Amazon CloudWatch FAQs

Q: What is Amazon CloudWatch?

Amazon CloudWatch is a monitoring service for AWS cloud resources and the applications you run on AWS. You can use Amazon CloudWatch to collect and track metrics, collect and monitor log files, and set alarms. Amazon CloudWatch can monitor AWS resources such as Amazon EC2 instances, Amazon DynamoDB tables, and Amazon RDS DB instances, as well as custom metrics generated by your applications and services, and any log files your applications generate. You can use Amazon CloudWatch to gain system-wide visibility into resource utilization, application performance, and operational health. You can use these insights to react and keep your application running smoothly.

To get started with monitoring, you can use Automatic Dashboards with built-in AWS best practices, explore account and resource-based view of metrics and alarms, and easily drill-down to understand the root cause of performance issues.

Q: What can I use to access CloudWatch?

Amazon CloudWatch can be accessed via API, command-line interface, AWS SDKs, and the AWS Management Console.

Q: Which operating systems does Amazon CloudWatch support?

Amazon CloudWatch receives and provides metrics for all Amazon EC2 instances and should work with any operating system currently supported by the Amazon EC2 service.

Q: What access management policies can I implement for CloudWatch?

Amazon CloudWatch integrates with AWS Identity and Access Management (IAM) so that you can specify which CloudWatch actions a user in your AWS Account can perform. For example, you could create an IAM policy that gives only certain users in your organization permission to use GetMetricStatistics. They could then use the action to retrieve data about your cloud resources.

You can't use IAM to control access to CloudWatch data for specific resources. For example, you can't give a user access to CloudWatch data for only a specific set of instances or a specific LoadBalancer. Permissions granted using IAM cover all the cloud resources you use with CloudWatch. In addition, you can't use IAM roles with the Amazon CloudWatch command line tools.

Q: What is Amazon CloudWatch Logs?

Amazon CloudWatch Logs lets you monitor and troubleshoot your systems and applications using your existing system, application and custom log files.

With CloudWatch Logs, you can monitor your logs, in near real time, for specific phrases, values or patterns. For example, you could set an alarm on the number of errors that occur in your system logs or view graphs of latency of web requests from your application logs. You can then view the original log data to see the source of the problem. Log data can be stored and accessed indefinitely in highly durable, low-cost storage so you don’t have to worry about filling up hard drives.

Q: What kinds of things can I do with CloudWatch Logs?

CloudWatch Logs is capable of monitoring and storing your logs to help you better understand and operate your systems and applications. You can use CloudWatch Logs in a number of ways.

Real time application and system monitoring: You can use CloudWatch Logs to monitor applications and systems using log data. For example, CloudWatch Logs can track the number of errors that occur in your application logs and send you a notification whenever the rate of errors exceeds a threshold you specify. CloudWatch Logs uses your log data for monitoring; so, no code changes are required.

Long term log retention: You can use CloudWatch Logs to store your log data indefinitely in highly durable and cost effective storage without worrying about hard drives running out of space. The CloudWatch Logs Agent makes it easy to quickly move both rotated and non rotated log files off of a host and into the log service. You can then access the raw log event data when you need it.

Q: What platforms does the CloudWatch Logs Agent support?

The CloudWatch Logs Agent is supported on Amazon Linux, Ubuntu, CentOS, Red Hat Enterprise Linux, and Windows. This agent will support the ability to monitor individual log files on the host.

Q: Does the CloudWatch Logs Agent support IAM roles?

Yes. The CloudWatch Logs Agent is integrated with Identity and Access Management (IAM) and includes support for both access keys and IAM roles.

Q: What is Amazon CloudWatch Logs Insights?

Amazon CloudWatch Logs Insights is an interactive, pay-as-you-go, and integrated log analytics capability for CloudWatch Logs. It helps developers, operators, and systems engineers understand, improve, and debug their applications, by allowing them to search and visualize their logs. Logs Insights is fully integrated with CloudWatch, enabling you to manage, explore, and analyze your logs. You can also leverage CloudWatch Metrics, Alarms and Dashboards with Logs to get full operational visibility into your applications. This empowers you to understand your applications, make improvements, and find and fix problems quickly, so that you can continue to innovate rapidly. You can write queries with aggregations, filters, and regular expressions to derive actionable insights from your logs. You can also visualize timeseries data, drill down into individual log events, and export your query results to CloudWatch Dashboards.

