Topics from the previous lecture

1. Site to Site VPN
2. Client to Site VPN
3. Open Source VPN -

Advanced AWS Workshop

Instructor: Govind Kumar

Date: 21-May-2025 | Time: 9:00 PM - 11:30 PM IST

CloudWatch Metrics

Monitor your AWS resources and applications with customizable dashboards.

CloudWatch Alarms

Create alerts based on metrics and take automated actions when thresholds are breached.

CloudWatch Logs

Centralize logs from all your systems, applications, and AWS services.

AWS CloudWatch

A monitoring and observability service built for DevOps engineers, developers, and IT managers.

What is CloudWatch?

Amazon CloudWatch is a monitoring and observability service that provides data and actionable insights for AWS, hybrid, and on-premises applications and infrastructure resources. You can collect and track metrics, collect and monitor log files, and set alarms.

Key Components

- CloudWatch Metrics
- CloudWatch Alarms
- CloudWatch Logs
- CloudWatch Events/EventBridge
- CloudWatch Dashboards

Core Capabilities

- Collect metrics and logs
- Monitor applications
- Set alarms and create automated actions
- Visualize logs and metrics
- Troubleshoot issues

- CloudWatch automatically collects metrics from many AWS services
- Standard metrics are collected at 5-minute intervals by default
- Detailed monitoring provides 1-minute metrics (additional charges apply)
- Custom metrics can be published at minimum 1-minute intervals

CloudWatch Metrics

CloudWatch Metrics

Time-series data points that represent the behavior of your resources and applications.

Understanding CloudWatch Metrics

Metrics are data about the performance of your systems. By default, many services provide free metrics for resources. You can also publish your own custom metrics.

Default Metrics

- EC2: CPU Utilization, Network, Disk I/O
- S3: Bucket Size, Number of Objects
- RDS: Database Connections, CPU
- Lambda: Invocations, Duration, Errors
- ELB: Request Count, Latency

Metric Properties

- Namespace (e.g., AWS/EC2)
- Metric Name (e.g., CPUUtilization)
- Dimensions (e.g., InstanceId)
- Resolution (Standard or High)
- Statistics (Average, Min, Max, Sum)

- Metrics exist only in the region in which they are created
- Metrics cannot be deleted but expire after 15 months if no new data is published
- Up to 10 dimensions can be assigned to a metric

CloudWatch Alarms

CloudWatch Alarms

Monitor metrics and trigger actions when thresholds are breached.

Working with CloudWatch Alarms

CloudWatch Alarms watch a single metric over a specified time period and perform actions based on the metric's value relative to a threshold.

Alarm States

- OK - Metric is within threshold
- ALARM - Metric is outside threshold
- INSUFFICIENT_DATA - Not enough data

Alarm Actions

- Send notification to SNS topic
- Execute Auto Scaling action
- EC2 actions (stop, terminate, reboot)
- Create OpsItems or incidents
- Trigger Lambda functions

- Alarms can be based on static thresholds or anomaly detection
- Composite alarms combine multiple alarms using AND/OR logic
- Evaluation periods determine how many data points trigger state change

CloudWatch Logs

CloudWatch Logs

Centralize logs from all your systems, applications, and AWS services.

Log Management with CloudWatch

CloudWatch Logs enables you to centralize the logs from all of your systems, applications, and AWS services in a single, highly scalable service.

Key Concepts

- Log Groups - Collection of log streams
- Log Streams - Sequence of log events
- Log Events - Records of activity
- Metric Filters - Extract metrics from logs
- Subscription Filters - Real-time processing

Log Analysis Tools

- CloudWatch Logs Insights - Query logs
- Contributor Insights - Top contributors
- Live Tail - Real-time log viewing
- Export to S3 - Long-term storage
- Integration with Athena - SQL queries

- Log data can be retained for as long as needed with configurable retention
- CloudWatch Logs Insights uses purpose-built query language for log analysis
- Logs can be encrypted using KMS for enhanced security

CloudWatch Best Practices

Monitoring Best Practices

Optimize your CloudWatch implementation for effective monitoring and observability.

CloudWatch Implementation Strategies

Follow these best practices to get the most out of your CloudWatch implementation and build a robust monitoring strategy.

Monitoring Strategy

- Define clear monitoring objectives
- Identify critical metrics for each service
- Set appropriate thresholds based on baselines
- Implement multi-level alerting
- Document runbooks for alarm responses

Technical Implementation

- Use detailed monitoring for critical resources
- Create consolidated dashboards
- Implement cross-account monitoring
- Set up log retention policies
- Use metric math for derived metrics

Cost Optimization

- Monitor CloudWatch usage and costs
- Use log filter patterns effectively
- Implement appropriate log retention
- Consider CloudWatch Logs Insights for analysis
- Use metric filters instead of custom metrics when possible

Integration with AWS Services

- Use EventBridge for event-driven architectures
- Integrate with AWS X-Ray for tracing
- Connect with AWS Systems Manager for remediation
- Leverage AWS Security Hub for security monitoring
- Use AWS Backup for data protection monitoring

Sample Dashboard

1h | 6h | 24h | 7d

CPU Utilization

42%

Memory Usage

2.1 GB

Network In/Out

3.4 MB/s

Request Count

1,243

CloudWatch Notification Center

- All Alerts 4
- Critical 1
- Warning 2
- Info 1

High CPU Utilization
EC2 instance i-0123456789abcdef0 has exceeded 90% CPU utilization for 5 minutes
10 minutes ago EC2

Elevated Error Rate
Lambda function ProcessOrders has 5% error rate in the last 15 minutes
25 minutes ago Lambda

Low Free Storage Space
RDS instance database-1 has less than 20% free storage space
1 hour ago RDS

Auto Scaling Event
Auto Scaling group web-tier-asg has launched 2 new instances
3 hours ago Auto Scaling