WHITEPAPER

Monitoring AWS beyond CloudWatch

By Mick England, DevOps professional in the Boston area

Amazon CloudWatch is a powerful tool that can monitor Amazon Web Services (AWS) in real-time. CloudWatch can collect and track metrics, including ones that you define to measure the performance of your application.

This paper will examine some of the beneficial features provided by Amazon CloudWatch, as well its limitations. It will explain why and how to go beyond CloudWatch to provide full visibility into the health and performance of your AWS resources.

CloudWatch Use Cases

CloudWatch was first released into public beta in May 2009. In December the following year, Amazon introduced CloudWatch Alarms which allowed for up to 5000 alarms per account. These alarms could be used to cause a Simple Notification Services (SNS) or auto scaling event. Though announced as a monitoring solution, CloudWatch was clearly intended to do much more than monitor your resources. While CloudWatch does provide the traditional function of a monitoring solution – detection and alerting of system or application problems – it is as much concerned with management as it is with monitoring. For instance, CloudWatch is the mechanisms through which auto scaling is managed and based on criteria you define, CloudWatch can start, stop or terminate EC2 instances.

The Amazon CloudWatch Developer Guide gives a good overview of the services and explains certain key concepts. CloudWatch is described as "basically a metrics repository". Metrics can come into CloudWatch from multiple sources (e.g., AWS resources that use CloudWatch or your own custom data). These metrics can be used to calculate statistics and the data can be presented graphically through the CloudWatch console. With the introduction of CloudWatch Logs in July 2014 it is now possible to send system and application logs into CloudWatch.

In addition to data collection, CloudWatch has alarms. These can be used to perform actions on Amazon resources, such as stop, start or terminate an EC2 instance when certain criteria are met. These alarms can be used to manage Auto Scaling Amazon and SNS actions you define. In January 2016 Amazon announced CloudWatch Events which it described as "a stream of events describing changes to your AWS resources, such as an Amazon Elastic Compute Cloud (EC2) instance coming into service or failing a system health check, the Auto Scaling service adding instances to a deployment, or suspicious API call patterns being logged in AWS CloudTrail." A post by Jeff Bar gives some indication of the future of CloudWatch Events when he writes: "Like many emerging AWS services, we are launching CloudWatch Events with an initial set of features (and a lot of infrastructure behind the scenes) and some really big plans, including AWS CloudFormation support". Cloud Formation is Amazon's template based orchestration tool.