Q: How can I get started with CloudWatch Logs Insights?

You can immediately start using Logs Insights to run queries on all your logs being sent to CloudWatch Logs. There is no setup required and no infrastructure to manage. You can access Logs Insights from the AWS Management Console or programmatically through your applications by using the AWS SDK.

Q: What is CloudWatch Container Insights?

CloudWatch Container Insights is a feature for monitoring, troubleshooting, and alarming on your containerized applications and microservices. Container Insights simplifies the isolation and analysis of performance issues impacting your container environment. DevOps and systems engineers have access to automatic dashboards in the CloudWatch console, giving them end-to-end operational visibility of metrics, logs, and distributed traces summarizing the performance and health of their [Amazon Elastic Container Service for Kubernetes (EKS)](https://aws.amazon.com/eks/" \t "_blank), [Amazon Elastic Container Service (ECS)](https://aws.amazon.com/ecs/" \t "_blank), [AWS Fargate](https://aws.amazon.com/fargate/" \t "_blank), and [Kubernetes](https://aws.amazon.com/kubernetes/" \t "_blank) clusters by pods/tasks, containers, and services.

Q: How can I get started with CloudWatch Container Insights??

You can get started collecting detailed performance metrics, logs, and metadata from your containers and clusters in just a few clicks by following these steps in the [CloudWatch Container Insights documentation](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/ContainerInsights.html" \t "_blank).

## **Pricing**

Q: How much does Amazon CloudWatch cost?

Please see our [pricing page](https://aws.amazon.com/cloudwatch/pricing/" \t "_blank) for the latest information.

Q: Does the Amazon CloudWatch monitoring charge change depending on which type of Amazon EC2 instance I monitor?

No, the Amazon CloudWatch monitoring charge does not vary by Amazon EC2 instance type.

Q: Do your prices include taxes?

Except as otherwise noted, our prices are exclusive of applicable taxes and duties, including VAT and applicable sales tax. [Learn more.](https://aws.amazon.com/c-tax-faqs/)

Q: Why does my AWS monthly bill for CloudWatch appear different between July 2017 and previous months?

Prior to July 2017, charges for CloudWatch were split under two different sections in your AWS bill and Cost and Usage Reports. For historical reasons, charges for CloudWatch Alarms, CloudWatch Metrics, and CloudWatch API usage were reported under the “Elastic Compute Cloud” (EC2) detail section of your bill, while charges for CloudWatch Logs and CloudWatch Dashboards are reported under the “CloudWatch” detail section. To help consolidate and simplify your monthly AWS CloudWatch usage and billing, we moved the charges for your CloudWatch Metrics, Alarms, and API usage from the “EC2” section of your bill to the “CloudWatch” section, effectively bringing together all of your CloudWatch monitoring charges under the “CloudWatch” section. Note that this has no impact to your total AWS bill amount. Your bill and Cost and Usage Reports will now simply display charges for CloudWatch under a single section.

Additionally, there is a Billing Metric in CloudWatch named “Estimated Charges” that can be viewed as Total Estimated Charge or broken down By Service. The “Total Estimated Charge” metric will not change. However, the “EstimatedCharges” metric broken down by Service will change for dimension ServiceName equal to “AmazonEC2” and dimension ServiceName equal to “AmazonCloudWatch”. Due to the billing consolidation, you may see that your AmazonEC2 billing metric decrease and AmazonCloudWatch billing metric increase as usage and billing charges get moved out of EC2 and into CloudWatch.

Q: How is CloudWatch Logs Insights priced?

Logs Insights is priced per query and charges based on the amount of ingested log data scanned by the query. For additional details about pricing, you can see [CloudWatch pricing](https://aws.amazon.com/cloudwatch/pricing/" \t "_blank).

Q: Does CloudWatch Logs Insights charge me for cancelled queries?

Yes, if you cancel a query manually, you are charged for the amount of ingested log data scanned up to the point at which you cancelled the query.

Q: Does CloudWatch Logs Insights charge me for failed queries?

No, you are not charged for failed queries.

Q: How is CloudWatch Container Insights priced?

CloudWatch Container Insights automatically collects custom metrics from performance events ingested as CloudWatch Logs from your container environment. More details on pricing is available on the [CloudWatch pricing page](https://aws.amazon.com/cloudwatch/pricing/" \t "_blank).

## **AWS resource & custom metrics monitoring**

Q: What can I measure with Amazon CloudWatch Metrics?

