Open source vs Proprietary
* Level of Automation and Self remediation

**Jenkins Events and logs monitoring (ELK stack)**

**Step 1 –**

Collecting Jenkins Build logs from Jenkins. In the Jenkins Build server, the Jenkins build logs will be present under /var/log/jenkins/jenkins.log. These logs contain record of Jenkins job execution and output. It has build name, build number, execution time of build, build results (success, failure and aborted) and also errors.

**Step 2 –**

An agent in Jenkins Server to send logs to the ETL system. This agent can be Prometheus plugin, fluentD agent, File beat, Logstash or Docker Application logs if Jenkins in running as a container. The logs are sent to a destination for further Analysis.

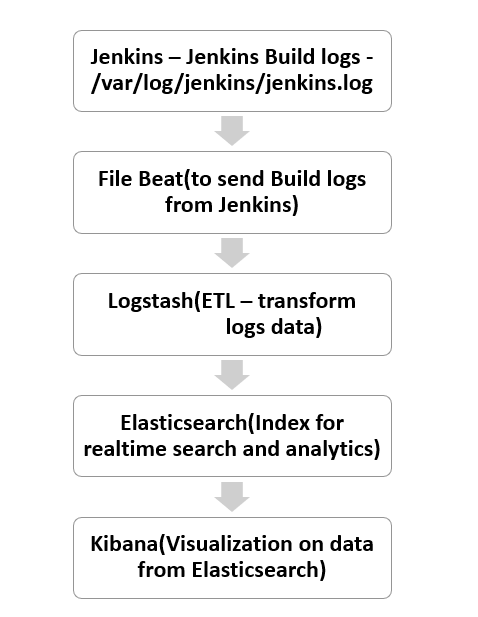
**Step 3 –**

Logs are sent to Elasticsearch where the data can be indexed and searched for patterns. Elasticsearch is can be run on AWS as Elastic search cluster which also exposes a Kibana Endpoint.

Logs can be stored in S3, DynamoDB, streamed through Kinesis. Elasticsearch cluster is attached with EBS for storage.

**Step 4 –**

Data can be visualized using Kibana by creating dashboards.



**Time series data (Prometheus and Grafana, TICK stack)**

Time series data is data that is timestamped by sequence of events. Jenkins statistics like Total builds, number of jobs, overall Jenkins health, JVM free memory can be tracked through Prometheus and visualized using Grafana. Grafana is open source and can make observability dashboards.

**Step 1 –**

Spin up a Prometheus Server. Install Prometheus plugin in Jenkins. The Prometheus server will scrape data for metrics from this endpoint.

**Step 2 –**

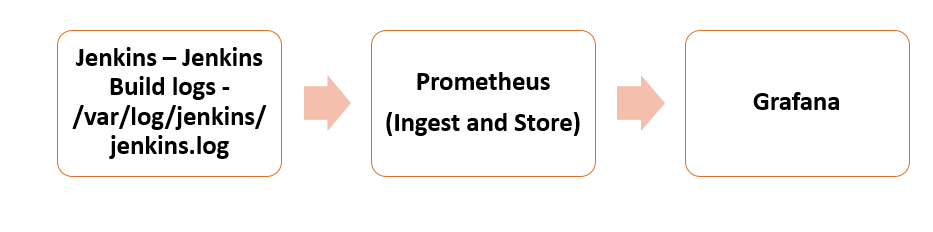
Choose relevant metrics for Jenkins in Prometheus.

**Step 3 –**

Spin up a server for Grafana and login to the dashboard. Integration between Grafana and Prometheus is established by configuring Prometheus as a data source.

**Step 4 –**

Create dashboard in Grafana for monitoring the required stats.



**Jenkins Server Monitoring –**

Jenkins Server will be run as a cluster with master/slave configuration. When run in a container, the logs can be got from the docker logs. AWS CloudWatch can send events, logs, metrics through which Alarms can be configured for various metrics.

**Notification plugin, AWS CloudWatch logs publisher plugin 🡪 AWS CloudWatch 🡪 CloudWatch Dashboards, Alarms, Metrics, Log Insights**

**InfluxDB, Prometheus, Timescale DB as datastore -**

For Jenkins monitoring we can use InfluxDB as the data store for fast retrieval of time series data.



**Implementation of above architectures –**

Any of the above-mentioned open source components can be combined to build a monitoring solution for Jenkins.

Most of them are open source and involves configuring them in a server (AWS EC2 instance) and integrating it with the Jenkins server.

A plugin or agent must be installed in Jenkins to fetch all the logs and send it for analysis.

On choosing any of the above tools, the implementation would follow the below steps,

