Grafana + Loki Monitoring Stack
Complete Setup Guide for Security Monitoring

Overview ■

Grafana with Loki provides a powerful monitoring solution for security metrics visualization. This stack enables real-time monitoring of network connections, system resources, and security events.

- Grafana: Visualization dashboard
- Loki: Log aggregation system
- Docker: Containerization platform

Architecture ■■

Below is the high-level architecture of the monitoring stack:

- Data Sources → Loki (Log Store) → Grafana (Dashboard)

Prerequisites ■

Ensure the following system requirements and configurations are in place before deployment.

- System Requirements: Ubuntu 20.04/22.04, 2+ CPU cores, 4GB+ RAM, 10GB+ Storage, Docker Installed
- Network Ports: 3000 (Grafana UI), 3100 (Loki API)

Quick Start ■

Use the following commands for one-command deployment and verification.

- Make the setup script executable and run: chmod +x grafana-full-setup.sh && ./grafana-full-setup.sh
- Verify running services with: docker ps
- Access Grafana Dashboard: http://:3000 (admin/admin)

Script Breakdown ■

This section explains the key setup and metric scripts used in the monitoring stack.

- grafana-security-setup.sh — Creates and runs Grafana and Loki containers using Docker Compose.
- send-security-metrics.sh — Collects and sends real-time security metrics to Loki.
- import-dashboard.sh — Automates Grafana dashboard import via API.

Dashboard Setup ■

You can create or import dashboards manually or via script for visualization of collected metrics.

- Add Loki as a data source in Grafana: Configuration → Data Sources → Loki → http://loki:3100
- Create panels for metrics like IP Connections, Open Ports, Containers, and ARP Entries.

Troubleshooting ■

Common issues and diagnostic commands for maintaining Grafana-Loki services.

- No Data in Panels — Check if Loki is receiving data and Grafana data source configuration.
- Containers Not Starting — Inspect Docker service and container logs.
- Port Conflicts — Identify and free occupied ports before container start.

Maintenance ■

Regular upkeep ensures optimal performance of your monitoring stack.

- Update containers: docker-compose pull && docker-compose up -d
- Clean up unused data: docker system prune -f
- Backup dashboards: curl -s http://admin:admin@localhost:3000/api/search | jq .

Conclusion ■

Your Grafana + Loki security monitoring stack is now ready! You now have real-time monitoring, visual dashboards, and an automated data pipeline for enhanced security visibility.