Amazon CloudWatch allows you to monitor AWS cloud resources and the applications you run on AWS. Metrics are provided automatically for a number of AWS products and services, including Amazon EC2 instances, EBS volumes, Elastic Load Balancers, Auto Scaling groups, EMR job flows, RDS DB instances, DynamoDB tables, ElastiCache clusters, RedShift clusters, OpsWorks stacks, Route 53 health checks, SNS topics, SQS queues, SWF workflows, and Storage Gateways. You can also monitor custom metrics generated by your own applications and services.

Q: What is the retention period of all metrics?

CloudWatch launched High Resolution Custom Metrics on July 26, 2017. This enables you to publish and store custom metrics down to 1-second resolution. Extended retention of metrics was launched on November 1, 2016, and enabled storage of all metrics for customers from the previous 14 days to 15 months. CloudWatch retains metric data as follows:

* Data points with a period of less than 60 seconds are available for 3 hours. These data points are high-resolution custom metrics.
* Data points with a period of 60 seconds (1 minute) are available for 15 days
* Data points with a period of 300 seconds (5 minute) are available for 63 days
* Data points with a period of 3600 seconds (1 hour) are available for 455 days (15 months)

Data points that are initially published with a shorter period are aggregated together for long-term storage. For example, if you collect data using a period of 1 minute, the data remains available for 15 days with 1-minute resolution. After 15 days this data is still available, but is aggregated and is retrievable only with a resolution of 5 minutes. After 63 days, the data is further aggregated and is available with a resolution of 1 hour. If you need availability of metrics longer than these periods, you can use the GetMetricStatistics API to retrieve the datapoints for offline or different storage.

The feature is currently available in US East (N. Virginia), US West (Oregon), US West (N. California), EU (Ireland), EU (Frankfurt), S. America (São Paulo), Asia Pacific (Singapore), Asia Pacific (Tokyo), Asia Pacific (Seoul), Asia Pacific (Mumbai), Asia Pacific (Sydney), EU (London), Canada (Central), US East (Ohio), and China (Beijing).

Q: What is the minimum resolution for the data that Amazon CloudWatch receives and aggregates?

The minimum resolution supported by CloudWatch is 1-second data points, which is a high-resolution metric, or you can store metrics at 1-minute granularity. Sometimes metrics are received by Cloudwatch at varying intervals, such as 3-minute or 5-minute intervals. If you do not specify that a metric is high resolution, by setting the StorageResolution field in the PutMetricData API request, then by default CloudWatch will aggregate and store the metrics at 1-minute resolution.

Depending on the age of data requested, metrics will be available at the resolutions defined in the retention schedules above. For example, if you request for 1-minute data for a day from 10 days ago, you will receive the 1440 data points. However, if you request for 1-minute data from 5 months back, the UI will automatically change the granularity to 1-hour and the GetMetricStatistics API will not return any output.

Q: Can I delete any metrics?

CloudWatch does not support metric deletion. Metrics expire based on the retention schedules described above.

Q: Will I lose the metrics data if I disable monitoring for an Amazon EC2 instance?

No. You can always retrieve metrics data for any Amazon EC2 instance based on the retention schedules described above. However, the CloudWatch console limits the search of metrics to 2 weeks after a metric is last ingested to ensure that the most up to date instances are shown in your namespace.

Q: Can I access the metrics data for a terminated Amazon EC2 instance or a deleted Elastic Load Balancer?

Yes. Amazon CloudWatch stores metrics for terminated Amazon EC2 instances or deleted Elastic Load Balancers for 15 months.

Q: Why does the graphing of the same time window look different when I view the metrics in 5 minute and 1 minute periods?

If you view the same time window in a 5 minute period versus a 1 minute period, you may see that data points are displayed in different places on the graph. For the period you specify in your graph, Amazon CloudWatch will find all the available data points and calculates a single, aggregate point to represent the entire period. In the case of a 5 minute period, the single data point is placed at the beginning of the 5 minute time window. In the case of a 1 minute period, the single data point is placed at the 1 minute mark. We recommend using a 1 minute period for troubleshooting and other activities that require the most precise graphing of time periods.

Q: What is a Custom Metric?

