



Monitoring Amazon Web Services

Amazon Web Services PowerPack version 122

Table of Contents

Introduction	5
What is AWS?	6
What is an AWS Region?	6
What is an AWS Zone?	7
What Does the Amazon Web Services PowerPack Monitor?	7
Installing the Amazon Web Services PowerPack	10
Controlling What is Discovered by the PowerPack	12
Configuring AWS for Monitoring Regions with AWS Config Enabled	12
Configuring AWS for Monitoring Regions with AWS CloudWatch	13
The Regions Header in the SOAP/XML Credential	13
Using IAM Permissions to Restrict SL1 Access to Specific Regions and Services	15
Example 1: One Region	15
Example 2: Multiple Regions	16
Configuring AWS for Monitoring Regions Using CloudWatch Namespaces	17
Configuration	19
Configuring AWS to Report Billing Metrics	20
Filtering EC2 Instances By Tag	22
Filtering EC2 Instances by Tag in the SL1 Classic User Interface	24
Automatic SL1 Organization Creation	25
Monitoring Consolidated Billing Accounts	26
ScienceLogic Events and AWS Alarms	26
Using a Proxy Server	26
Configuring "AWS: Lambda Service Discovery"	28
Configuring "AWS: Lambda Function Qualified Discovery"	31
Configuring AWS Integration with Docker	35
Configuring AWS Integration with Kubernetes	35
Enabling the Prometheus Metrics Server	37
Define the Cluster Role	37
Define the ClusterRoleBinding	39
Map the IAM User or Role to the Kubernetes RBAC Role	40
Example 1	40
Example 2	41
Amazon API Throttling Events	42
Support for AWS China Regions	42
Support for AWS GovCloud Regions	42
Minimum Permissions	44
Minimum Permissions Needed to Monitor Your AWS Accounts	44
AWS Discovery	51
Discovering Amazon Web Services	52
Manual Discovery	53
Configuring a User in AWS	53
Creating the SOAP/XML Credential for AWS	57
Creating the SOAP/XML Credential for AWS in the SL1 Classic User Interface	59
Creating an AWS Virtual Device for Discovery in the SL1 Classic User Interface	62
Aligning the Discovery Dynamic Application	62
Aligning the Discovery Dynamic Application in the SL1 Classic User Interface	64
Automated Discovery Using AssumeRole with a Single IAM Key from the AWS Master Account	65
Configure a User in the Master Billing Account	66
Create a Role in Each Account	67
Configure the SL1 Credential	68

Configure the SL1 Credential in the SL1 Classic User Interface	71
Create and Run the Discovery Session	73
Create and Run the Discovery Session in the SL1 Classic User Interface	75
Manually Creating the Organization and Aligning Dynamic Applications	77
Automated Discovery when the Data Collector Runs as an EC2 Instance	78
Create a Role in the Master Billing Account	78
Create an AWS Role in the Account your Data Collector is In	80
Create a Role in Each Account	81
Configuring the Credential to Discover AWS on an EC2 Collector	82
Configuring the Credential to Discover AWS on an EC2 Collector in the SL1 Classic User Interface ..	83
Create and Run the Discovery Session	85
Create and Run the Discovery Session in the SL1 Classic User Interface	88
AWS Guided Discovery	89
Defining an AWS Assume Role Credential	91
Defining an AWS EC2 Credential	93
Defining an AWS IAM Credential	96
Completing the Discovery Session	98
The AWS Credential Test and Viewing Component Devices	100
Testing the AWS Credential	100
Testing the AWS Credential in the SL1 Classic User Interface	102
Viewing AWS Component Devices	103
Relationships Between Component Devices	105
Vanishing Component Devices	107
Configuring Inbound CloudWatch Alarms	108
CloudWatch Alarm Event Policies	108
Creating Custom CloudWatch Metrics	110
Configuring CloudWatch to Send Alarms for a Metric	113
Enabling Custom Metrics Collection in SL1	115
Configuring the "AWS: CloudWatch Alarms Performance" Dynamic Application	115
Enabling CloudWatch Alarm Events in SL1	118
Preserving CloudWatch Alarm Event Changes	119
Reports	120
AWS Billing Report	121
AWS Inventory Report	123
AWS Running Config Report	125
Dashboards	127
Installing the Amazon Web Services: Dashboards PowerPack	127
AWS Account Billing Dashboard	128
AWS Health Status Dashboard	129
Configuring the AWS Dashboards	130
AWS Service Instance Performance Dashboards	131
Run Book Actions and Automations	133
About the Run Book Actions and Automations	134
Disabling EC2 and EBS Instances by EC2 Tag	135
Modifying the Parameters of the Automation Actions	136
Enabling the Component Device Record Created Event Policy	137
Enabling the Automation Policies	137
Preserving Automation Changes	137
Discovering EC2 Instances by Public or Private IP Address	138
Modifying the Parameters of the Automation Actions	139
Enabling the Component Device Record Created Event Policy	141
Enabling the Device Record Created Event Policy	142

Enabling the Automation Policies	142
Preserving Automation Changes	143
Aligning AWS Regions to the AWS Region Device Class	143
Vanishing Terminated or Terminating EC2 Instances	144
Enabling the Automation Policies	145
Preserving Automation Changes	145
Key Metrics Collected by the PowerPack	146
AWS API Gateway Service	147
AWS Application ELB Service	148
AWS Auto Scale Service	150
AWS CloudFront Service	152
AWS CloudTrail Service	157
AWS CloudWatch Service	158
AWS DDB Service	159
AWS Direct Connect Service	160
AWS DynamoDB Service	165
AWS EBS Service	166
AWS EC2 Service	168
AWS ECS Service	175
AWS EFS Service	182
AWS EKS Service	184
AWS Elastic Beanstalk Service	186
AWS ElastiCache Service	190
AWS ELB Service	196
AWS EMR Service	200
AWS Glacier Service	204
AWS IoT Service	205
AWS KMS Service	206
AWS Lambda Service	207
AWS LightSail Service	213
AWS Network ELB Service	216
AWS OpsWorks Service	217
AWS RDS Service	219
AWS Redshift Service	230
AWS Route 53 Service	233
AWS S3 Service	235
AWS SES Service	237
AWS Shield Standard Service	238
AWS SNS Service	239
AWS SQS Service	241
AWS Storage Gateway Service	243
AWS STS Service	245
AWS Transit Gateway Service	245
AWS VPC Service	247
AWS WAF Global Service	254
AWS Workspaces Service	256

Chapter

1

Introduction

Overview

This manual describes how to monitor Amazon Web Services (AWS) in SL1 using the *Amazon Web Services PowerPack*. It also describes the reports you can generate and the dashboards you can view after you collect data from AWS, as well as the Run Book Action and Automation policies you can use to automate certain aspects of monitoring AWS.

Use the following menu options to navigate the SL1 user interface:

- To view a pop-out list of menu options, click the menu icon (.
- To view a page containing all the menu options, click the Advanced menu icon (.

The following sections provide an overview of Amazon Web Services and the *Amazon Web Services PowerPack*:

What is AWS?	6
What is an AWS Region?	6
What is an AWS Zone?	7
What Does the Amazon Web Services PowerPack Monitor?	7
Installing the Amazon Web Services PowerPack	10

NOTE: For more information about setting up a SL1 appliance on an Amazon Web Services EC2 instance, see the *Installation and Initial Configuration* manual.

NOTE: For more information about setting up an AWS Elasticsearch, Logstash, and Kibana (ELK) stack, see the *Monitoring AWS ELK Stacks* manual.

NOTE: ScienceLogic provides this documentation for the convenience of ScienceLogic customers. Some of the configuration information contained herein pertains to third-party vendor software that is subject to change without notice to ScienceLogic. ScienceLogic makes every attempt to maintain accurate technical information and cannot be held responsible for defects or changes in third-party vendor software. There is no written or implied guarantee that information contained herein will work for all third-party variants. See the End User License Agreement (EULA) for more information.

What is AWS?

Amazon Web Services is Amazon's "Infrastructure as a Service" offering. AWS includes multiple products (called **Services**) including compute, DNS, networking, content delivery, analytics, storage, and database services, among many others.

What is an AWS Region?

An AWS region is a geographical area made up of availability zones located within that region. Each zone may have multiple data centers. Regions have a canonical naming scheme of:

country/continent-direction-number

For example, the 'us-east-1' region is located in the United States, on the east coast, and it is the #1 data center in that region.

AWS regions are also commonly referred to by the city or state in which the data center is located. For example, us-west-2 is commonly referred to as "Oregon", ap-northeast-1 is commonly referred to as "Tokyo", etc.

The Dynamic Applications in the *Amazon Web Services PowerPack* create a "region" component device for each discovered region. The component devices for regions include both the region name and city/state description. For example, the Dynamic Applications might discover a component device called "Oregon: us-west-2". Component devices that represent region-specific AWS services reside under the appropriate "region" component device and appropriate "zone" component device.

NOTE: For more information about AWS regions, see <https://docs.amazonaws.cn/en-us/general/latest/gr/rande.html>.

What is an AWS Zone?

All instances of an AWS service reside in one or more Zones. A zone is a physical network and power partition (air-gap firewall) within a regional data center. Some AWS instances, like EC2 instances, are in a single zone. Other AWS instances, like an SNS queue, exist in all zones simultaneously.

The AWS naming convention for a zone is:

`region[a-z]`

For example, zone 'a' for the region 'us-east-1' is named 'us-east-1a'.

When a user deploys a service instance, the user can specify a "zone preference", but the final zone for that service instance is decided by AWS, not the user.

The Dynamic Applications in the *Amazon Web Services PowerPack* create a "zone" component device for each discovered zone.

AWS services with a specific zone affinity reside under the appropriate zone component device. For example, the Dynamic Applications in the PowerPack might discover the zone "us-west-1b" and create a component device called "us-west-1b".

AWS services that are specific to a zone reside under the appropriate "region" component device and appropriate "zone" component device. The Dynamic Applications in the PowerPack create a "multi-zoned" component device for services that are inherently zone agnostic such as the Simple Queue Service (SQS).

Component devices that represent Zones are a named container with no associated performance metrics.

What Does the Amazon Web Services PowerPack Monitor?

To collect data from Amazon Web Services, the ScienceLogic Data Collector or All-In-One Appliance connects via HTTPS to the URLs listed in the following AWS document:

<http://docs.aws.amazon.com/general/latest/gr/rande.html>.

The *Amazon Web Services PowerPack* includes Dynamic Applications that can monitor performance metrics and collect configuration data for the following AWS Services and components:

- API Gateways
- Aurora
- AutoScale
- CloudFront
- CloudTrail
- CloudWatch
- Direct Connect

- DynamoDB (DDB)
- ElastiCache
- Elastic Beanstalk
- Elastic Block Store (EBS)
- Elastic Compute Cloud (EC2)
- Elastic Container Services (ECS)
- Elastic File System (EFS)
- Elastic Kubernetes Service (EKS)
- Elastic Load Balancers (ELB)
- Elastic Map Reduce (EMR)
- Glacier
- IoT
- Key Management Service (KMS)
- Lambda
- Lightsail
- OpsWorks
- RedShift
- Relational Data Store (RDS)
- Route53
- Security Groups
- Shield
- Simple Email Service (SES)
- Simple Notification Service (SNS)
- Simple Queue Service (SQS)
- Simple Storage Service (S3)
- Storage Gateways (ASG)
- Storage Gateway Volumes
- Transit Gateways
- Virtual Private Cloud Service (VPC)
- Virtual Private Networks (VPN)
- Web Application Firewall (WAF)
- WorkSpaces

NOTE: The following services are not monitored for GovCloud accounts:

- API Gateway private integrations
- CloudFront
- Lightsail
- OpsWorks
- Replica Lambda functions
- Shield
- Web Application Firewall

NOTE: Not all AWS services are supported by all AWS regions. For more information about which AWS services are supported by which AWS regions, see <https://aws.amazon.com/about-aws/global-infrastructure/regional-product-services>.

NOTE: To monitor performance metrics for an AutoScale group, you must activate detailed instance monitoring for that group. For instructions on how to perform this task, see <http://docs.aws.amazon.com/AutoScaling/latest/DeveloperGuide/as-instance-monitoring.html>.

NOTE: When monitoring EC2-backed ECS clusters, you can optionally use the *Docker* PowerPack to collect container information in addition to what the AWS API provides for the ECS service. For more information, see the section on [Configuring AWS Integration with Docker](#).

NOTE: To monitor Lambda services, you must first configure some of the Dynamic Applications in the *Amazon Web Services* PowerPack prior to discovery. For more information, see the [Configuring "AWS Lambda Service Discovery"](#) and [Configuring "AWS Lambda Function Qualified Discovery"](#) sections.