Monitoring AWS beyond CloudWatch

By Mick England, DevOps professional in the Boston area

CloudWatch Monitoring Limitations

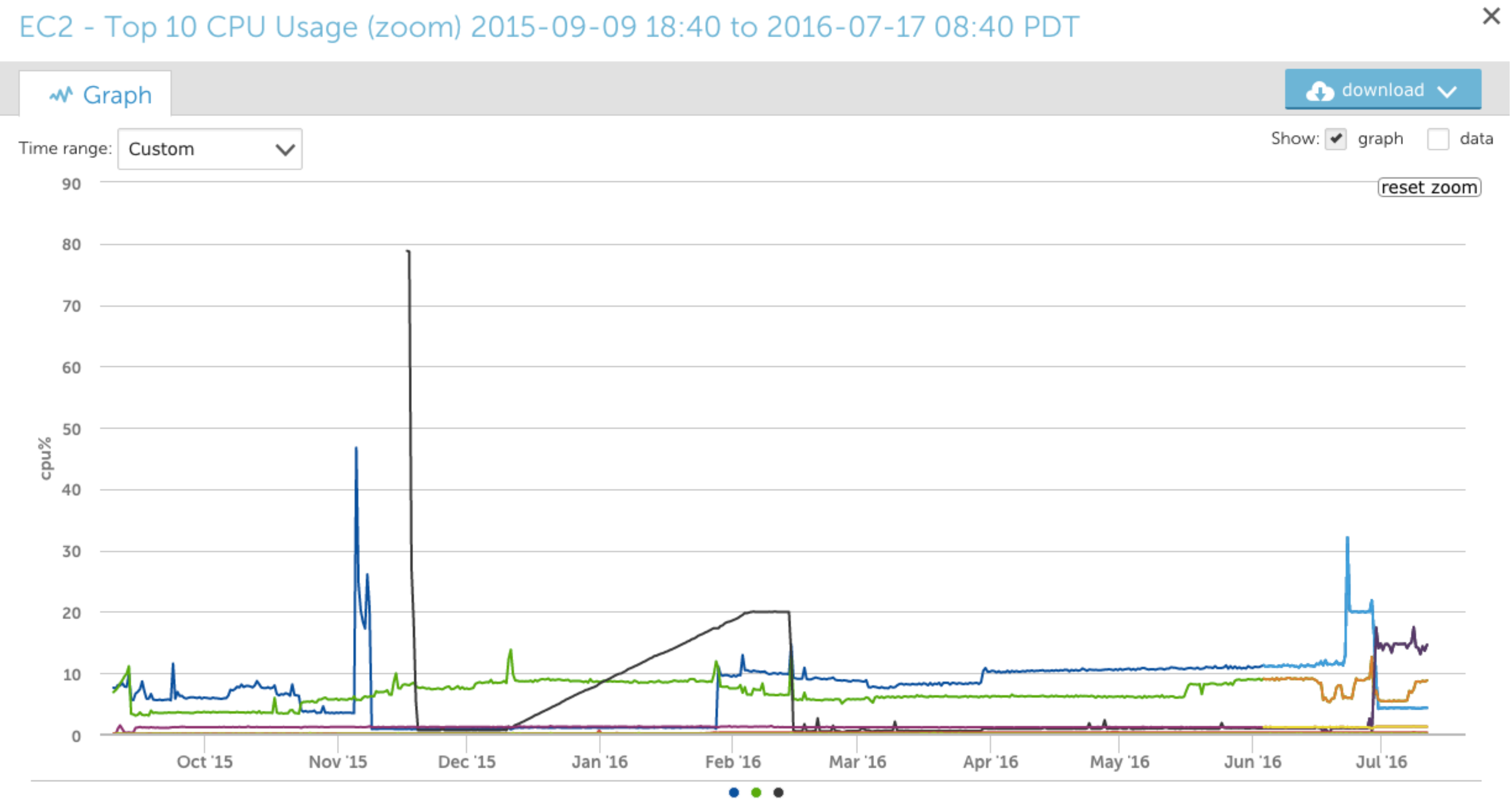
As a monitoring tool, CloudWatch has many limitations. We will examine some of the best known of these below, as well as address ways to go beyond these limitations in implementing your full stack monitoring solution.

1. Limited Data Retention

The first limitation of CloudWatch is the restriction to two weeks of metrics data. This time period is fine for real-time monitoring, but often you will want to track metrics over longer periods of time. For instance, if you have a monthly major release cycle it is good practice to compare performance metrics of different releases. If you are in a seasonal business such as retail, you may want to go back a year or more to see how you did on Black Friday last year. CloudWatch alone can't do this.

The most basic work around is to use the GetMetricStatistics API to pull down data for external processing. Additionally, the popular open source log centralization tool Logstash has a plugin to allow CloudWatch metrics to be pulled from Amazon into your ELK (Elastic Search, Logstash, Kibana) stack and analyzed alongside other log data.

Modern monitoring tools, like LogicMonitor, capture CloudWatch metrics via the CloudWatch API and display them in the same platform as your other infrastructure. LogicMonitor also stores historical metrics for up to two years with one minute granularity.



LogicMonitor dashboard widget showing granular historical data

2. Limited Metrics Available

Amazon has a free tier of CloudWatch which provides basic monitoring metrics for EC2 instances at no charge. This basic monitoring collections data at a frequency of five minutes, which may not be frequent enough for managing auto scaling in a critical environment. Also included in the free tier are all metrics for Amazon EBS volumes, Elastic Load Balancers, and Amazon RDS DB instances. Customers receive 10 metrics that can be used for detailed monitoring of EC2 instances at 1-minute intervals, custom metrics or CloudWatch Logs. The free tier includes 10 alarms, and 1 million API requests each month at no additional charge, as well as 5 GB of data ingestion and 5 GB of archived storage per month.