You can use Amazon CloudWatch to monitor data produced by your own applications, scripts, and services. A custom metric is any metric you provide to Amazon CloudWatch. For example, you can use custom metrics as a way to monitor the time to load a web page, request error rates, number of processes or threads on your instance, or amount of work performed by your application. You can get started with custom metrics by using the PutMetricData API, our sample monitoring scripts for Windows and Linux, CloudWatch collectd plugin, as well as a number of applications and tools offered by AWS partners.

Q: What resolution can I get from a Custom Metric?

A custom metric can be one of the following:

* Standard resolution, with data having one-minute granularity
* High resolution, with data at a granularity of one second

By default, metrics are stored at 1-minute resolution in CloudWatch. You can define a metric as high-resolution by setting the StorageResolution parameter to 1 in the PutMetricData API request. If you do not set the optional StorageResolution parameter, then CloudWatch will default to storing the metrics at 1-minute resolution.

When you publish a high-resolution metric, CloudWatch stores it with a resolution of 1 second, and you can read and retrieve it with a period of 1 second, 5 seconds, 10 seconds, 30 seconds, or any multiple of 60 seconds.

Custom metrics follow the same retention schedule listed above.

Q: What metrics are available at high resolution?

Currently, only custom metrics that you publish to CloudWatch are available at high resolution. High-resolution custom metrics are stored in CloudWatch at 1-second resolution. High resolution is defined by the StorageResolution parameter in the PutMetricData API request, with a value of 1, and is not a required field. If you do not specify a value for the optional StorageResolution field, then CloudWatch will store the custom metric at 1-minute resolution by default.

Q: Are high-resolution custom metrics priced differently than regular custom metrics?

No, high-resolution custom metrics are priced in the same manner as standard 1-minute custom metrics.

Q: When would I use a Custom Metric over having my program emit a log to CloudWatch Logs?

You can monitor your own data using custom metrics, CloudWatch Logs, or both. You may want to use custom metrics if your data is not already produced in log format, for example operating system processes or performance measurements. Or, you may want to write your own application or script, or one provided by an AWS partner. If you want to store and save individual measurements along with additional detail, you may want to use CloudWatch Logs.

Q: What statistics can I view and graph in CloudWatch?

You can retrieve, graph, and set alarms on the following statistical values for Amazon CloudWatch metrics: Average, Sum, Minimum, Maximum, and Sample Count. Statistics can be computed for any time periods between 60-seconds and 1-day. For high-resolution custom metrics, statistics can be computed for time periods between 1-second and 3-hours.

Q: What is CloudWatch Application Insights for .NET and SQL Server?

Amazon CloudWatch Application Insights for .NET and SQL Server is a capability that you can use to easily monitor your .NET and SQL Server applications. It helps identify and set up key metrics and logs across your application resources and technology stack, i.e. database, web (IIS) and application servers, OS, load balancers, queues, etc. It constantly monitors these telemetry data to detect and correlate anomalies and errors, to notify you of any problems in your application. To aid in troubleshooting, it creates automatic dashboards to visualize problems it detects which includes correlated metric anomalies and log errors, along with additional insights to point you to their potential root-cause.

Q: What are the benefits of using CloudWatch Application Insights for .NET and SQL Server?

* Automatically recognize application metrics and logs: It scans your application resources, provides a list of recommended metrics and logs to monitor, and sets them up automatically, making it easier to set up monitoring for your applications.
* Intelligent problem detection: It uses built-in rules and machine learning algorithms to dynamically monitor and analyze symptoms of a problem across your application stack and detect application problems. It helps you reduce the overhead of dealing with individual metric spikes, or events, or log exceptions, and instead get notified on real problems, along with contextual information these problems.
* Faster troubleshooting: It assesses the detected problems to give you insights on them, such as the possible root-cause of the detected problem and list of metrics and logs impacted because of the problem. You can provide feedback on generated insights to make the problem detection engine specific to your use-case.

Q: How do I get started with monitoring using CloudWatch Application Insights for .NET and SQL Server?

On-board application: Specify the application you want to monitor by choosing the [AWS Resource Group](https://docs.aws.amazon.com/systems-manager/latest/userguide/systems-manager-resource-groups.html" \t "_blank) associated with it.

Identify application components: It analyzes your application resources to identify application components (standalone resources, or groups of related resources such as auto scaling groups and load balancer groups). You can also customize components by grouping resources for better insights and easy onboarding.

Enable monitoring: For your application components, you can specify the technology tier i.e. IIS front-end, .NET worker tier, etc. Based on your selection it provides a recommended set of metrics and logs that can be customized based on your needs. Once you save these “monitors”, Application Insights for .NET and SQL Server sets up CloudWatch to collect these on your behalf.