The Dynamic Applications in the PowerPack also monitor:

- The general health of each AWS service
- Current billing metrics for each service aligned with the account
- Custom, application-specific performance metrics configured on the account
- The state of any AWS Alarms set on metrics in Cloudwatch

In addition to Dynamic Applications, the PowerPack includes the following features:

- Event Policies and corresponding alerts that are triggered when AWS component devices meet certain status criteria

- Device Classes for each of the AWS component devices monitored
- Sample Credentials for discovering AWS component devices
- Reports and dashboards that display information about AWS instances and component devices
- Run Book Action and Automation policies that can automate certain AWS monitoring processes

NOTE: To view Amazon Web Services dashboards, you must first install the *Amazon Web Services: Dashboards* PowerPack. For more information, see the [AWS Dashboards](#) chapter.

Installing the Amazon Web Services PowerPack

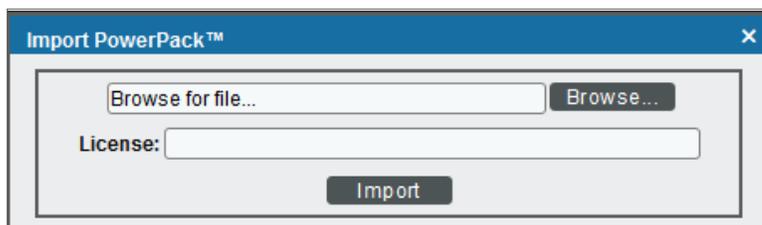
Before completing the steps in this manual, you must import and install the latest version of the *Amazon Web Services* PowerPack.

NOTE: If you are upgrading from an earlier version of the PowerPack, see the [Release Notes](#) for the version you are installing for upgrade instructions.

TIP: By default, installing a new version of a PowerPack overwrites all content from a previous version of that PowerPack that has already been installed on the target system. You can use the **Enable Selective PowerPack Field Protection** setting in the **Behavior Settings** page (System > Settings > Behavior) to prevent new PowerPacks from overwriting local changes for some commonly customized fields. (For more information, see the **System Administration** manual.)

To download and install a PowerPack:

1. Download the PowerPack from the [ScienceLogic Support Site](#).
2. Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
3. In the **PowerPack Manager** page, click the **[Actions]** button, then select *Import PowerPack*.
4. The **Import PowerPack** dialog box appears:



5. Click the **[Browse]** button and navigate to the PowerPack file.
6. When the **PowerPack Installer** modal appears, click the **[Install]** button to install the PowerPack.

NOTE: If you exit the **PowerPack Installer** modal without installing the imported PowerPack, the imported PowerPack will not appear in the **PowerPack Manager** page. However, the imported PowerPack will appear in the **Imported PowerPacks** modal. This page appears when you click the **[Actions]** menu and select *Install PowerPack*.

For information about opportunities and challenges with AWS, watch the video at <https://sciencelogic.com/product/resources/whiteboard-aws-opportunities-challenges>.

Chapter

2

Controlling What is Discovered by the PowerPack

Overview

The following sections describe the different methods to control what you can discover and monitor with the Amazon Web Services PowerPack:

<i>Configuring AWS for Monitoring Regions with AWS Config Enabled</i>	12
<i>Configuring AWS for Monitoring Regions with AWS CloudWatch</i>	13
<i>The Regions Header in the SOAP/XML Credential</i>	13
<i>Using IAM Permissions to Restrict SL1 Access to Specific Regions and Services</i>	15
<i>Example 1: One Region</i>	15
<i>Example 2: Multiple Regions</i>	16
<i>Configuring AWS for Monitoring Regions Using CloudWatch Namespaces</i>	17

Configuring AWS for Monitoring Regions with AWS Config Enabled

If your accounts have the AWS Config service enabled, then ScienceLogic recommends setting the **Embed Value [%2]** field in the **SOAP Options** section of the SOAP/XML credential you will create to "AUTO". The AWS Config service will then be used by SL1 to determine which regions and services are being used and only create the components needed. This will reduce the number of components created and will also reduce the load on the Data Collector.

NOTE: The Dynamic Applications "AWS: Account Resource Count Performance" and "AWS: Region Resource Count Performance" will only show data if the AWS Config service is enabled for those accounts/regions.

Configuring AWS for Monitoring Regions with AWS CloudWatch

If AWS config is not enabled, then ScienceLogic recommends setting the **Embed Value [%2]** in the **SOAP Options** section of the SOAP/XML credential you will create to "FILTER". This will use AWS Cloudwatch to determine which regions are reporting CloudWatch metrics and discover those regions. This will reduce the number of components created and will also reduce the load on the Data Collector.

The Regions Header in the SOAP/XML Credential

The **Regions** header is an optional header that can be inserted into the AWS SOAP/XML credential you will create to restrict which regions are discovered. This header supports a comma-separated list of regions that will be discovered and monitored. For example, the credential below shows the header with two specific regions. In this case, only those two regions would be discovered and monitored.

NOTE: The **Regions** header must not be included if "FILTER" or "AUTO" are used in the **Embed Value %2** field.

Edit Credential
✕

All Organizations
Select the organizations the credential belongs to *
Timeout (ms)
5000

Content Encoding:

Method:

HTTP Version:

URL *

HTTP Auth User: HTTP Auth Password:

Proxy Hostname/IP: Proxy Port:

Proxy User: Proxy Password:

Embedded Password [%P]:

Embed Value [%1]: Embed Value [%2]:

Embed Value [%3]: Embed Value [%4]:

HTTP Headers Add Header

- OrganizationArn:aws:iam:987654321 ✕
- AssumeRole:ScienceLogic-Monitor ✕
- AssumeRoleSession:SL1 ✕
- Regions:us-east-1,us-west-1 ✕
- OrganizationCreation:NAME:ID ✕

CURL Options Add CURL Option

Save & Test

Credential Tester

Select Credential Test:

Select Collector:

IP or Hostname to test *

Test Credential

Save & Close

Credential Editor [22]
✕

Edit SOAP/XML Credential #22
New Reset

Basic Settings

Profile Name: Content Encoding: Method: HTTP Version:

URL [http(s)://Host:Port/Path | %D = Aligned Device Address | %N = Aligned Device Host Name]

HTTP Auth User: HTTP Auth Password: Timeout (seconds):

Proxy Settings

Hostname/IP: Port: User:

CURL Options

- CAINFO
- CAPATH
- CLOSEPOLICY
- CONNECTTIMEOUT
- COOKIE
- COOKIEFILE
- COOKIEJAR
- COOKIELIST
- CRLF
- CUSTOMREQUEST
- DNSCACHETIMEOUT

Soap Options

Embedded Password [%P]:

Embed Value [%1]: Embed Value [%2]:

Embed Value [%3]: Embed Value [%4]:

HTTP Headers

+ Add a header

- OrganizationArn:am.aws.iam:0737868515 🔔
- AssumeRole:ScienceLogic-Monitor 🔔
- AssumeRoleSession:SL1 🔔
- Regions:us-east-1,us-west-1 🔔

Save Save As

The Regions Header in the SOAP/XML Credential

14

Using IAM Permissions to Restrict SL1 Access to Specific Regions and Services

You can use IAM policies in AWS to restrict which regions and services SL1 will monitor. To do this, you can create another IAM policy and apply that along with the SL1 monitoring policy to the applicable user or role(s).

To monitor specific regions and services, you must create a JSON policy in the AWS Management Console that uses the `NotAction`, `Allow`, and `Deny` policy elements to specify the regions and services you want to monitor as well as which regions and services you **do not** want to monitor.

NOTE: You must have at least Read-Only JSON policy permissions for the regions you want to monitor. You cannot discover regions for which you do not have policy permissions. At a minimum, you must at least have permissions for the us-east-1 (Virginia) region; without permissions for this region, you cannot discover general AWS services such as CloudFront, Route53, and OpsWorks.

TIP: When discovering resources in specific regions, you should ensure that any Global services or resources you want to monitor have the necessary access permissions.

NOTE: For more information about the `NotAction`, `Allow`, and `Deny` policy elements, see https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_elements_notaction.html.

The following sections provide two examples of region-specific JSON policies.

Example 1: One Region

This JSON Policy will deny any service that is not in the us-east-1 region. As a result, SL1 will discover only components in the us-east-1 region.

NOTE: In addition to the code below, you would need to specify the other resource permissions you want to allow in the policy.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyAllOutsideUSEast1",
      "Effect": "Deny",
      "NotAction": [
        "iam:*",
        "organizations:*",
        "support:*",
        "aws-portal:*",
        "s3:ListAllMyBuckets"
      ],
      "Resource": "*",
      "Condition": {
        "StringNotEquals": {
          "aws:RequestedRegion": "us-east-1"
        }
      }
    }
  ]
}

```

Example 2: Multiple Regions

This JSON Policy will deny any service that is not in the us-east-1, us-west-2, and ap-northeast-1 regions. As a result, SL1 will discover only components in the us-east-1, us-west-2, and ap-northeast-1 regions.

NOTE: In addition to the code below, you would need to specify the other resource permissions you want to allow in the policy.

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyAllOutsideUSWest2USEast1APNortheast1",
      "Effect": "Deny",
      "NotAction": [
        "iam:*",
        "organizations:*",
        "support:*",
        "aws-portal:*",
        "s3:ListAllMyBuckets"
      ],
      "Resource": "*",
      "Condition": {
        "StringNotEquals": {
          "aws:RequestedRegion": ["us-east-1", "us-west-2", "ap-northeast-1"]
        }
      }
    }
  ]
}

```

Configuring AWS for Monitoring Regions Using CloudWatch Namespaces

NOTE: These steps will be applied to all discovered AWS accounts on your SL1 system.

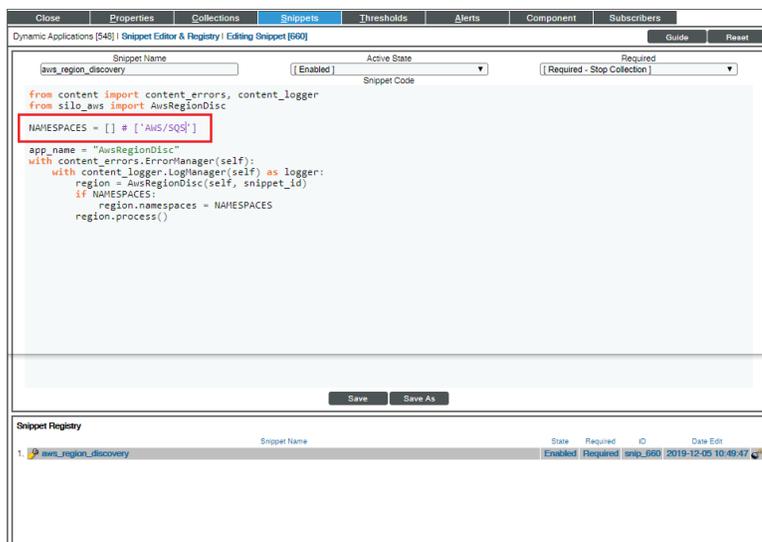
Users discovering with CloudWatch metrics can also discover regions where a specific namespace is available by editing the NAMESPACES field in the `aws_region_discovery` snippet in the "AWS: Region Discovery" Dynamic Application.

To edit the snippet:

1. Find the "AWS: Region Discovery" Dynamic Application in the **Dynamic Applications Manager** page (System > Manage > Applications) and click its wrench icon (🔧).
2. Click the **[Snippets]** tab and then click the wrench icon (🔧) for the `aws_region_discovery` snippet.
3. Edit the NAMESPACES field to include the namespace for your region. For example:

```
NAMESPACES = [ 'AWS/SQS' ]
```

4. Click **[Save]**.



Only regions that have services grouped in the specified namespace will be discovered. Global services will also be discovered.

NOTE: For more information about namespaces, see https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/viewing_metrics_with_cloudwatch.html.

Chapter

3

Configuration

Overview

Use the following menu options to navigate the SL1 user interface:

- To view a pop-out list of menu options, click the menu icon (.
- To view a page containing all the menu options, click the Advanced menu icon (.

The following sections describe several options available for using the *Amazon Web Services PowerPack* to monitor your AWS accounts.

