Practical – 10

AIM: Study of any two Open-source tools in DevOps for Infrastructure Automation, Configuration Management, Deployment Automation, Performance Management, Log Management and Monitoring.

Tool 1: Ansible (Configuration Management & Automation)

Introduction:

Ansible is an open-source automation tool used for configuration management, application deployment, and task automation. It uses simple YAML syntax (called playbooks) and requires no agent on target machines, making it lightweight and easy to use.

Features:

- Agentless architecture
- Uses SSH for communication
- Simple playbook-based automation
- Scales to manage thousands of nodes
- Idempotent (ensures the desired state is achieved without repetition)
- Declarative language allows easy definition of infrastructure as code
- Easily integrates with cloud platforms like AWS, Azure, and GCP
- Supports roles for reusable playbooks and tasks
- Large community and extensive module support

Use Case:

Used to install and configure web servers (like Apache, Nginx) on multiple machines with a single command. It is widely adopted for provisioning servers, automating application deployments, and ensuring consistency across environments.

For example, in a large-scale deployment environment, Ansible can be used to set up 100+ web servers with the same configuration in a matter of minutes. It helps in reducing human error and improving deployment speed during CI/CD processes.

Tool 2: Prometheus (Monitoring & Performance Management)

Introduction:

Prometheus is an open-source system monitoring and alerting toolkit originally developed at SoundCloud. It is now part of the Cloud Native Computing Foundation (CNCF). Prometheus collects metrics from configured targets at given intervals and stores them in a time-series database.

Features:

- Powerful query language (PromQL)
- Multi-dimensional data model
- Pull-based metrics collection
- Integration with Grafana for visualization
- Alerting capabilities with Alertmanager
- Automatically discovers targets via service discovery
- Stores data locally and supports federation for scaling
- Supports blackbox and whitebox monitoring
- Integration with Kubernetes for pod-level monitoring

Use Case:

Prometheus can be used to monitor server CPU usage, memory consumption, request rates, and application performance metrics. It provides real-time visibility into system health and can trigger alerts based on custom thresholds**.**

Prometheus is often used with Grafana to create real-time dashboards. For instance, it can monitor microservices performance, identify bottlenecks, and send alerts to Slack or email when system health degrades, ensuring proactive resolution.

Feature	**Ansible**	**Prometheus**
Purpose	Configuration Management, Automation	Monitoring & Alerting
Language	YAML	PromQL (Query Language)
Execution Model	Agentless (push)	Pull-based metrics collection
Primary Use	Server provisioning, app deployment	System & app performance monitoring
Integration	AWS, GCP, Azure	Grafana, Kubernetes, Alertmanager
Feature	Ansible	Prometheus
Purpose	Configuration Management, Automation	Monitoring & Alerting

Conclusion:

Both Ansible and Prometheus are powerful tools that support automation and observability in the DevOps lifecycle. Ansible helps manage infrastructure and deployments effectively, while Prometheus ensures continuous performance monitoring and quick issue detection.