Once on-boarded, Application Insights for .NET and SQL Server uses a combination of pre-built rules and machine learning models to start identifying application problems. It creates automated dashboards on CloudWatch with the list of problems detected, and a detailed view for these problems along with related anomalies and errors.

## **Log monitoring**

Q: What log monitoring does Amazon CloudWatch provide?

CloudWatch Logs lets you monitor and troubleshoot your systems and applications using your existing system, application and custom log files.

With CloudWatch Logs, you can monitor your logs, in near real time, for specific phrases, values or patterns. For example, you could set an alarm on the number of errors that occur in your system logs or view graphs of latency of web requests from your application logs. You can then view the original log data to see the source of the problem. Log data can be stored and accessed for up to as long as you need in highly durable, low-cost storage so you don’t have to worry about filling up hard drives.

Q: What are Amazon CloudWatch Vended Logs?

Amazon CloudWatch Vended logs are logs that are natively published by AWS services on behalf of the customer. VPC Flow logs is the first Vended log type that will benefit from this tiered model. However, more AWS Service log types will be added to Vended Log type in the future.

Q: Is CloudWatch Logs available in all regions?

Please refer to [Regional Products and Services](https://aws.amazon.com/about-aws/global-infrastructure/regional-product-services/) for details of CloudWatch Logs service availability by region.

Q: How much does CloudWatch Logs cost?

Please see our [pricing page](https://aws.amazon.com/cloudwatch/pricing-cloudwatch.json/) for the latest information.

Q: What kinds of things can I do with my logs and Amazon CloudWatch?

CloudWatch Logs is capable of monitoring and storing your logs to help you better understand and operate your systems and applications. When you use CloudWatch Logs with your logs, your existing log data is used for monitoring, so no code change are required. Here are a two examples of what you can do with Amazon CloudWatch and your logs:

Real time Application and System Monitoring: You can use CloudWatch Logs to monitor applications and systems using log data in near real time. For example, CloudWatch Logs can track the number of errors that occur in your application logs and send you a notification whenever the rate of errors exceeds a threshold you specify. Amazon CloudWatch uses your log data for monitoring and consequently it doesn't involve any code changes from you.

Long Term Log Retention: You can use CloudWatch Logs to store your log data for as long as you need in highly durable and cost effective storage without worrying about hard drives running out of space. The CloudWatch Logs Agent makes it easy to quickly move both rotated and non rotated log files off of a host and into the log service. You can then access the raw log event data when you need it.

Q: What types of data can I send to Amazon CloudWatch Logs from my EC2 instances running Microsoft SQL Server and Microsoft Windows Server?

You can configure the EC2Config service to send a variety of data and log files to CloudWatch including: custom text logs, Event (Application, Custom, Security, System) logs, Event Tracing (ETW) logs, and Performance Counter (PCW) data. Learn more about the EC2Config service [here](http://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/UsingConfig_WinAMI.html).

Q: How frequently does the CloudWatch Logs Agent send data?

The CloudWatch Logs Agent will send log data every five seconds by default and is configurable by the user.

Q: What log formats does CloudWatch Logs support?

CloudWatch Logs can ingest, aggregate and monitor any text based common log data or JSON-formatted logs.

Q: What if I configure the CloudWatch Logs Agent to send non-text log data?

The CloudWatch Logs Agent will record an error in the event it has been configured to report non text log data. This error is recorded in the /var/logs/awslogs.log.

Q: How do I start monitoring my logs with CloudWatch Logs?

You can monitor log events as they are sent to CloudWatch Logs by creating Metric Filters. Metric Filters turn log data into Amazon CloudWatch Metrics for graphing or alarming. Metric Filters can be created in the Console or the CLI. Metric Filters search for and match terms, phrases or values in your log events. When a Metric Filter finds one of the terms, phrases or values in your log events, it counts it in an Amazon CloudWatch Metric that you choose. For example, you can create a Metric Filter to search for and count the occurrence of the word “Error” in your log events. Metric Filters can also extract values from space delimited log events, such as the latency of web requests. You can also use conditional operators and wildcards to create exact matches. The Amazon CloudWatch Console can help you test your patterns before creating Metric Filters.

Q: What is the syntax of Metric Filter patterns?

A Metric Filter pattern can contain search terms or a specification of your common log or JSON event format.