<i>Configuring AWS to Report Billing Metrics</i>	20
<i>Filtering EC2 Instances By Tag</i>	22
<i>Filtering EC2 Instances by Tag in the SL1 Classic User Interface</i>	24
<i>Automatic SL1 Organization Creation</i>	25
<i>Monitoring Consolidated Billing Accounts</i>	26
<i>ScienceLogic Events and AWS Alarms</i>	26
<i>Using a Proxy Server</i>	26
<i>Configuring "AWS: Lambda Service Discovery"</i>	28
<i>Configuring "AWS: Lambda Function Qualified Discovery"</i>	31
<i>Configuring AWS Integration with Docker</i>	35
<i>Configuring AWS Integration with Kubernetes</i>	35
<i>Enabling the Prometheus Metrics Server</i>	37
<i>Define the Cluster Role</i>	37
<i>Define the ClusterRoleBinding</i>	39

<i>Map the IAM User or Role to the Kubernetes RBAC Role</i>	40
Example 1	40
Example 2	41
Amazon API Throttling Events	42
Support for AWS China Regions	42
Support for AWS GovCloud Regions	42

Configuring AWS to Report Billing Metrics

To use the "AWS: Billing Performance Percent" Dynamic Application, your AWS account must meet the following requirements:

- The user account you supplied in the AWS credential must have permission to view the us-east-1 zone.
- Your AWS account must be configured to export billing metrics to the CloudWatch service.

If your AWS account is not configured to export billing metrics to the CloudWatch service, the "AWS: Billing Performance Percent" Dynamic Application will generate the following event:

```
No billing metrics can be retrieved. Your AWS account is not configured to export
billing metrics into CloudWatch.
```

To configure your AWS account to export billing metrics to the CloudWatch service, perform the following steps:

1. Open a browser session and go to aws.amazon.com.
2. Click **[My Account]** and then select *Billing & Cost Management*. If you are not currently logged in to the AWS site, you will be prompted to log in:

Sign In or Create an AWS Account

What is your e-mail or mobile number?

E-mail or mobile number:

I am a new user.

I am a returning user and my password is:

[Sign in using our secure server](#)

[Forgot your password?](#)

Now Available
Amazon Aurora
Enterprise-class database at 1/10th the cost

[Learn more](#)

Learn more about [AWS Identity and Access Management](#) and [AWS Multi-Factor Authentication](#), features that provide additional security for your AWS Account. View full [AWS Free Usage Tier](#) offer terms.

About Amazon.com Sign In

Amazon Web Services uses information from your Amazon.com account to identify you and allow access to Amazon Web Services. Your use of this site is governed by our [Terms of Use](#) and [Privacy Policy](#) linked below.

[Terms of Use](#) [Privacy Policy](#) © 1996-2015, Amazon.com, Inc. or its affiliates

An **amazon.com** company

- After logging in, the **Billing & Cost Management Dashboard** page appears. In the left navigation bar, click **[Preferences]**. The **Preferences** page appears:

Preferences

Receive PDF Invoice By Email
Turn on this feature to receive a PDF version of your invoice by email. Invoices are generally available within the first three days of the month.

Receive Billing Alerts
Turn on this feature to monitor your AWS usage charges and recurring fees automatically, making it easier to track and manage your spending on AWS. You can set up billing alerts to receive email notifications when your charges reach a specified threshold. Once enabled, this preference cannot be disabled. [Manage Billing Alerts](#)

Receive Billing Reports
Turn on this feature to receive ongoing reports of your AWS charges once or more daily. AWS delivers these reports to the Amazon S3 bucket that you specify where indicated below. For consolidated billing customers, AWS generates reports only for paying accounts. Linked accounts cannot sign up for billing reports.

Save to S3 Bucket: [Verify](#)

[Save preferences](#)

- Select the **Receive Billing Alerts** checkbox.

CAUTION: If you enable this option, this option cannot be disabled.

5. Click the **[Save Preferences]** button.

Filtering EC2 Instances By Tag

To discover EC2 instances and filter them by tag, you can use the "AWS Credential - Tag Filter" sample credential to enter EC2 tag keys and values.

NOTE: Filtering EC2 instance by tag will apply to all accounts discovered.

NOTE: Any EC2 instances that have already been discovered, but do not match the tag filter, will be set to "Unavailable."

To define an AWS credential:

NOTE: If you are using an SL1 system prior to version 11.1.0, the new user interface does not include the **Duplicate** option for sample credential(s). ScienceLogic recommends that you use [the classic user interface and the Save As button](#) to create new credentials from sample credentials. This will prevent you from overwriting the sample credential(s).

1. Go to the **Credentials** page (Manage > Credentials).

2. Locate the **AWS Credential - Tag Filter** sample credential, click its **[Actions]** icon (⋮) and select **Duplicate**. A copy of the credential, called **AWS Credential - Tag Filter copy** appears.
3. Click the **[Actions]** icon (⋮) for the **AWS Credential - Tag Filter copy** credential and select **Edit**. The **Edit Credential** modal page appears:

4. Supply values in the following fields:
 - **Name**. Type a new name for your AWS credential.
 - **All Organizations**. Toggle on (blue) to align the credential to all organizations, or toggle off (gray) and then select one or more specific organizations from the **What organization manages this service?** drop-down field to align the credential with those specific organizations.
 - **Timeout (ms)**. Keep the default value.
 - **URL**. Enter a valid URL. This field is not used for this discovery method but must be populated with a valid URL for discovery to complete.
 - **HTTP Auth User**. Type your AWS access key ID.
 - **HTTP Auth Password**. Type your AWS secret access key.
 - Under **HTTP Headers**, edit the header provided:
 - **Tags**: `<operation>#<EC2-Tag-Key>#<EC2-Tag-Value>`. Type the tag, followed by its operation, tag key, or tag value. For example, if you want to filter by Tag Name, you would type the following:

Tags: equals#Name#Example

Valid operations include:

- equals
- notEquals

- contains
- notContains

You can chain together multiple filters separating them by a comma. For example:

```
Tags>equals#Name#Example,contains#Owner#Someone
```

5. Click the [Save & Close] button.

Filtering EC2 Instances by Tag in the SL1 Classic User Interface

To discover EC2 instances and filter them by tag, you can use the "AWS Credential - Tag Filter" sample credential to enter EC2 tag keys and values.

NOTE: Filtering EC2 instance by tag will apply to all accounts discovered.

NOTE: Any EC2 instances that have already been discovered, but do not match the tag filter, will be set to "Unavailable."

To define an AWS credential to discover EC2 instances and filter them by tag:

1. Go to the **Credential Management** page (System > Manage > Credentials).
2. Locate the **AWS Credential - Tag Filter** sample credential and click its wrench icon (🔧). The **Credential Editor** modal page appears:

The screenshot shows the 'Credential Editor [96]' window for editing a 'SOAP/XML Credential #96'. The interface is divided into several sections:

- Basic Settings:** Profile Name (AWS Credential - Tag Filter), Content Encoding ([text/xml]), Method ([POST]), HTTP Version ([HTTP/1.1]), URL (http://example.com/), HTTP Auth User ([AWS Account Access Key]), HTTP Auth Password, and Timeout (seconds) (5).
- Proxy Settings:** Hostname/IP, Port (0), and User.
- CURL Options:** A list of options including CAINFO, CAPATH, CLOSEPOLICY, CONNECTTIMEOUT, COOKIE, COOKIEFILE, COOKIEJAR, COOKIELIST, CRLF, CUSTOMREQUEST, and DNSCACHETIMEOUT.
- Soap Options:** Embedded Password [%P] and four Embed Value [%1] through [%4] fields.
- HTTP Headers:** A section with a '+ Add a header' button and a text input field containing 'Tags:<operation>#<EC2-Tag-Key>#<EC2-T'.

Buttons for 'New', 'Reset', 'Save', and 'Save As' are visible at the bottom of the editor.

3. Enter values in the following fields:

Basic Settings

- **Profile Name.** Type a new name for your AWS credential.
- **HTTP Auth User.** Type your AWS access key ID.
- **HTTP Auth Password.** Type your AWS secret access key.

HTTP Headers

- Edit the HTTP header provided:
 - **Tags:** <operation>#<EC2-Tag-Key>#<EC2-Tag-Value>. Type the tag, followed by its operation, tag key, or tag value. For example, if you want to filter by Tag Name, you would type the following:

```
Tags:equals#Name#Example
```

Valid operations include:

- equals
- notEquals
- contains
- notContains

You can chain together multiple filters separating them by a comma. For example:

```
Tags:equals#Name#Example,contains#Owner#Someone
```

4. Click the **[Save As]** button, and then click **[OK]**.

Automatic SL1 Organization Creation

This feature is only applicable to the two discovery methods that use the Assume Role and automatically discover multiple accounts.

When multiple accounts are discovered, this feature places each account in its own SL1 organization. This feature requires an optional header in the SOAP/XML credential you will create. When this header is present, it will place each account into a new SL1 organization. When this header is not present, each account will be placed in the SL1 organization selected in the discovery session. The name of the organization can be controlled depending on what is provided in the header as follows:

- **OrganizationCreation:NAME:ID.** Autocreates an SL1 organization for accounts using AssumeRole. You can enter one of the following options:
 - **OrganizationCreation:NAME.** The name of the organization will contain the name of the user.
 - **OrganizationCreation:ID.** The name of the organization will contain the ID of the user.

- **OrganizationCreation:ID:NAME**. The name of the organization will contain both the ID and name of the user, in that order.
- **OrganizationCreation:NAME:ID**. The name of the organization will contain both the name and ID of the user, in that order.

Monitoring Consolidated Billing Accounts

Consolidated billing is an option provided by Amazon that allows multiple AWS accounts to be billed under a single account. For more information about consolidated billing, see <http://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/consolidated-billing.html>.

If a consolidated billing account is monitored by SL1, the billing metrics associated with that account include only the consolidated amounts, per service. If you use consolidated billing and want to collect billing metrics per-account, you must discover each account separately. To monitor only the billing metrics for an AWS account, you can create credentials that include only billing permissions.

ScienceLogic Events and AWS Alarms

In addition to SL1 collecting metrics for AWS instances, you can configure CloudWatch to send alarm information to SL1 via API. SL1 can then generate an event for each alarm.

For instructions on how to configure CloudWatch and SL1 to generate events based on CloudWatch alarms, see the [Configuring Inbound CloudWatch Alarms](#) section.

Using a Proxy Server

You can use a proxy server with the [Manual Discovery](#) and the [Automated Discovery Using AssumeRole with a Single IAM Key from the AWS Master Account](#) discovery methods.

To use a proxy server in both cases, you must fill in the proxy settings in the SOAP/XML credential.

Edit Credential

Name*
AWS Credential - Proxy

All Organizations Select the organizations the credential belongs to*

Timeout (ms)
5000

Content Encoding: text/xml | Method: POST | HTTP Version: http/1.1

URL*
http://example.com/

HTTP Auth User: [AWS Account Access Key] | HTTP Auth Password: *****

Proxy Hostname/IP: Host-Or-Ip | Proxy Port: 0

Proxy User: <Proxy_User> | Proxy Password: *****

Embedded Password [%P]

Embed Value [%1] | Embed Value [%2]
Embed Value [%3] | Embed Value [%4]

Close

Credential Tester

Select Credential Test

Select Collector
CUG | s1a1a1: 10.128.68.26

IP or Hostname to test*

Test Credential

Credential Editor [31]

Edit SOAP/XML Credential #31 [New] [Reset]

Basic Settings

Profile Name: AWS Credential - Proxy | Content Encoding: [text/xml] | Method: [POST] | HTTP Version: [HTTP/1.1]

URL [http(s)://Host:Port/Path | %D = Aligned Device Address | %N = Aligned Device Host Name]
http://example.com/

HTTP Auth User: [AWS Account Access Key] | HTTP Auth Password: [] | Timeout (seconds): 2

Proxy Settings

Hostname/IP: Host-Or-Ip | Port: 0 | User: <Proxy_User>

CURL Options

- CAINFO
- CAPATH
- CLOSEPOLICY
- CONNECTTIMEOUT
- COOKIE
- COOKIEFILE
- COOKIEJAR
- COOKIELIST
- CRLF
- CUSTOMREQUEST
- DNSCACHETIMEOUT

Soap Options

Embedded Password [%P]
Embed Value [%1] | Embed Value [%2]
Embed Value [%3] | Embed Value [%4]

