Monitoring and Logging

Overview
AutoSync Cloud Service employs a comprehensive monitoring and logging system to track application performance, ensure system health, and detect potential issues before they impact users. This page outlines the monitoring tools used, types of logs generated, and best practices for configuring and managing logs.

1.1 Monitoring Tools

AutoSync Cloud Service integrates several monitoring tools to provide real-time insights into the system's health and performance:

- **Prometheus**: A powerful open-source monitoring system and time-series database used to gather metrics and monitor the health of the system. It collects data such as CPU usage, memory usage, network activity, and custom application metrics.
  - **Integration**: Prometheus is deployed within Kubernetes clusters to monitor containerized services.
  - **Alerting**: Prometheus supports alerting rules for notifying stakeholders when thresholds are exceeded (e.g., CPU usage > 80%).
- **Grafana**: A visualization tool used in conjunction with Prometheus for real-time monitoring. Grafana dashboards provide detailed insights into application performance, resource utilization, and user activities.
  - **Custom Dashboards**: Tailored dashboards display system metrics, service statuses, and error logs, allowing engineers to monitor performance trends.
  - **Alerts**: Configured Grafana alerts notify administrators of any system anomalies or failures.
- **Cloud-native Monitoring Solutions**: Cloud services such as **AWS CloudWatch** or **Google Stackdriver** may also be used to monitor infrastructure health, including VMs, storage, and network activity.

1.2 Types of Logs

- **Audit Logs**: Detailed records of all system events, such as user logins, data modifications, and configuration changes. These logs help track user activity and detect potential security incidents.
- **Application Logs**: Logs generated by the backend services and microservices, providing insights into application-level events (e.g., API calls, error messages, and exceptions).
- **System Logs**: Logs from the underlying operating system and container runtime (e.g., Kubernetes, Docker). These logs provide data about server resource usage and system-level issues.
- **Access Logs**: Logs related to HTTP request traffic, including IP addresses, response times, and status codes. These logs help analyze user access patterns and track potential malicious activity.
- **Error Logs**: Logs generated when system errors, exceptions, or crashes occur. These are critical for troubleshooting and improving system reliability.

1.3 Log Management

- **Log Retention**: Logs are retained for a specified period (e.g., 90 days) to ensure compliance with industry standards and regulations. Older logs are archived or securely deleted.
- **Centralized Logging**: All logs are aggregated in a centralized logging system using the **ELK Stack (Elasticsearch, Logstash, and Kibana)** or **AWS CloudWatch Logs** to enable fast searching, filtering, and analysis.
- **Search and Filter**: Kibana or CloudWatch allows system administrators to query logs using specific parameters (e.g., error types, time range) for quick troubleshooting.
- **Alerting**: Automated alerts notify engineers when critical errors occur, such as high latency or repeated service failures.