For example, if you want to search for the term Error, the pattern for the metric filter would just be the term Error. Multiple search terms can be included to search for multiple terms. For example, if you wanted to count events which contained the terms Error and Exception you would use the pattern Error Exception. If you wanted to match the term Error Exception exactly, you would put double quotes around the search term, "Error Exception". You can specify as many search terms as you like.

CloudWatch Logs can also be used to extract values from a log event in common log or JSON format. For example, you could track the bytes transferred from your Apache access logs. You can also use conditional operators and wildcards to match and extract the data you are interested in. To use the extraction feature of Metric Filters, log events must be space delimited and use a starting and ending double quote """, or, a starting square brace "[" and a closing square brace "]"square, to enclose fields. Alternatively, they can be JSON-formatted log events. For the full details of the syntax and examples, please see the [Developer Guide for Metric Filters](http://docs.aws.amazon.com/AmazonCloudWatch/latest/DeveloperGuide/WhatIsCloudWatchLogs.html).

Q: How do I know that a Metric Filter pattern I specified will match my log events?

CloudWatch Logs lets you test the Metric Filter patterns you want before you create a Metric Filter. You can test your patterns against your own log data that is already in CloudWatch Logs or you can supply your own log events to test. Testing your pattern will show you which log events matched the Metric Filter pattern and, if extracting values, what the extracted value is in the test data. Metric Filter testing is available for use in the console and the CLI.

Q: Can I use regular expressions with my log data?

Amazon CloudWatch Metric Filters does not support regular expressions. To process your log data with regular expressions, consider using [Amazon Kinesis](https://aws.amazon.com/kinesis/) and connect the stream with a regular expression processing engine.

## **Log management**

Q: How do I retrieve my log data?

You can retrieve any of your log data using the CloudWatch Logs console or through the CloudWatch Logs CLI. Log events are retrieved based on the Log Group, Log Stream and time with which they are associated. The CloudWatch Logs API for retrieving log events is GetLogEvents.

Q: How do I search my logs?

You can use the CLI to retrieve your log events and search through them using command line grep or similar search functions.

Q: How long does CloudWatch Logs store my log data?

You can store your log data in CloudWatch Logs for as long as you want. By default, CloudWatch Logs will store your log data indefinitely. You can change the retention for each Log Group at any time.

## **Log analytics**

Q: What permissions do I need to access Logs Insights?

To access Logs Insights, your IAM policy must include permissions for logs:DescribeLogGroups and logs:FilterLogEvents.

Q: What logs can I query with CloudWatch Logs Insights?

You can use Logs Insights to query all logs being sent to CloudWatch. Logs Insights automatically discovers the logs fields from logs from AWS services such as Lambda, CloudTrail, Route53, and VPC Flow Logs; and any application log that generates log events in JSON format. Additionally, for all log types, it generates 3 system fields @message, @logStream, and @timestamp for all logs sent to CloudWatch. @message contains the raw unparsed log event, @logStream contains the name of the source that generated the log event, and @timestamp contains the time at which the log event was added to CloudWatch.

Q: Which query language does CloudWatch Logs Insights support?

Logs Insights introduces a new purpose-built query language for log processing. The query language supports a few simple, but powerful query commands. You can write commands to retrieve one or more log fields, find log events that match one or more search criteria, aggregate your log data, and extract ephemeral fields from your text-based logs. The query language is easy to learn, and Logs Insights offers in-product help in the form of sample queries, command descriptions, and query auto-completion to help you get started. You can find additional details about the query language here.

Q: What are the service limits for CloudWatch Logs Insights?

The service limits are documented [here](https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/cloudwatch_limits_cwl.html" \t "_blank).

Q: What regions is CloudWatch Logs Insights available in?

Logs Insights is available in US West (Oregon), US West (N. California), US East (Ohio), US East (N. Virginia), Asia Pacific (Mumbai), Asia Pacific (Seoul), Asia Pacific (Singapore), Asia Pacific (Sydney), Asia Pacific (Tokyo), Canada (Central), EU (Frankfurt), EU (Ireland), EU (London), EU (Paris), South America (São Paulo).

Q: What type of queries does CloudWatch Logs Insights support?