HTTP Headers

+ Add a header

Save | Save As

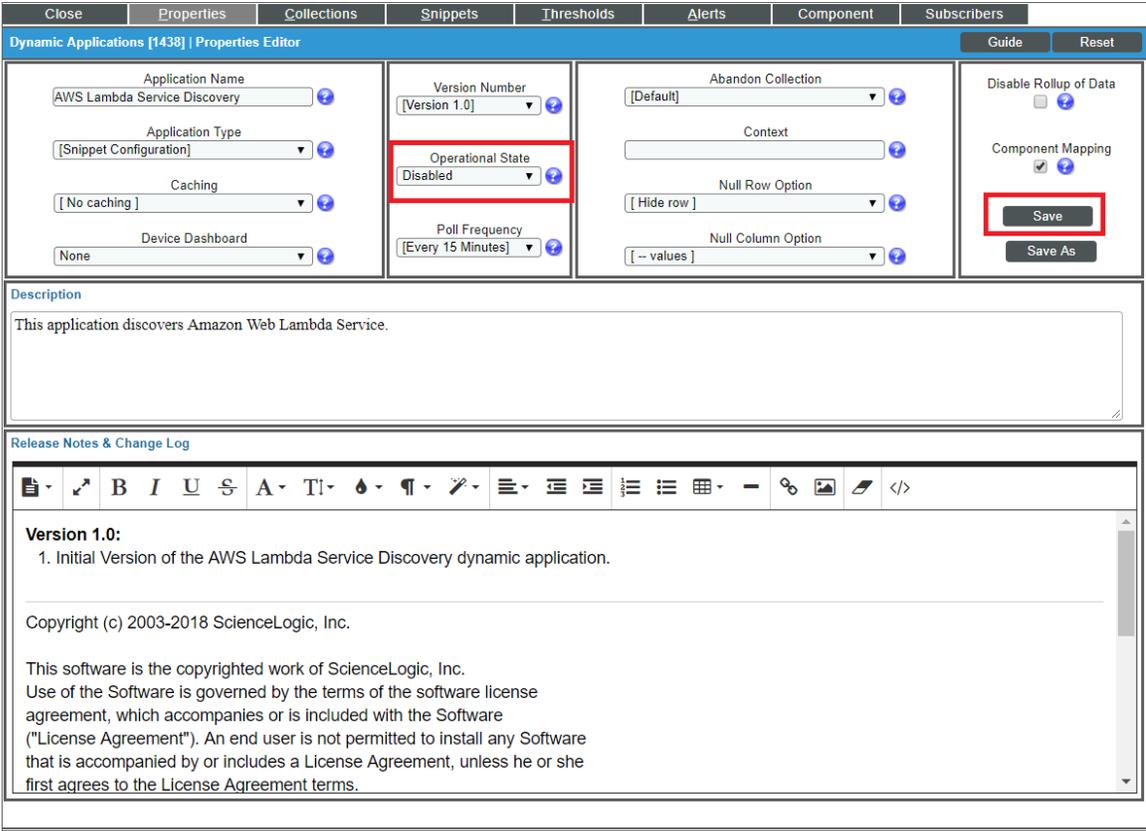
For the [Automated Discovery Using AssumeRole with a Single IAM Key from the AWS Master Account](#) discovery method, if the proxy does not support ping passthrough you will also need to follow the steps in the [Automated Discovery Using AssumeRole with a Single IAM Key from the AWS Master Account](#) section without ping support.

Configuring "AWS: Lambda Service Discovery"

By default, the "AWS: Lambda Service Discovery" Dynamic Application is configured to discover only regular Lambda functions, not replica functions. If you want to discover both regular and replica Lambda functions, then you must configure the "AWS: Lambda Service Discovery" Dynamic Application to do so **prior** to discovering your Lambda service.

To configure the "AWS: Lambda Service Discovery" Dynamic Application to discover both regular and replica Lambda functions:

1. Go to the **Dynamic Applications Manager** page (System > Manage > Applications).
2. Locate the "AWS: Lambda Service Discovery" Dynamic Application and click its wrench icon (). The **Dynamic Applications Properties Editor** page appears.
3. In the **Operational State** field, select *Disabled*, and then click **[Save]**. This disables the Dynamic Application from collecting data.



The screenshot shows the 'Dynamic Applications Properties Editor' for 'AWS Lambda Service Discovery'. The 'Operational State' dropdown is set to 'Disabled' and is highlighted with a red box. The 'Save' button is also highlighted with a red box. The interface includes tabs for Close, Properties, Collections, Snippets, Thresholds, Alerts, Component, and Subscribers. The main configuration area has fields for Application Name, Application Type, Caching, Device Dashboard, Version Number, Operational State, Poll Frequency, Abandon Collection, Context, Null Row Option, Null Column Option, Disable Rollup of Data, and Component Mapping. A description box states: 'This application discovers Amazon Web Lambda Service.' Below is a 'Release Notes & Change Log' section with a rich text editor showing version 1.0 details and copyright information for ScienceLogic, Inc.

4. Click the **[Snippets]** tab. The **Dynamic Applications Snippet Editor & Registry** page appears.
5. In the **Snippet Registry** pane, click the wrench icon () for the "aws_lambda_service_discovery" snippet.

- In the **Active State** field, select *Disabled*, and then click **[Save]**. This disables the "aws_lambda_service_discovery" snippet.

Dynamic Applications [1438] | Snippet Editor & Registry | Editing Snippet [1782] Guide Reset

Snippet Name: aws_lambda_service_discovery Active State: Disabled Required: [Required - Stop Collection]

```

from content import content_errors, content_logger
from silo_aws import AwsLambdaServiceDiscovery

app_name = 'AwsLambdaServiceDiscovery'
with content_errors.ErrorManager(self):
    with content_logger.LogManager(self) as logger:
        replica_discovery = False
        AwsLambdaServiceDiscovery(self, snippet_id, replica_discovery).process()

```

Save Save As

Snippet Registry

	Snippet Name	State	Required	ID	Date Edit
1	aws_lambda_service_discovery	Enabled	Required	snip_1782	2018-07-09 09:58:21
2	aws_lambda_service_discovery_show_replicas	Enabled	Required	snip_1783	2018-07-10 07:51:04

- In the **Snippet Registry** pane, click the wrench icon () for the "aws_lambda_service_discovery_show_replicas" snippet.
- In the **Active State** field, select *Enabled*, and then click **[Save]**. This enables the "aws_lambda_service_discovery_show_replicas" snippet.
- Click the **[Collections]** tab. The **Dynamic Applications | Collections Objects** page appears.

- Click the wrench icon (🔧) for the first Collection Object listed in the **Collection Object Registry** pane, select `aws_lambda_service_discovery_show_replicas` in the **Snippet** field for that Collection Object, and then click **[Save]**.

Dynamic Applications [1438] | Collection Objects

Object Name: Availability

Snippet Arguments: exists

Class Type: [10 Config Character]

String Type: [Standard]

Custom Attribute: [None]

Snippet: [aws_lambda_service_discovery_show_replicas]

Group / Usage Type: [Group 1] [Standard]

Asset / Form Link: [None] [None]

Inventory Link: [Disabled]

Change Alerting: [Disabled]

Table Alignment: [Left]

Hide Object:

Save Save As

Disable Object Maintenance

Collection Object Registry

	Object Name	Class Type	Class ID	Snippet Arguments	Group	ID	Asset Link	Change Alerting	Align	Edit Date	
1	Availability	Config Character	10	exists	1	o_16713	--	Disabled	Left	2018-07-10 07:51:52	<input type="checkbox"/>
2	Distinguished Name	Config Character	10	arn	1	o_16717	--	Disabled	Left	2018-07-10 07:51:17	<input type="checkbox"/>
3	Id	Config Character	10	id	1	o_16714	--	Disabled	Left	2018-07-10 07:51:23	<input type="checkbox"/>
4	Lambda	Label (Config Group)	108		1	o_16716	--	Disabled	Left	2018-07-10 07:51:28	<input type="checkbox"/>
5	Name	Config Character	10	name	1	o_16715	--	Disabled	Left	2018-07-10 07:51:32	<input type="checkbox"/>

[Select Action] Go

- Repeat step 10 for all of the remaining Collection Objects listed in the **Collection Object Registry** pane.
- Click the **[Properties]** tab.
- In the **Operational State** field, select *Enabled*, and then click **[Save]**. This re-enables data collection for the Dynamic Application.

NOTE: If you configure the "AWS: Lambda Service Discovery" Dynamic Application to discover both regular and replica Lambda functions, then when you run discovery, the Dynamic Applications in the Amazon Web Services PowerPack will create *parent/child relationships* between replica Lambda functions and their corresponding master Lambda functions. In this scenario, the *Device View and other device component maps* will display the relationship in this order: Lambda Function Service > Lambda Replica Function > Master Lambda Function. The replica appears as the parent to the master Lambda function because the replica could be in the same or a different region than the master Lambda function.

Configuring "AWS: Lambda Function Qualified Discovery"

By default, the "AWS: Lambda Function Qualified Discovery" Dynamic Application is configured to discover and model all Lambda alias components. An **alias** is a qualifier inside an AWS Lambda function that enables the user to control which versions of the Lambda function are executable—for instance, a production version and a test version.

When the "AWS: Lambda Function Qualified Discovery" Dynamic Application is configured to discover alias components, SL1 collects data only for the Lambda function versions specified in the alias.

Depending on your needs, you can optionally configure the Dynamic Application to instead do one of the following:

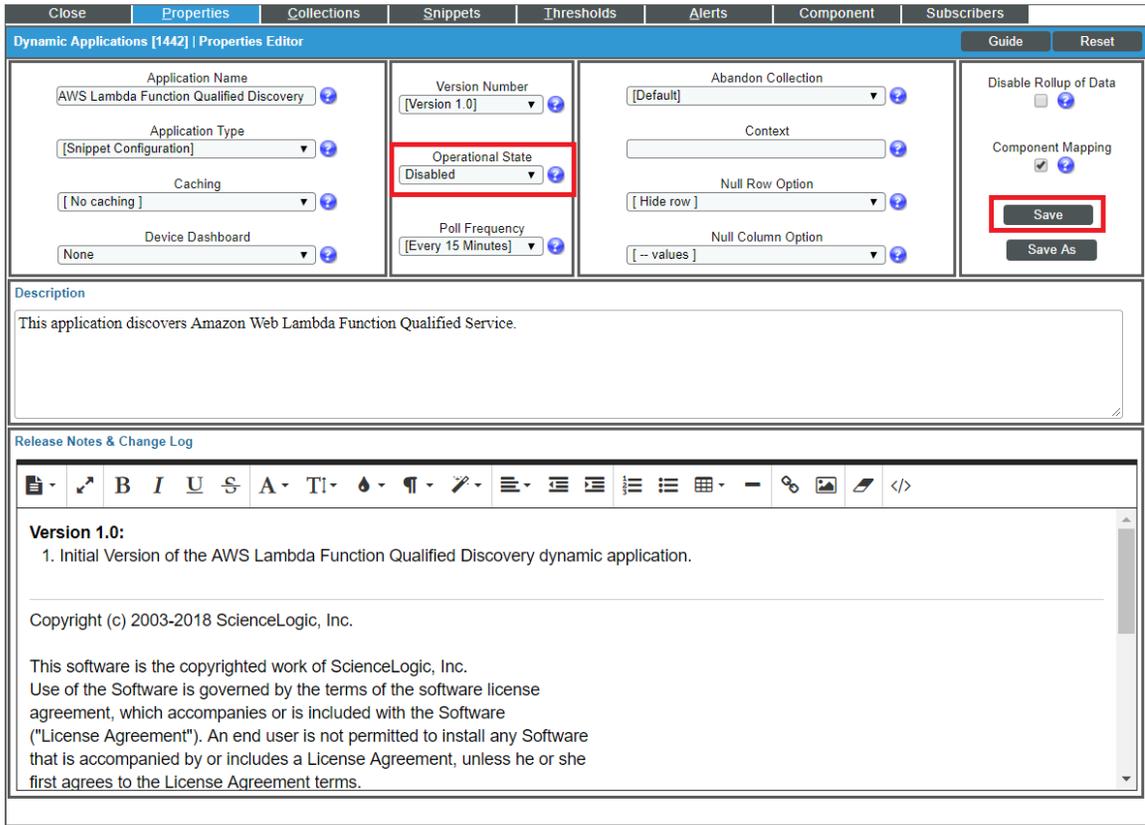
- Discover and model all Lambda version components. If you select this configuration, SL1 collects data for all existing versions of the Lambda function.
- Discover and model only Lambda version components with AWS configurations filtered by a trigger. If you select this configuration, SL1 collects data only for versions of the Lambda function that have triggers or are specified in an alias.

NOTE: If you have [configured the "AWS: Lambda Service Discovery" Dynamic Application](#) to discover both regular and replica Lambda functions and you want SL1 to [create dynamic component map relationships](#) between replica Lambda functions and their parent Lambda function versions, you must follow these instructions to configure the "AWS: Lambda Function Qualified Discovery" Dynamic Application to discover and model all Lambda version components.