Reality is that beyond the free tier, costs can quickly start to add up. Prices vary by region, but in the popular US East (N. Virginia) region, detailed monitoring of EC2 instances costs $3.50 per instance per month. Custom metrics are $0.50 per metric and logs are $0.50 per GB of ingested data, with the usual charges for data in and out of AWS.

Monitoring AWS beyond CloudWatch

By Mick England, DevOps professional in the Boston area

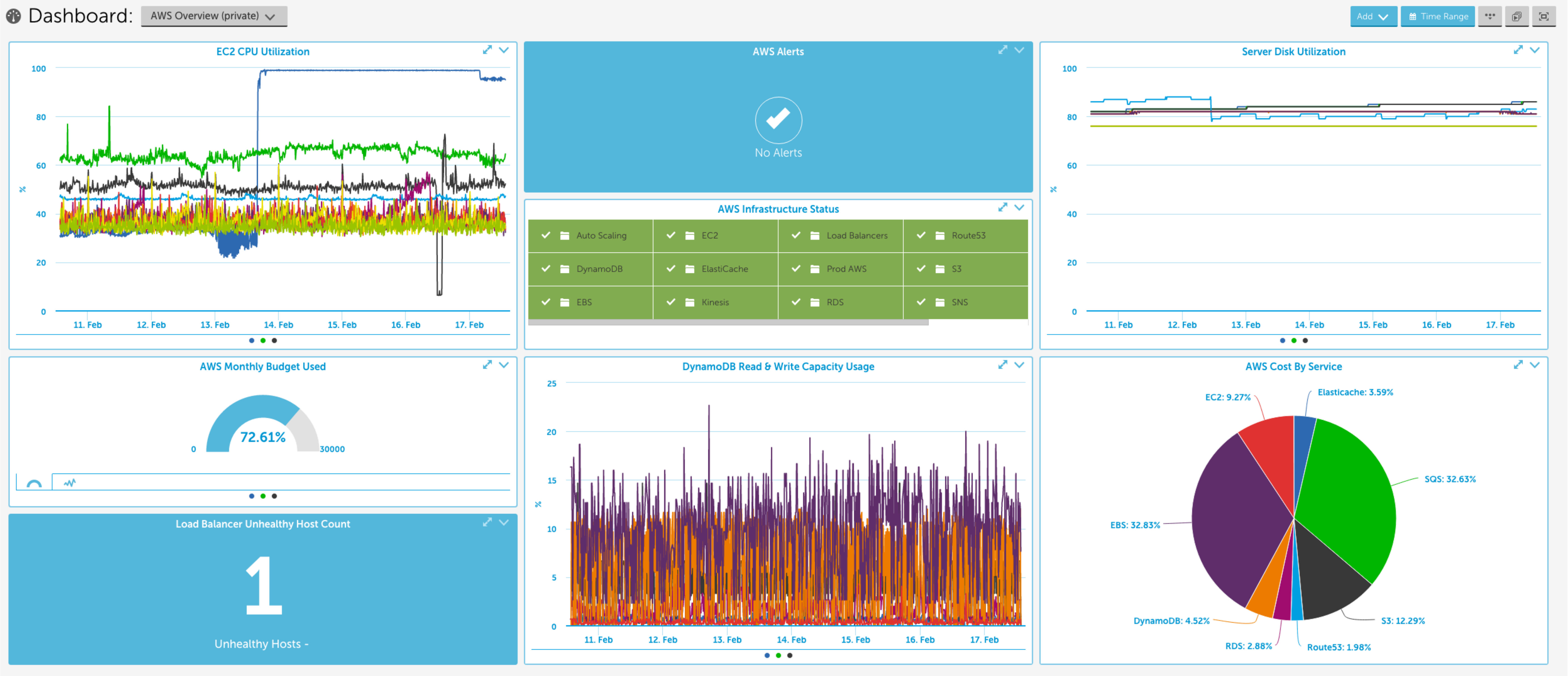
Amazon CloudWatch has broad coverage of the most used services provided in AWS. However, looking at the set of metrics provided will show some surprising limitations. For example, CloudWatch does not collect memory metrics for EC2 instances. The reason for this is that CloudWatch has no visibility into what is happening inside an EC2 instance. While not providing memory usage for an individual instance, CloudWatch can see aggregated memory usage for an auto scaling group, making it possible to auto-scale based on memory usage thresholds if desired.

