Monitoring & Alerting Stack

-Shubham Sachdeva
-Pranjal Mehta

What is Monitoring & Alerting?

Monitoring means continuously collecting data from systems, servers, and applications to check their health and performance.
It tracks things like:

- CPU and memory usage
- Network traffic
- Application errors
- Request latency

Alerting is about getting notified automatically when something abnormal happens like high CPU, service downtime, or too many failed logins.
Alerts are triggered based on predefined thresholds or conditions.

Prometheus: Collecting Metrics

Concept:

- Prometheus is an open-source tool for metrics-based monitoring.
- It scrapes metrics from targets (applications, services, infrastructure) via HTTP endpoints.
- Stores data in a time-series format (value + timestamp).

Key Concepts:

- Uses a pull model – Prometheus reaches out to services at regular intervals.
- Supports PromQL – a powerful query language for filtering and analyzing metrics.
- Allows users to define alerting rules based on metric thresholds.

Use Cases:

- Monitoring server resource usage (CPU, memory, disk)
- Tracking application response times
- Creating custom service-level indicators (SLIs)

Why It’s Useful:

- Lightweight and easy to set up
- Flexible metric collection with labels for filtering
- Great foundation for real-time and historical analysis

Grafana – Visualizing Metrics

Concept:

- Grafana is a powerful tool used to visualize metrics from Prometheus and other sources.
- It allows you to build interactive dashboards to monitor system performance at a glance.

What You Can Do:

- Create real-time graphs and charts for CPU, latency, request counts, etc.
- Set up panel-based dashboards for teams or specific services.
- Define alert rules visually and send alerts to multiple channels (email, Slack, PagerDuty).

Why Grafana Matters:

- Makes complex data easy to understand
- Helps in identifying trends and patterns quickly
- Useful for both operations teams and developers

Mimir: Long-Term Metrics Storage

Content:

- Prometheus is great, but it keeps data only for a short time (usually days).
- Grafana Mimir solves this by acting as remote storage for Prometheus metrics.

Key Features:

- Highly scalable – stores billions of time series
- Supports long-term retention – weeks, months, or even years
- Designed for high availability and data durability
- Compatible with Prometheus remote_write — easy to plug in

Why Mimir is Needed:

- You can analyze historical trends (e.g., CPU usage over 6 months)
- Helps with capacity planning, SLO tracking, and audits
- Keeps Prometheus lightweight by offloading storage

PagerDuty: Incident Response and Alerting

Content:

- When something breaks, you need the right person to be notified immediately that’s where PagerDuty helps.

How it Works:

- Prometheus detects a problem → Alertmanager sends alert → PagerDuty notifies on-call engineer.
- Sends alerts via:
  - SMS, email, phone call
  - Mobile app push notification

Key Features:

- On-call schedules – define who’s responsible and when
- Escalation policies – if person A doesn’t respond, notify person B
- Incident tracking – log, acknowledge, and resolve issues

Why PagerDuty Is Valuable:

- Ensures no critical alert goes unnoticed
- Helps teams respond quickly and reduce downtime
- Makes alerting organized, not chaotic

ObServe: Log Management & Correlation

Concept:

- ObServe is a cloud-based platform for managing and analyzing logs.
- Logs help answer: "What exactly happened?" when something goes wrong.

What It Does:

- Ingests logs from apps, containers, services, etc.
- Uses schema-on-read – you don’t need to define log format up front
- Lets you search, filter, and analyze logs quickly

Key Benefits:

- Connects logs, metrics, and traces in one place
- Enables root cause analysis during incidents
- Fast and flexible UI for investigations

Use Case:

- You get an alert from Prometheus → You check dashboards in Grafana → You dig into logs using ObServe to find out why it happened

Benefits of This Monitoring & Alerting Stack

Concept:

- Prometheus: Tracks system metrics in real-time
- Grafana: Visual dashboards for faster understanding
- Mimir: Scalable storage for long-term metric history
- PagerDuty: Alert delivery with on-call automation
- ObServe: Deep dive into logs for root cause analysis

Combined Value:

- Full observability: metrics + logs + alerts
- Faster troubleshooting, fewer outages
- Scales from small teams to enterprise environments