To configure the "AWS: Lambda Function Qualified Discovery" Dynamic Application:

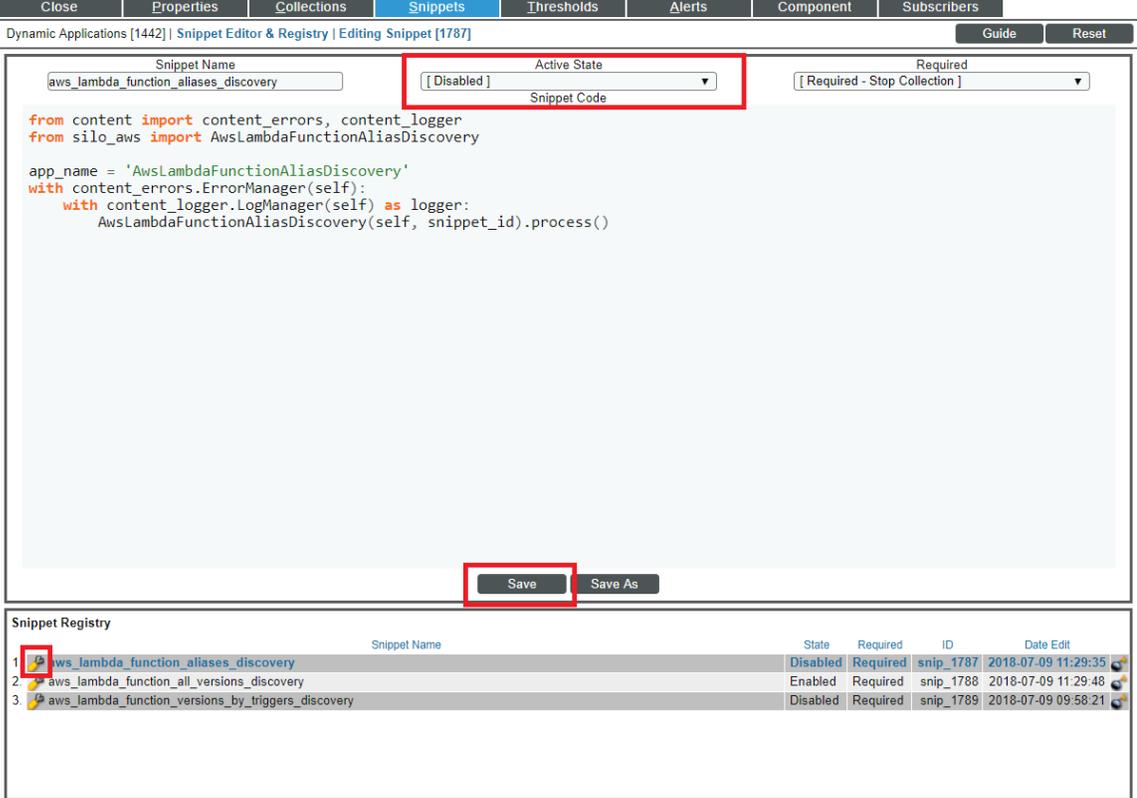
1. Go to the **Dynamic Applications Manager** page (System > Manage > Applications).
2. Locate the "AWS: Lambda Function Qualified Discovery" Dynamic Application and click its wrench icon (). The **Dynamic Applications Properties Editor** page appears.

- In the **Operational State** field, select *Disabled*, and then click **[Save]**. This disables the Dynamic Application from collecting data.



- Click the **[Snippets]** tab. The **Dynamic Applications Snippet Editor & Registry** page appears. The **Snippet Registry** pane includes the following snippets:
 - aws_lambda_function_aliases_discovery*. When this snippet is enabled, the Dynamic Application discovers all Lambda alias components.
 - aws_lambda_function_all_versions_discovery*. When this snippet is enabled, the Dynamic Application discovers all Lambda version components.
 - aws_lambda_function_versions_by_triggers_discovery*. When this snippet is enabled, the Dynamic Application discovers Lambda version components with AWS configurations containing a trigger or those with an alias.

5. One at a time, click the wrench icon (🔧) for each of the snippets, select *Enabled* or *Disabled* in the **Active State** field, and then click **[Save]** to enable the appropriate snippet and disable the others.



The screenshot shows the 'Snippet Editor & Registry' interface. The 'Active State' dropdown menu is set to 'Disabled'. The 'Save' button is highlighted with a red box. The snippet code is as follows:

```
aws_lambda_function_aliases_discovery

from content import content_errors, content_logger
from silo_aws import AwsLambdaFunctionAliasDiscovery

app_name = 'AwsLambdaFunctionAliasDiscovery'
with content_errors.ErrorManager(self):
    with content_logger.LogManager(self) as logger:
        AwsLambdaFunctionAliasDiscovery(self, snippet_id).process()
```

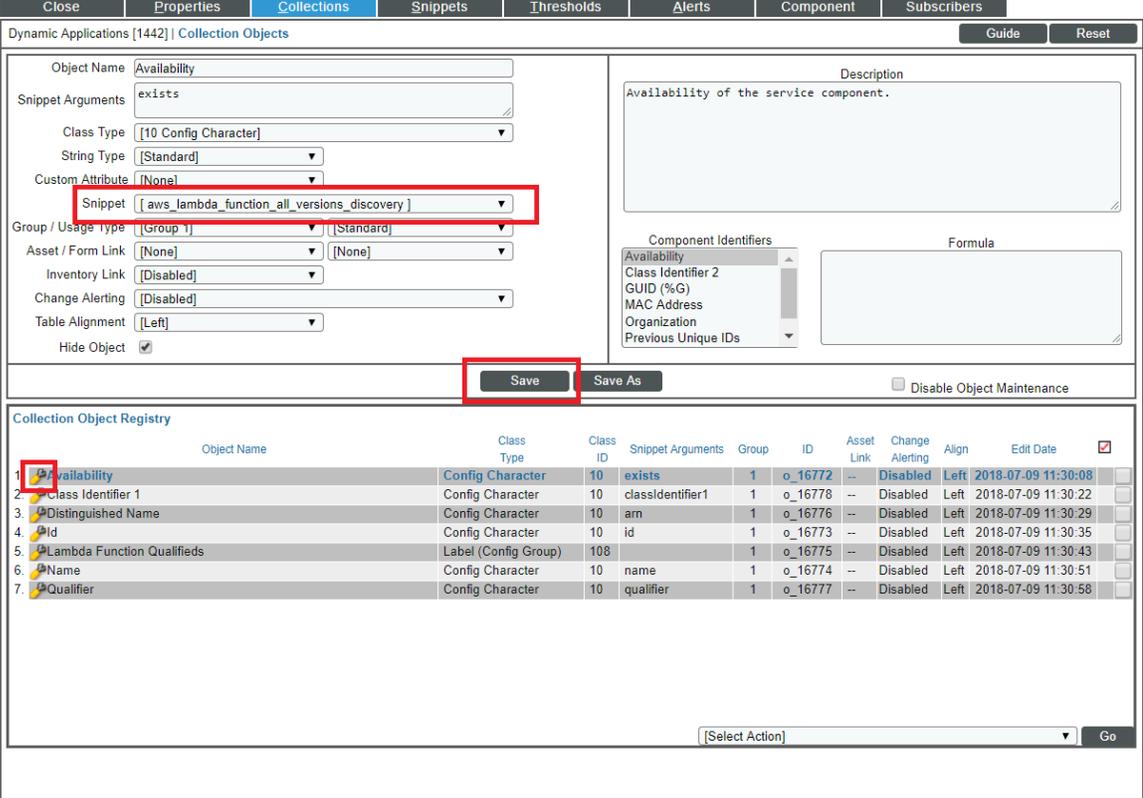
The Snippet Registry table below shows the state of three snippets:

	Snippet Name	State	Required	ID	Date Edit
1	aws_lambda_function_aliases_discovery	Disabled	Required	snip_1787	2018-07-09 11:29:35
2	aws_lambda_function_all_versions_discovery	Enabled	Required	snip_1788	2018-07-09 11:29:48
3	aws_lambda_function_versions_by_triggers_discovery	Disabled	Required	snip_1789	2018-07-09 09:58:21

NOTE: You can enable only one of these snippets at a time.

6. Click the **[Collections]** tab. The **Dynamic Applications | Collections Objects** page appears.

- Click the wrench icon () for the first Collection Object listed in the **Collection Object Registry** pane, select the snippet you enabled in step 5 in the **Snippet** field for that Collection Object, and then click **[Save]**.



Dynamic Applications [1442] | Collection Objects

Object Name: Availability

Snippet Arguments: exists

Class Type: [10 Config Character]

String Type: [Standard]

Custom Attribute: [None]

Snippet: **[aws_lambda_function_all_versions_discovery]**

Group / Usage Type: [Group 1] [Standard]

Asset / Form Link: [None] [None]

Inventory Link: [Disabled]

Change Alerting: [Disabled]

Table Alignment: [Left]

Hide Object:

Save Save As Disable Object Maintenance

Collection Object Registry

	Object Name	Class Type	Class ID	Snippet Arguments	Group	ID	Asset Link	Change Alerting	Align	Edit Date	
1	Availability	Config Character	10	exists	1	o_16772	--	Disabled	Left	2018-07-09 11:30:08	<input type="checkbox"/>
2	Class Identifier 1	Config Character	10	classIdentifier1	1	o_16778	--	Disabled	Left	2018-07-09 11:30:22	<input type="checkbox"/>
3	Distinguished Name	Config Character	10	am	1	o_16776	--	Disabled	Left	2018-07-09 11:30:29	<input type="checkbox"/>
4	Id	Config Character	10	id	1	o_16773	--	Disabled	Left	2018-07-09 11:30:35	<input type="checkbox"/>
5	Lambda Function Qualified	Label (Config Group)	108		1	o_16775	--	Disabled	Left	2018-07-09 11:30:43	<input type="checkbox"/>
6	Name	Config Character	10	name	1	o_16774	--	Disabled	Left	2018-07-09 11:30:51	<input type="checkbox"/>
7	Qualifier	Config Character	10	qualifier	1	o_16777	--	Disabled	Left	2018-07-09 11:30:58	<input type="checkbox"/>

[Select Action] Go

- Repeat step 7 for all of the remaining Collection Objects listed in the **Collection Object Registry** pane.
- Click the **[Properties]** tab.
- In the **Operational State** field, select *Enabled*, and then click **[Save]**. This re-enables data collection for the Dynamic Application. The next time discovery is run, new component devices might be discovered and some previously discovered components might become unavailable, depending on how you configured the Dynamic Application.

NOTE: If you configure the "AWS: Lambda Function Qualified Discovery" Dynamic Application to discover Lambda alias or version components and your AWS service includes an API Gateway that triggers a Lambda Function, then the Dynamic Applications in the Amazon Web Services PowerPack will create [a device relationship](#) between that Lambda Function and its corresponding Lambda alias or version component device.

Configuring AWS Integration with Docker

If you have discovered EC2-backed ECS clusters using the *Amazon Web Services PowerPack*, you can optionally use the *Docker PowerPack* to collect container information in addition to what the AWS API provides for the ECS service.

NOTE: This integration does not work with Fargate-backed ECS clusters.

To configure this integration, cURL version 7.40 or later must be installed on the ECS AMI image. For example, the 2018.03 ECS AMI image is compatible because it includes cURL 7.43.1.

Additionally, you must install the most recent version of the *Docker PowerPack* on your SL1 System and run a discovery session using an SSH credential that will work on the EC2 host(s). This discovery session will discover the EC2 instances that comprise the ECS cluster and align the Docker host Dynamic Applications with those EC2 instances. Optionally, you can merge the EC2 host with the Docker host if you so choose.

NOTE: For more information about the *Docker PowerPack*, including instructions about creating the SSH credential and running discovery, see the *Monitoring Docker* manual.

NOTE: ScienceLogic does not recommend enabling and securing the Docker HTTP API when aligning EC2 instances with Docker hosts. Doing so requires you to complete manual steps on each EC2 host. Furthermore, if you use this method and then merge the EC2 host with the Docker host, data collection will fail for all containers that are children of the merged host.

Configuring AWS Integration with Kubernetes

If you are using the AWS EKS service you can optionally use the *Kubernetes PowerPack* to provide visibility into your Kubernetes worker nodes and their associated workloads.

To use the *Kubernetes PowerPack* with the *Amazon Web Services PowerPack*, you must have the following versions of these PowerPacks installed:

- *Amazon Web Services* version 118 or later
- *Kubernetes* version 104 or later

If you are using AWS EKS but do **not** want to use this feature, then it is recommended to disable the "AWS EKS Cluster Virtual Discovery" Dynamic Application. To do this:

1. Go to the **Dynamic Applications Manager** page (System > Manage > Dynamic Applications).
2. Search for "AWS EKS" in the **Dynamic Application Name** column.

3. Click on the wrench icon () for the "AWS EKS Cluster Virtual Device Discovery" Dynamic Application and set the **Operational State** dropdown to *Disabled*.
4. Click the **[Save]** button.