You can write queries containing aggregations, filters, regular expressions, and text searches. You can also extract data from log events to create ephemeral fields, which can be further processed by the query language to help you access the information you are looking for. The query language supports string, numeric, and mathematical functions, such as concat, strlen, trim, log, and sqrt, among others. You can also use boolean and logical expressions, and aggregate functions such as min, max, sum, average, and percentile, among others. You can find additional details about the query language and supported functions [here](https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/AnalyzingLogData.html" \t "_blank).

Q: What query commands and functions can I use with CloudWatch Logs Insights?

You can find a list of query commands [here](https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/AnalyzingLogData.html" \t "_blank). You can find a list of supported functions [here](https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/AnalyzingLogData.html" \t "_blank).

Q: What data visualizations can I use with CloudWatch Logs Insights?

You can use visualizations to identify trends and patterns that occur over time within your logs. Logs Insights supports visualizing data using line charts and stacked area charts. It generates visualizations for all queries containing one or more aggregate functions, where data is grouped over a time interval specified using the bin() function. You can find additional details about visualizing timeseries data [here](https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/AnalyzingLogData.html" \t "_blank).

Q: Can I use regular expressions with CloudWatch Logs Insights?

You can use Java-style regular expressions with Logs Insights. Regular expressions can be used in the filter command. You can find examples of queries with regular expressions using the in-product help or [here](https://docs.aws.amazon.com/AmazonCloudWatch/latest/logs/AnalyzingLogData.html" \t "_blank).

Q: How do I escape special characters with CloudWatch Logs Insights queries?

You can use backticks to escape special characters. Log field names that contain characters other than alphanumeric characters, @, and . require escaping with backticks.

Q: Why do certain log fields have a “@” sign and others don’t?

System fields generated by Logs Insights begin with @. Logs Insights currently generates 3 system fields @message which contains the raw, unparsed log event as sent to CloudWatch, @logStream which contains the name of the source that generated the log event, and @timestamp which contains the time when the log event was added to CloudWatch.

Q: Can I query historical logs with CloudWatch Logs Insights?

Logs Insights enables you to query log data that was added to CloudWatch Logs on or after November 5, 2018.

Q: Can I search for log events from a specific log stream?

You can search for log events from a specific log stream by adding the query command  
filter @logStream = "log\_stream\_name" to your log query.

Q: Today I use an AWS Partner ISV solution to analyze my logs from CloudWatch. What does CloudWatch Logs Insights change for me?

CloudWatch Logs already supports integration options with other AWS Services such as Amazon Kinesis, Amazon Kinesis Data Firehose, Amazon Elasticsearch and AWS Partner ISV solutions such as Splunk, Sumo Logic, and DataDog, among others, to provide you with choice and flexibility across all environments, for your custom log processing, enrichment, analytics, and visualization needs. In addition, the query capabilities of CloudWatch Logs Insights are available for programmatic access through the AWS SDK, to facilitate AWS ISV Partners to build deeper integrations, advanced analytics, and additional value on top of CloudWatch Logs Insights.

Q: How will I benefit from having access to query capabilities of CloudWatch Logs Insights through an AWS ISV Partner solution?

ISV Partner integrations with CloudWatch Logs Insights enable you to bring in your log data into one place and have the ability to analyze using the tools and frameworks of your choice in a high performance, cost-effective way, without having to move large amounts of data. It also provides you with faster access to your logs by removing the associated data transfer latencies and eliminates the operational complexities of configuring and maintaining certain data transfers.

## **Alarms**

Q: What types of CloudWatch Alarms can be created?

You can create an alarm to monitor any Amazon CloudWatch metric in your account. For example, you can create alarms on an Amazon EC2 instance CPU utilization, Amazon ELB request latency, Amazon DynamoDB table throughput, Amazon SQS queue length, or even the charges on your AWS bill.

You can also create an alarm on custom metrics that are specific to your custom applications or infrastructure. If the custom metric is a high-resolution metric, you have the option of creating high-resolution alarms that alert as soon as 10-second or 30-second periods.

Please reference the [CloudWatch pricing page](https://aws.amazon.com/cloudwatch/pricing/) to learn more.

Q: What actions can I take from a CloudWatch Alarm?

When you create an alarm, you can configure it to perform one or more automated actions when the metric you chose to monitor exceeds a threshold you define. For example, you can set an alarm that sends you an email, publishes to an SQS queue, stops or terminates an Amazon EC2 instance, or executes an Auto Scaling policy. Since Amazon CloudWatch alarms are integrated with Amazon Simple Notification Service, you can also use any notification type supported by SNS.

