Monitoring in AWS

Overview

> *With CloudWatch we can track and monitor a lot of metrics across many AWS's products and set alarms when certain conditions are met. When these alarms are triggered, they can notify us or automate actions such as shrinking or increasing an AutoScaling Group capacity. CloudWatch knows a lot about our EC2 instances' at the hardware level but it lacks the software's point of view. In this post we will explain how to use CloudWatch to monitor important resources it can't monitor by default.*

How standard monitoring works in EC2

When it comes to monitoring EC2 instances, we have to keep in mind that an instance in the cloud is not an actual single computer, but a virtual machine running alongside some siblings on a bigger host, which runs the virtualization solution, or hypervisor. Specifically, AWS uses a customized version of Xen Hypervisor.

CloudWatch relies on the information provided by this hypervisor, which can only see the most hardware-sided part of the instance's status, including CPU usage (but not load), total memory size (but not memory usage), number of I/O operations on the hard disks (but not it's partition layout and space usage) and network traffic (but not the processes generating it).

While this can be seen as a shortcoming on the hypervisor's part, it's actually very convenient in terms of security and performance, otherwise the hypervisor would be an all-seeing eye, with more powers than the root user itself.

How to monitor key elements of an EC2 instance

By default, CloudWatch can't see what the hypervisor can't see. Luckily, CloudWatch accepts inputs from sources other than the hypervisor. This is what enables CloudWatch to monitor RDS's instances details (such as replica lag) or the depth of an SQS queue, and it's available to the end user under the label "Custom metrics". In order to use custom metrics to monitor machine resources (disk space, memory, etc.) we need to:

1. Grant permissions to the VM put data in cloud watch
2. Install perquisite software on the VM
3. Install monitoring scripts
4. Cron the monitoring scripts

For our purposes we will assume the OS is ubuntu

Grant permissions to the VM put data in cloud watch

We need to navigate to the IAM console, Roles section. There we then need to select our role and press the "Attach Role Policy" button in the permissions tab. In the new window, we choose "Custom Policy", press the "Select" button, assign it a name of our choice and paste this policy in:

```
{
  "Statement": [
    {
      "Action": [
        "cloudwatch:GetMetricStatistics",
        "cloudwatch:ListMetrics",
        "cloudwatch:PutMetricData",
        "ec2:DescribeTags"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Now all our instance with the defined role can talk to cloudwatch.

Install perquisite software on the VM

sudo apt-get install unzip libwww-perl libcrypt-ssleay-perl

sudo apt-get install libswitch-perl

Install monitoring scripts

1. *wget http://ec2-downloads.s3.amazonaws.com/cloudwatch-samples/CloudWatchMonitoringScripts-v1.1.0.zip*
2. *unzip CloudWatchMonitoringScripts-v1.1.0.zip*
3. *rm CloudWatchMonitoringScripts-v1.1.0.zip*

Cron the monitoring script

1. *crontab -e*
2. **/5 * * * * ~/aws-scripts-mon/mon-put-instance-data.pl --mem-util --disk-space-util --disk-path=/ --from-cron*