Using the *Kubernetes* PowerPack is completely automated on SL1. If the proper credentials have been assigned on AWS and the AWS EKS Cluster, then SL1 will automatically discover the Kubernetes worker nodes and the associated workloads. The following additional components will be automatically created:

1. A new DCM tree root device to represent the Kubernetes cluster. This will be a virtual device of the type "Kubernetes Cluster".
2. A child component of the cluster will be created for each worker node in the cluster. This will be a component device of the type "Kubernetes Node".
3. A child component of the cluster will be created that represents the Namespaces. This will be a component device of the type "Kubernetes Namespace Folder".
4. A child component of the Namespace Folder will be created for each Namespace discovered. This will be a component device of the type "Kubernetes Namespace".
5. A child component of the Namespace will be created for each controller discovered as follows:
 - Kubernetes Daemon Set
 - Kubernetes Deployment

NOTE: At most only a single component is created to represent a controller. If a deployment and replica set exists, SL1 models only the deployment and replica set info as provided by the deployment component.

- Kubernetes Job
 - Kubernetes Cronjob
 - Kubernetes Replication Controller
 - Kubernetes Replication Set
 - Kubernetes Stateful Set
6. A child component of the cluster will be created for each ingress defined. This will be a component device of the type "Kubernetes: Ingress".

For SL1 to automatically discover the EKS cluster, you must perform the following steps:

NOTE: When logging into the Kubernetes cluster, ensure that the AWS credentials that `kubectl` is using are already authorized for your cluster. The IAM user that created the cluster has these permissions by default.

1. **Enable the Prometheus Metrics Server.** AWS EKS does not have the metrics server enabled by default. This is highly recommended as it will provide CPU and memory utilization metrics for both the worker nodes as well as the pods.

NOTE: SL1 automatically aggregates the CPU and memory utilization for pods and presents data at the controller level.

2. [Define the cluster role](#) needed by SL1 so that it can access the necessary APIs. This is done on the EKS Cluster.
3. [Define the ClusterRoleBinding](#). This is done on the EKS Cluster.
4. [Map the IAM user or role to the RBAC role and groups](#) using the aws-auth ConfigMap. This is done on the EKS Cluster.

Enabling the Prometheus Metrics Server

The Prometheus Metrics Server is required to provide CPU and memory utilization for pods and for nodes. The metrics server can be easily installed on Kubernetes clusters with the following:

```
kubectl apply -f https://github.com/kubernetes-sigs/metrics-server/releases/latest/download/components.yaml
```

To verify that the server is running, execute the command:

```
kubectl get deployment metrics-server -n kube-system
```

The following output will show that the metrics server is running:

NAME	READY	UP-TO-DATE	AVAILABLE	AGE
metrics-server	1/1	1	14h	

Define the Cluster Role

The cluster role defines the minimum permissions that SL1 needs to monitor the Kubernetes cluster. ClusterRole is used as it provides access to all namespaces. Since SL1 is directly monitoring the Kubernetes cluster via the Kubernetes API, this role's permissions need to be defined on the cluster itself.

To define the cluster role in Kubernetes:

1. Log in to the EKS cluster with the same user or role that created the cluster.
2. Create a new file called `SL1_cluster_role.yaml` and cut and paste the following text into that file:

CAUTION: YAML requires specific spacing. Please double-check the spacing after cutting-and-pasting code into YAML files.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: eks-readonly-clusterrole
```

```

rules:
- apiGroups:
  - ""
  resources:
  - nodes
  - namespaces
  - pods
  - replicationscontrollers
  - events
  - persistentvolumes
  - persistentvolumeclaims
  - componentstatuses
  - services
  verbs:
  - get
  - list
  - watch
- apiGroups:
  - apps
  resources:
  - deployments
  - daemonsets
  - statefulsets
  - replicaset
  verbs:
  - get
  - list
  - watch
- apiGroups:
  - batch
  resources:
  - jobs
  - cronjobs
  verbs:
  - get
  - list
  - watch
- apiGroups:
  - metrics.k8s.io
  resources:
  - nodes
  - pods
  verbs:
  - get
  - list
  - watch
- apiGroups:
  - networking.k8s.io
  resources:
  - ingresses
  verbs:
  - get
  - list
  - watch
- apiGroups:
  - autoscaling

```

```
resources:
- horizontalpodautoscalers
verbs:
- get
- list
- watch
```

The above file defines the minimum read-only permissions needed for SL1 to monitor Kubernetes.

3. Once the file is defined, execute the following command to apply the file:

```
kubectl apply -f cluster_role.yaml
```

Define the ClusterRoleBinding

Once the role is defined, it must be bound to users, groups, or services. This is done by defining a ClusterRoleBinding:

1. Log in to the EKS cluster with the same user or role that created the cluster.
2. Create a new file called `SL1_ClusterRoleBinding.yaml` and cut and paste the following text into that file:

CAUTION: YAML requires specific spacing. Please double-check the spacing after cutting-and-pasting code into YAML files.

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRoleBinding
metadata:
  name: eks-cluster-role-binding
subjects:
- kind: User
  name: Sciencelogic-Monitor
  apiGroup: rbac.authorization.k8s.io
roleRef:
  kind: ClusterRole
  name: eks-readonly-clusterrole
  apiGroup: rbac.authorization.k8s.io
```

3. Once the file is created, apply the ClusterRoleBinding by executing the following command:

```
kubectl apply -f SL1_ClusterRoleBinding.yaml
```

NOTE: Under subjects, "name: Sciencelogic-Monitor" defines the Kubernetes user and it must match the username field in the config map shown below.

NOTE: Under roleRef, "name: eks-readonly-clusterrole" must match the name defined in the cluster role.

Map the IAM User or Role to the Kubernetes RBAC Role

After defining the ClusterRoleBinding, you must map the AWS credentials that SL1 is using to the username created above in the `SL1_ClusterRoleBinding.yaml` file. To do this, perform the following steps:

1. Enter the `kubectl edit -n kube-system configmap/aws-auth` command. This will bring up the `configmap`. How the `configmap` is updated depends on what type of IAM was used to discover SL1.

NOTE: If the `configmap/aws-auth` does not exist, follow the procedures defined in <https://docs.aws.amazon.com/eks/latest/userguide/add-user-role.html>

Example 1

If SL1 has discovered your AWS organization using assume role, add the following text to the `mapRoles` section in the `configmap`:

CAUTION: YAML requires specific spacing. Please double-check the spacing after cutting-and-pasting code into YAML files.

```
- groups:
  - eks-cluster-role-binding
    rolearn:arn:aws:iam::<Account number that hosts the Kubernetes cluster-
>:role/Sciencelogic-Monitor
    username: Sciencelogic-Monitor
```

NOTE: If `mapRoles` does not exist, then you can add the `mapRoles` section to the `configmap`.

The text should appear in the `configmap` as the highlighted text below:

```
# Please edit the object below. Lines beginning with a '#' will be ignored,
# and an empty file will abort the edit. If an error occurs while saving, this file
will be
# reopened with the relevant failures
#
apiVersion: v1
data:
  mapRoles: |
    - groups:
      - system:bootstrappers
      - system:nodes
      rolearn: arn:aws-us-gov:iam::<account number>:role/eksctl-eks-cluster-testfriday-
nod-NodeInstanceRole-6VCMS669U9NA
      username: system:node:{{EC2PrivateDNSName}}
    - groups:
```

```

- eks-cluster-role-binding
  rolearn: arn:aws:iam::<account number>:role/Sciencelogic-Monitor
  username: Sciencelogic-Monitor
kind: ConfigMap
metadata:
  creationTimestamp: "2021-07-30T20:43:55Z"
  name: aws-auth
  namespace: kube-system
  resourceVersion: "173718"
  selfLink: /api/v1/namespaces/kube-system/configmaps/aws-auth
  uid: dlbcdafd-fc40-44e6-96d4-9a079b407d06

```

Example 2

If SL1 has been discovered with a single IAM key for the account, add the following text to the mapUsers: section of the configmap:

CAUTION: YAML requires specific spacing. Please double-check the spacing after cutting-and-pasting code into YAML files.

```

- groups:
  - eks-cluster-role-binding
    userarn:arn:aws:iam::<Account number that hosts the Kubernetes cluster>:user/<Name
of the user associated with the IAM key
    username: Sciencelogic-Monitor

```

The text should appear in the configmap as the highlighted text below:

```

# Please edit the object below. Lines beginning with a '#' will be ignored,
# and an empty file will abort the edit. If an error occurs while saving, this file
will be
# reopened with the relevant failures
#
apiVersion: v1
data:
  mapRoles: |
    - groups:
      - system:bootstrappers
      - system:nodes
      rolearn: arn:aws-us-gov:iam::<account number>:role/eksctl-eks-cluster-testfriday-
nod-NodeInstanceRole-6VCMS669U9NA
      username: system:node:{{EC2PrivateDNSName}}
  mapUsers: |
    - groups:
      - eks-cluster-role-binding
      userarn: arn:aws:iam::<account number>:user/<username>
      username: Sciencelogic-Monitor
kind: ConfigMap
metadata:
  creationTimestamp: "2021-07-30T20:43:55Z"
  name: aws-auth

```

```
namespace: kube-system
resourceVersion: "173718"
selfLink: /api/v1/namespaces/kube-system/configmaps/aws-auth
uid: d1bcdafd-fc40-44e6-96d4-9a079b407d06
```

NOTE: In `userarn: arn:aws:iam::<account number>:user/<username>`, the `username` is the `userarn` that SL1 is using to monitor the Kubernetes cluster.

NOTE: Under `mapUsers`, the `username:` is the name used in the `ClusterRoleBinding`.

Amazon API Throttling Events

By default, SL1 will use the Collector Group aligned with the root AWS virtual device to retrieve data from AWS devices and services.

If SL1 must collect data from a large set of AWS devices and services, SL1 might generate Notify events with a message ending in the text "Retry #1-10 Sleeping: ... seconds". SL1 generates these events when the Amazon API throttles collection in response to a large number of requests to the API. Even though SL1 is generating Notify "Retry" events, SL1 is still collecting data from AWS. This issue commonly occurs when a specific Amazon data center edge is close to capacity.

If SL1 generates the Minor event "Collection missed on <device> on 5 minute poll", this indicates that SL1 was unable to retrieve that specific datum from the Amazon cloud during the most recent five-minute polling cycle. If you frequently see the "Collection missed" event across your cloud, you must contact Amazon support to whitelist the IP address of your Data Collector. This will prevent further throttling from occurring.

Support for AWS China Regions

Currently, the only method of discovery for AWS China Regions is the [Manual Discovery](#) method. In this case, the **Embed Value %1** field in the [SOAP/XML credential](#) must contain the specific Chinese region to be monitored.

Support for AWS GovCloud Regions

AWS GovCloud Regions can be discovered using all discovery methods as defined below:

- For an individual account using the [Manual Discovery](#) method, type the name of the AWS GovCloud region in the **Embed Value %1** field in the [SOAP/XML credential](#).
- For those using one of the discovery methods with AssumeRole, enter one of the following URLs in the **URL** field of the [SOAP/XML credential](#) to specify the specific government region:

- <https://organizations.us-gov-west-1.amazonaws.com>
- <https://organizations.us-gov-east-1.amazonaws.com>

NOTE: All examples shown are for commercial AWS accounts. When AWS Gov is being monitored, the JSON data that refers to ARN will need to be modified from "aws" to "aws-us-gov". For example:
Resource": "arn:aws:iam::<account number>:role/Sciencelogic-Monitor would need to be Resource": "arn:aws:iam-us-gov::<account number>:role/Sciencelogic-Monitor

Chapter

4

Minimum Permissions

Overview

The following sections describe the minimum permissions that must be set before you can run discovery with the *Amazon Web Services PowerPack*:

[Minimum Permissions Needed to Monitor Your AWS Accounts](#) 44

Minimum Permissions Needed to Monitor Your AWS Accounts

The following table displays the minimum permissions required for Dynamic Applications in the *Amazon Web Services PowerPack* to collect data.