A monitoring platform like LogicMonitor extends CloudWatch monitoring by also allowing users to monitor AWS resources at the SDK level through a library of monitoring templates. Customers can use LogicMonitor to report on the performance and availability of AWS resources where it matters - from the perspective of the servers accessing them.

4. Limited Dashboards

When CloudWatch was initially launched there was no ability to create custom dashboards. This was fixed in October 2015 with the release of CloudWatch Dashboards. These allow users to create text-based or graphical widgets, add custom text annotations and links to graphs, change the time range, resize and reorganize widgets, and reuse and share dashboards. While this is a significant addition, it is not as mature as the dashboards available within other monitoring tools. Though users can present AWS data alongside infrastructure in data centers through the use of CloudWatch Logs, this could become a very expensive solution.

In a monitoring platform like LogicMonitor, users can easily build dashboards to view the performance of their datacenter infrastructure alongside their AWS resources.



AWS dashboard in LogicMonitor

CloudWatch is not Full Stack Monitoring

The lack of visibility into what is happening inside an EC2 instance imposes severe restrictions on CloudWatch as a monitoring solution. We could of course write custom applications to query memory usage and push the data into CloudWatch. We would then need to do the same for services such as Apache, Tomcat and anything else running on top of EC2. At this point we would be writing our own monitoring solution with CloudWatch as a very limited data repository.

Monitoring AWS beyond CloudWatch

By Mick England, DevOps professional in the Boston area

LogicMonitor, a modern monitoring solution, provides a Collector application which is installed on EC2 instances and gathers additional information which is either pushed to CloudWatch or, more commonly, sent directly to the monitoring solution's data store. In addition, monitoring solutions such as LogicMonitor will use other protocols such as Simple Network Monitoring Protocol (SNMP), https, JMX or SQL to gather important information about how services are performing.

While CloudWatch can give you insight into your application if you give it the right metrics, defining and developing these can be troublesome. There are a host of tools in the Application Performance Monitoring (APM) space designed specifically for this purpose as well as monitoring tools, such as LogicMonitor, that allow you to easily monitor your hybrid infrastructure.

Conclusion

CloudWatch gives some valuable visibility into your AWS account and the resources running there. In any system of reasonable size and complexity, however, the limitations of CloudWatch will quickly become apparent.

CloudWatch should be considered a useful tool in your monitoring strategy, but it is not the only tool you will need. This is particularly true in hybrid environments where it is necessary to have a single view into data centers and cloud. Even in environments wholly hosted in AWS, it is necessary to take a full stack approach to monitoring. As always, choosing the right tools for the job is essential. Some considerations for full stack monitoring include the following:

Network Monitoring: The monitoring of underlying network infrastructure, including connections from datacenter to cloud

Centralized Logging: Full visibility into system logs from both cloud and data center

Metrics: Useful metrics from both CloudWatch and custom application metrics in the data center

Systems Monitoring: The monitoring of hardware and system performance up to and included the OS

Services Monitoring: Third party applications such as Apache HTTPD or Tomcat

Application Monitoring: Monitoring your own code base

I have yet to find any tool that can do all of the above well so it is no surprise that there is a need to go beyond CloudWatch. What it does, it does well, but like all tools it has its limitations.

This content is brought to you by LogicMonitor, the automated SaaS performance monitoring platform that provides IT Ops teams with end-to-end visibility and actionable metrics to manage today's sophisticated on-premise, hybrid, and cloud infrastructures. Find out more information about LogicMonitor's AWS monitoring here or sign up for a free 14 day trial today.