Q: What thresholds can I set to trigger a CloudWatch Alarm?

When you create an alarm, you first choose the Amazon CloudWatch metric you want it to monitor. Next, you choose the evaluation period (e.g., five minutes or one hour) and a statistical value to measure (e.g., Average or Maximum). To set a threshold, set a target value and choose whether the alarm will trigger when the value is greater than (>), greater than or equal to (>=), less than (<), or less than or equal to (<=) that value.

Q: My CloudWatch Alarm is constantly in the Alarm state, what did I do wrong?

Alarms continue to evaluate metrics against your chosen threshold, even after they have already triggered. This allows you to view its current up-to-date state at any time. You may notice that one of your alarms stays in the ALARM state for a long time. If your metric value is still in breach of your threshold, the alarm will remain in the ALARM state until it no longer breaches the threshold. This is normal behavior. If you want your alarm to treat this new level as OK, you can adjust the alarm threshold accordingly.

Q: How long can I view my Alarm history?

Alarm history is available for 14 days. To view your alarm history, log in to CloudWatch in the AWS Management Console, choose Alarms from the menu at left, select your alarm, and click the History tab in the lower panel. There you will find a history of any state changes to the alarm as well as any modifications to the alarm configuration.

## **Dashboards**

Q: What is CloudWatch Dashboards?

Amazon CloudWatch Dashboards allow you to create, customize, interact with, and save graphs of AWS resources and custom metrics.

Q: How do I get started with CloudWatch Dashboards?

To get started, visit the [Amazon CloudWatch Console](https://console.aws.amazon.com/cloudwatch) and select “Dashboards”. Click the “Create Dashboard” button. You can also copy the desired view from Automatic Dashboards by clicking on Options -> “Add to Dashboard”.

Q: What are the advantages of Automatic Dashboards?

Automatic Dashboards are pre-built with AWS service recommended best practices, remain resource aware, and dynamically update to reflect the latest state of important performance metrics. You can now filter and troubleshoot to a specific view without adding additional code to reflect the latest state of your AWS resources. Once you have identified the root cause of a performance issue, you can quickly act by going directly to the AWS resource.

Q: Do the dashboards support auto refresh?

Yes. Dashboards will auto refresh while you have them open.

Q: Can I share my dashboard?

Yes, a dashboard is available to anyone with the correct permissions for the account with the dashboard.

## **Events**

Q: What is CloudWatch Events?

Amazon CloudWatch Events (CWE) is a stream of system events describing changes in your AWS resources. The events stream augments the existing CloudWatch Metrics and Logs streams to provide a more complete picture of the health and state of your applications. You write declarative rules to associate events of interest with automated actions to be taken.

Q: What services emit CloudWatch Events?

Currently, Amazon EC2, Auto Scaling, and AWS CloudTrail are supported. Via AWS CloudTrail, mutating API calls (i.e., all calls except Describe\*, List\*, and Get\*) across all services are visible in CloudWatch Events.

Q: What can I do once an event is received?

When an event matches a rule you've created in the system, you can automatically invoke an AWS Lambda function, relay the event to an Amazon Kinesis stream, notify an Amazon SNS topic, or invoke a built-in workflow.

Q: Can I generate my own events?

Yes. Your applications can emit custom events by using the PutEvents API, with a payload uniquely suited to your needs.

Q: Can I do things on a fixed schedule?

CloudWatch Events is able to generate events on a schedule you set by using the popular Unix cron syntax. By monitoring for these events, you can implement a scheduled application.

Q: What is the difference between CloudWatch Events and AWS CloudTrail?

CloudWatch Events is a near real time stream of system events that describe changes to your AWS resources. With CloudWatch Events, you can define rules to monitor for specific events and perform actions in an automated manner. AWS CloudTrail is a service that records API calls for your AWS account and delivers log files containing API calls to your Amazon S3 bucket or a CloudWatch Logs log group. With AWS CloudTrail, you can look up API activity history related to creation, deletion and modification of AWS resources and troubleshoot operational or security issues.

Q: What is the difference between CloudWatch Events and AWS Config?

AWS Config is a fully managed service that provides you with an AWS resource inventory, configuration history, and configuration change notifications to enable security and governance. Config rules help you determine whether configuration changes are compliant. CloudWatch Events is for reacting in near real time to resource state changes. It doesn’t render a verdict on whether the changes comply with policy or give detailed history like Config/Config Rules do. It is a general purpose event stream.