Service	Actions	
API Gateway	Read	GET
CloudFront	List	ListDistributions ListInvalidations ListStreamingDistributions
	Read	GetDistribution GetStreamingDistribution
CloudTrail	List	DescribeTrails
	Read	GetTrailStatus

Service	Actions	
CloudWatch	List	ListMetrics
	Read	DescribeAlarmHistory DescribeAlarms GetMetricData GetMetricStatistics
Config	Read	GetDiscoveredResourceCounts
Direct Connect	Read	DescribeConnections DescribeTags DescribeVirtualInterfaces
DynamoDB	List	ListTables
	Read	DescribeTable
EC2	List	DescribeAvailabilityZones DescribeImages DescribeInstances DescribeNatGateways DescribeRegions DescribeRouteTables DescribeSecurityGroups DescribeSubnets DescribeSnapshots DescribeTransitGatewayRouteTables DescribeTransitGateways DescribeTransitGatewayAttachments DescribeVolumes DescribeVpcPeeringConnections DescribeVpcs DescribeVpnGateways
	Read	DescribeVpnConnections
EC2 Auto Scaling	List	DescribeAutoScalingGroups DescribeAutoScalingInstances DescribeLaunchConfigurations
EFS	List	DescribeFileSystems
Elastic Beanstalk	List	DescribeEnvironments
	Read	DescribeConfigurationSettings DescribeEnvironmentResources DescribeEnvironmentHealth DescribeInstancesHealth
Elastic Container Services (ECS)	List	ListClusters ListContainerInstances ListServices ListTasks
	Read	DescribeClusters DescribeContainerInstances DescribeServices

Service	Actions	
		DescribeTaskDefinition DescribeTasks
ElasticCache	List	DescribeCacheClusters
Elastic Kubernetes Service (EKS)	List	ListClusters
	Read	DescribeCluster
ELB	List	DescribeLoadBalancers
	Read	DescribeTags
ELB v2	Read	DescribeListeners DescribeLoadBalancers DescribeTags DescribeTargetGroups DescribeTargetHealth
EMR	List	ListClusters
	Read	ListInstances
Glacier	List	ListTagsForVault ListVaults
	Read	GetVaultNotifications
IAM	Read	GetUser GetAccountAuthorizationDetails
IoT	List	ListThings ListTagsForResource
	Read	DescribeThing
Key Management Service (KMS)	List	ListKeys ListAliases
	Read	DescribeKey ListResourceTags
Lambda	List	ListFunctions ListAliases ListEventSourceMappings
	Read	GetAccountSettings ListTags
Lightsail	List	GetBundles GetRegions
	Read	GetInstanceMetricData GetInstances
OpsWorks	List	DescribeInstances DescribeStacks
RDS	List	DescribeDBClusters DescribeDBInstances DescribeDBSubnetGroups

Service	Actions	
	Read	ListTagsForResource
Redshift	List	DescribeClusters
	Read	DescribeLoggingStatus
Route 53	List	GetHostedZone ListHealthChecks ListHostedZones ListResourceRecordSets
S3	List	ListAllMyBuckets ListBucket
	Read	GetBucketLocation GetBucketLogging GetBucketTagging GetBucketWebsite GetObject (Restrict access to specific resources of Elastic Beanstalk. For instance, Bucket name: elasticbeanstalk-*, Any Object name.)
Shield	List	ListAttacks ListProtections
	Read	DescribeEmergencyContactSettings GetSubscriptionState
Simple Email Service (SES)	List	ListIdentities
Simple Notification Service (SES)	List	ListTopics ListSubscriptions
SQS	List	ListQueues
	Read	GetQueueAttributes
Storage Gateway	List	ListGateways ListVolumes
STS	Read	GetCallerIdentity
WAF	List	ListWebACLs
	Read	GetRateBasedRule GetRule GetRuleGroup GetWebACL
WAF Regional	List	ListResourcesForWebACL ListWebACLs
	Read	GetRateBasedRule GetRule GetRuleGroup GetWebACL
WorkSpaces	List	DescribeWorkspaces DescribeWorkspaceDirectories

To create the Minimum Permission policy:

1. Go to the AWS console and select **IAM > Policies > Create Policy**. Select **JSON** and cut and paste the following JSON document:

```
{
  "Statement": [
    {
      "Action": [
        "apigateway:GET",
        "autoscaling:DescribeAutoScalingGroups",
        "autoscaling:DescribeAutoScalingInstances",
        "autoscaling:DescribeLaunchConfigurations",
        "cloudfront:GetDistribution",
        "cloudfront:GetStreamingDistribution",
        "cloudfront:ListDistributions",
        "cloudfront:ListInvalidations",
        "cloudfront:ListStreamingDistributions",
        "cloudtrail:DescribeTrails",
        "cloudtrail:GetTrailStatus",
        "cloudwatch:DescribeAlarmHistory",
        "cloudwatch:DescribeAlarms",
        "cloudwatch:GetMetricData",
        "cloudwatch:GetMetricStatistics",
        "cloudwatch:ListMetrics",
        "config:GetDiscoveredResourceCounts",
        "directconnect:DescribeConnections",
        "directconnect:DescribeTags",
        "directconnect:DescribeVirtualInterfaces",
        "dynamodb:DescribeTable",
        "dynamodb:ListTables",
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeImages",
        "ec2:DescribeInstances",
        "ec2:DescribeNatGateways",
        "ec2:DescribeRegions",
        "ec2:DescribeRouteTables",
        "ec2:DescribeSecurityGroups",
        "ec2:DescribeSnapshots",
        "ec2:DescribeSubnets",
        "ec2:DescribeTransitGatewayAttachments",
        "ec2:DescribeTransitGatewayRouteTables",
        "ec2:DescribeTransitGateways",
        "ec2:DescribeVolumes",
        "ec2:DescribeVpcPeeringConnections",
        "ec2:DescribeVpcs",
        "ec2:DescribeVpnConnections",
        "ec2:DescribeVpnGateways",
        "ecs:DescribeClusters",
        "ecs:DescribeContainerInstances",
        "ecs:DescribeServices",
        "ecs:DescribeTaskDefinition",
        "ecs:DescribeTasks",
        "ecs:ListClusters",
```

```

"ecs:ListContainerInstances",
"ecs:ListServices",
"ecs:ListTasks",
"eks:DescribeCluster",
"eks:ListClusters",
"elasticache:DescribeCacheClusters",
"elasticbeanstalk:DescribeConfigurationSettings",
"elasticbeanstalk:DescribeEnvironmentResources",
"elasticbeanstalk:DescribeEnvironments",
"elasticbeanstalk:DescribeEnvironmentHealth",
"elasticbeanstalk:DescribeInstancesHealth",
"elasticfilesystem:DescribeFileSystems",
"elasticloadbalancing:DescribeListeners",
"elasticloadbalancing:DescribeLoadBalancers",
"elasticloadbalancing:DescribeTags",
"elasticloadbalancing:DescribeTargetGroups",
"elasticloadbalancing:DescribeTargetHealth",
"elasticmapreduce:ListClusters",
"elasticmapreduce:ListInstances",
"glacier:GetVaultNotifications",
"glacier:ListTagsForVault",
"glacier:ListVaults",
"iam:GetAccountAuthorizationDetails",
"iam:GetUser",
"iot:DescribeThing",
"iot:ListTagsForResource",
"iot:ListThings",
"kms:DescribeKey",
"kms:ListAliases",
"kms:ListKeys",
"kms:ListResourceTags",
"lambda:GetAccountSettings",
"lambda:ListAliases",
"lambda:ListEventSourceMappings",
"lambda:ListFunctions",
"lambda:ListTags",
"lightsail:GetBundles",
"lightsail:GetInstanceMetricData",
"lightsail:GetInstances",
"lightsail:GetRegions",
"opsworks:DescribeInstances",
"opsworks:DescribeStacks",
"rds:DescribeDBClusters",
"rds:DescribeDBInstances",
"rds:DescribeDBSubnetGroups",
"rds:ListTagsForResource",
"redshift:DescribeClusters",
"redshift:DescribeLoggingStatus",
"route53:GetHostedZone",
"route53:ListHealthChecks",
"route53:ListHostedZones",
"route53:ListResourceRecordSets",
"s3:GetBucketLocation",
"s3:GetBucketLogging",
"s3:GetBucketTagging",
"s3:GetBucketWebsite",

```

```

        "s3:GetObject",
        "s3:ListAllMyBuckets",
        "s3:ListBucket",
        "ses:ListIdentities",
        "shield:DescribeEmergencyContactSettings",
        "shield:GetSubscriptionState",
        "shield:ListAttacks",
        "shield:ListProtections",
        "sns:ListSubscriptions",
        "sns:ListTopics",
        "sqs:GetQueueAttributes",
        "sqs:ListQueues",
        "ssm:GetParameters",
        "storagegateway:ListGateways",
        "storagegateway:ListVolumes",
        "sts:GetCallerIdentity",
        "tag:Get*",
        "waf-regional:GetRateBasedRule",
        "waf-regional:GetRule",
        "waf-regional:GetRuleGroup",
        "waf-regional:GetWebACL",
        "waf-regional:ListResourcesForWebACL",
        "waf-regional:ListWebACLs",
        "waf:GetRateBasedRule",
        "waf:GetRule",
        "waf:GetRuleGroup",
        "waf:GetWebACL",
        "waf:ListWebACLs",
        "workspaces:DescribeWorkspaces",
        "workspaces:DescribeWorkspaceDirectories"
    ],
    "Effect": "Allow",
    "Resource": "*",
    "Sid": "VisualEditor0"
  }
],
  "Version": "2012-10-17"
}

```

2. Click **[Next: Tags]**. If applicable, enter your Tags.
3. Click **[Next: Review]**. Name the policy "SL1 MinimumPermissions" and click **[Create Policy]**.

This policy needs to be available in each account that is to be monitored and will be referenced in the following sections.

5

AWS Discovery

Overview

The following sections describe the different methods of discovery that can be used with the Amazon Web Services PowerPack:

<i>Discovering Amazon Web Services</i>	52
<i>Manual Discovery</i>	53
<i>Configuring a User in AWS</i>	53
<i>Creating the SOAP/XML Credential for AWS</i>	57
<i>Creating the SOAP/XML Credential for AWS in the SL1 Classic User Interface</i>	59
<i>Creating an AWS Virtual Device for Discovery in the SL1 Classic User Interface</i>	62
<i>Aligning the Discovery Dynamic Application</i>	62
<i>Aligning the Discovery Dynamic Application in the SL1 Classic User Interface</i>	64
<i>Automated Discovery Using AssumeRole with a Single IAM Key from the AWS Master Account</i>	65
<i>Configure a User in the Master Billing Account</i>	66
<i>Create a Role in Each Account</i>	67
<i>Configure the SL1 Credential</i>	68
<i>Configure the SL1 Credential in the SL1 Classic User Interface</i>	71
<i>Create and Run the Discovery Session</i>	73
<i>Create and Run the Discovery Session in the SL1 Classic User Interface</i>	75
<i>Manually Creating the Organization and Aligning Dynamic Applications</i>	77
<i>Automated Discovery when the Data Collector Runs as an EC2 Instance</i>	78
<i>Create a Role in the Master Billing Account</i>	78
<i>Create an AWS Role in the Account your Data Collector is In</i>	80
<i>Create a Role in Each Account</i>	81
<i>Configuring the Credential to Discover AWS on an EC2 Collector</i>	82

Configuring the Credential to Discover AWS on an EC2 Collector in the SL1 Classic User Interface	83
Create and Run the Discovery Session	85
Create and Run the Discovery Session in the SL1 Classic User Interface	88
AWS Guided Discovery	89
Defining an AWS Assume Role Credential	91
Defining an AWS EC2 Credential	93
Defining an AWS IAM Credential	96
Completing the Discovery Session	98

Discovering Amazon Web Services

SL1 currently supports the following methods to discover your AWS accounts:

- **Manual Discovery.** Requires the creation of a virtual device, manual alignment of Dynamic Applications, and an IAM key. This process needs to be repeated for each AWS account.
- **Automated Discovery using Assume Role with single IAM key from Master Account.** Provides an automated mechanism to discover all your AWS accounts within an organization using a single IAM key. This is the recommended method of discovery when your Data Collector is not an EC2 instance.
- **Automated Discovery when the Data Collector runs as an EC2 instance.** Provides a fully automated mechanism to discover all your AWS accounts when your Data Collectors are running as EC2 instances. SL1 does not need any AWS credentials in this case. This is the recommended approach when your Data Collectors are EC2 instances.
- **AWS Guided Discovery.** Uses guided workflows in SL1. This method is recommended when you want to use a separate IAM key for each AWS account. The guided workflows provide a more user-friendly version of the manual process. Choose from the following workflows:
 - **AWS EC2**
 - **AWS IAM**
 - **AWS Assume Role**

NOTE: These Guided Discovery Workflows are available in SL1 version 11.2.0 and later. A basic Guided Discovery Workflow is available in earlier versions of SL1.

Before determining your method of discovery, it is recommended to define the minimum permissions policy in AWS. This policy defines the minimum permissions needed to monitor all AWS services and is needed regardless of which of the above methods is used.

You can discover a maximum of 10 accounts with the following requirements on the Data Collector:

- 8 cores
- 32 GB of RAM

- 100 GB of HDD

Manual Discovery

Manual discovery is used to discover a single AWS account at a time and requires an IAM key for the account.

NOTE: Using one of the Assume Role methods of discovery is recommended.

The process consists of the following steps:

1. [Configure a user in the AWS Account](#)
2. [Configure the SL1 Credential](#)
3. [Create a Virtual Device](#)
4. [Align the Discovery Dynamic Application](#)

Configuring a User in AWS

To create a read-only user account in AWS, perform the following steps:

1. Open a browser session and go to aws.amazon.com.

2. Click **[My Account]** and then select *AWS Management Console*. If you are not currently logged in to the AWS site, you will be prompted to log in:



Sign In or Create an AWS Account

What is your e-mail or mobile number?

E-mail or mobile number:

I am a new user.

I am a returning user and my password is:

[Forgot your password?](#)



Now Available
Amazon Aurora
Enterprise-class database at 1/10th the cost

Learn more about [AWS Identity and Access Management](#) and [AWS Multi-Factor Authentication](#), features that provide additional security for your AWS Account. View full [AWS Free Usage Tier](#) offer terms.

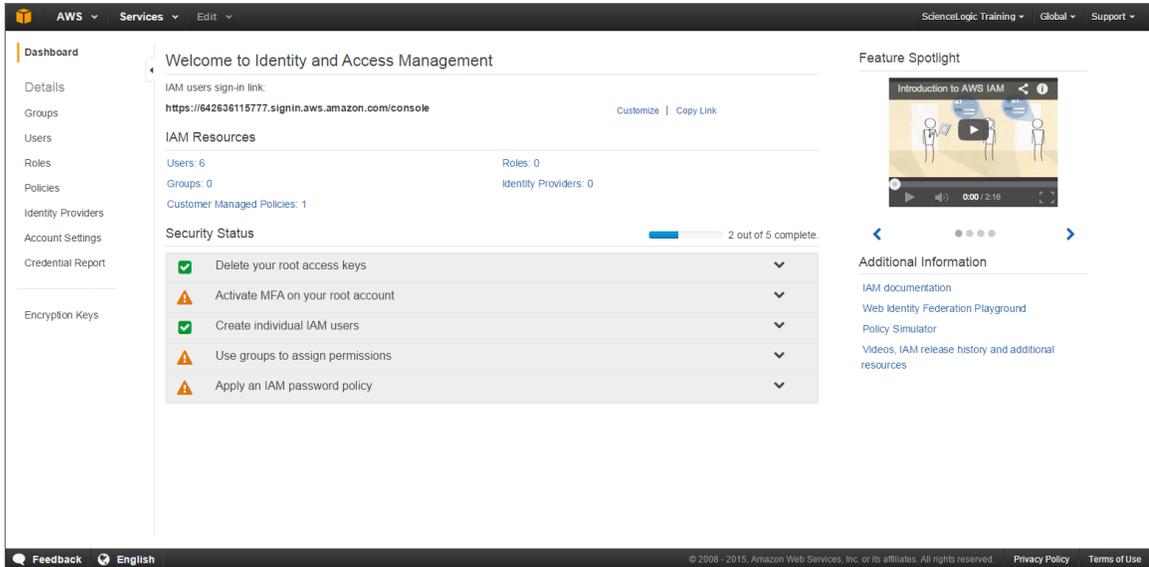
About Amazon.com Sign In

Amazon Web Services uses information from your Amazon.com account to identify you and allow access to Amazon Web Services. Your use of this site is governed by our [Terms of Use](#) and [Privacy Policy](#) linked below.

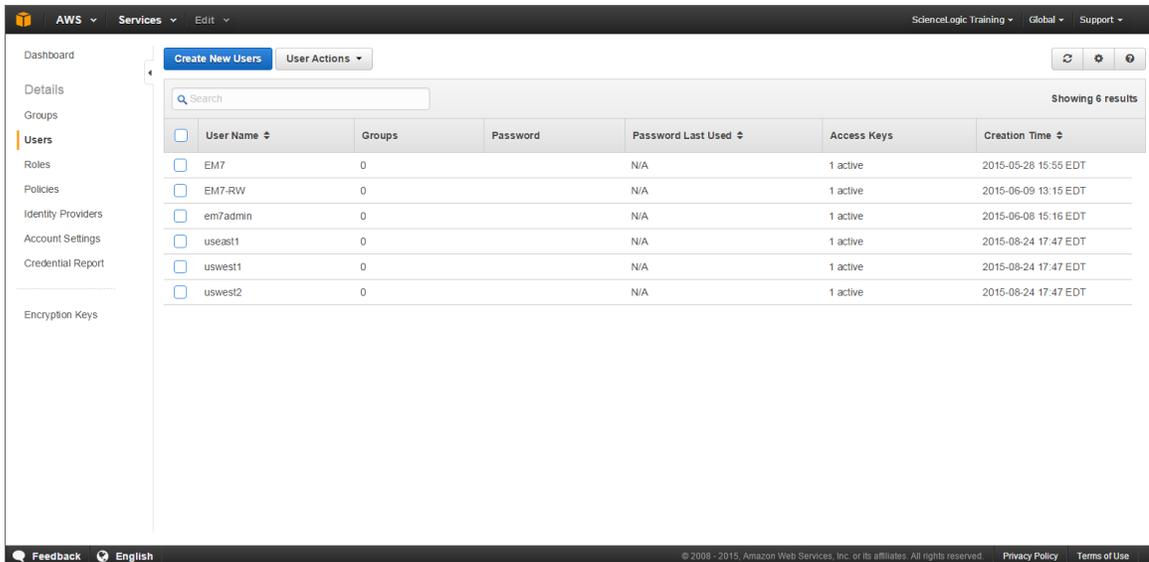
[Terms of Use](#) [Privacy Policy](#) © 1996-2015, Amazon.com, Inc. or its affiliates

An **amazon.com** company

3. In the **AWS Management Console**, under the **Security & Identity** heading, click [**Identity & Access Management**].
4. After logging in, the **Identity & Access Management Dashboard** page appears:



5. To create a user account for SL1, click [**Users**] on the Dashboard menu.

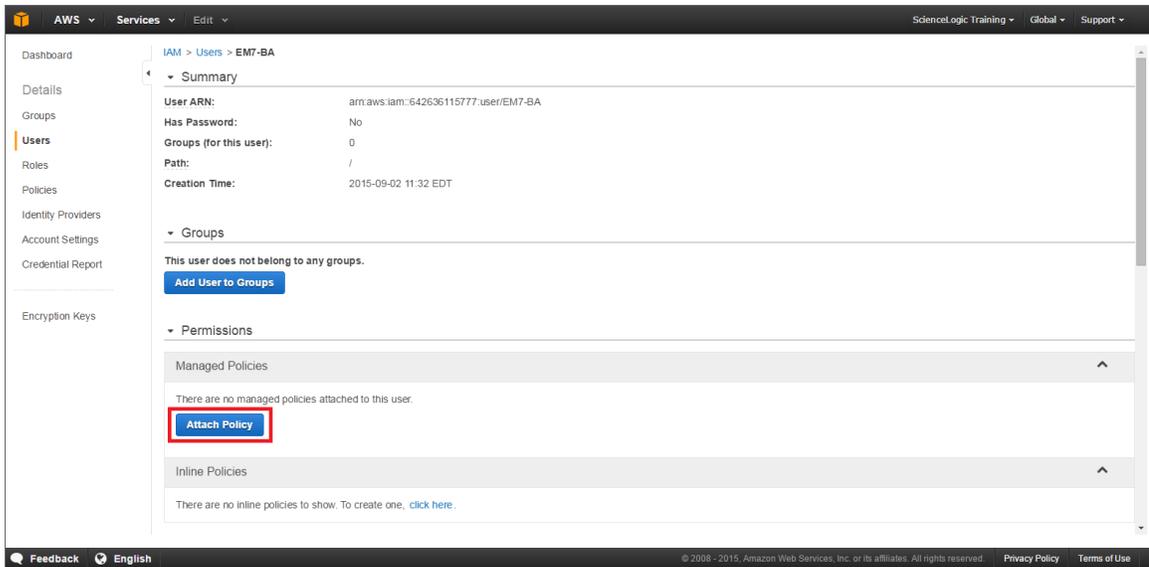


6. Click the [**Create New Users**] button.
7. Enter a username for the new user, e.g. "SL1", and make sure the **Generate an access key for each user** checkbox is selected.

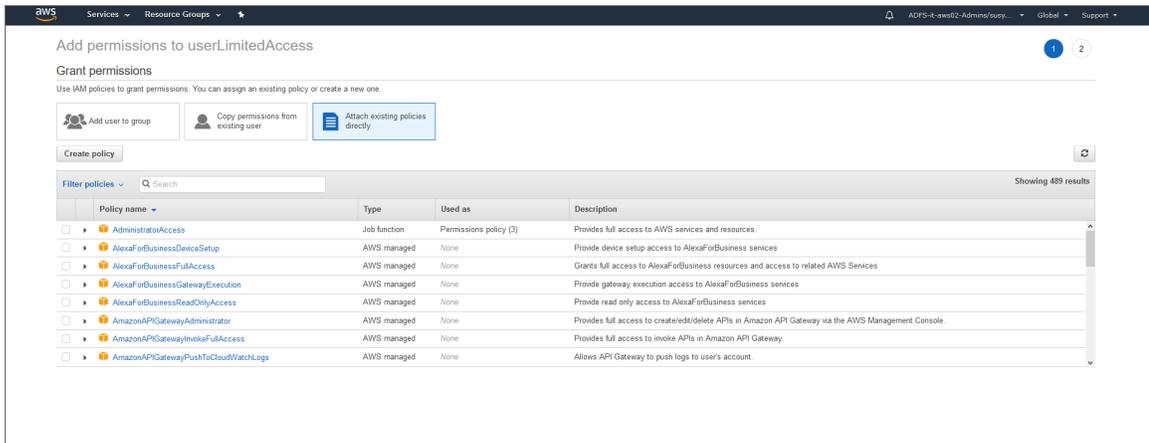
8. Click the **[Create]** button to generate your user account. The **Create User** page appears:



9. Click the **[Download Credentials]** button to save your Access Key ID and Secret Key as a CSV (comma-separated value) text file, and then click **[Close]**.
10. After creating a user, you must assign it a set of permissions policies. Click the username of the user account you created. The user's account information appears:



- Under the **Permissions** heading, click the **[Attach existing policies directly]** button. The **Add permissions** page appears:



- Select the checkbox for your policy based on the definition of the minimum required permissions described in the [Minimum Permissions for Dynamic Applications](#) section.
- Click the **[Attach Policy]** button.

Creating the SOAP/XML Credential for AWS

To discover AWS using the manual discovery method, you must first define an AWS credential in SL1.

NOTE: If you are using an SL1 system prior to version 11.1.0, the new user interface does not include the **Duplicate** option for sample credential(s). ScienceLogic recommends that you use [the classic user interface and the Save As button](#) to create new credentials from sample credentials. This will prevent you from overwriting the sample credential(s).

To define an AWS credential:

- Go to the **Credentials** page (Manage > Credentials).

2. Locate the **AWS Credential** sample credential, click its **[Actions]** icon (**⋮**) and select **Duplicate**. A copy of the credential, called **AWS Credential copy** appears.
3. Click the **[Actions]** icon (**⋮**) for the **AWS Credential copy** credential and select **Edit**. The **Edit Credential** modal page appears:

4. Supply values in the following fields:
 - **Name**. Type a new name for your AWS credential.
 - **All Organizations**. Toggle on (blue) to align the credential to all organizations, or toggle off (gray) and then select one or more specific organizations from the **What organization manages this service?** drop-down field to align the credential with those specific organizations.
 - **Timeout (ms)**. Keep the default value.
 - **URL**. Enter a valid URL. This field is not used for this discovery method but must be populated with a valid URL for discovery to complete.
 - **HTTP Auth User**. Type your **Access Key ID**.
 - **HTTP Auth Password**. Type your **Secret Access Key**. The characters appear as asterisks to protect your password privacy.
 - **Proxy Hostname/IP**. Type the host name or IP address of the proxy server.

NOTE: The proxy fields are required only if you are discovering AWS services through a proxy server. Otherwise, leave these fields blank.

- **Proxy Port**. Type the port on the proxy server to which you will connect.
- **Proxy User**. Type the username used to access the proxy server.
- **Proxy Password**. Type the password used to access the proxy server.

CAUTION: If you are creating a credential from the **AWS Credential - Proxy** example and the proxy server does not require a username and password, then the **Proxy User** and **Proxy Password** fields must both be blank. In that scenario, if you leave the "<Proxy_User>" text in the **Proxy User** field, SL1 cannot properly discover your AWS services.

- **Embed Value [%1].** Do one of the following:
 - To monitor a GovCloud account, type "us-gov-west-1" or "us-gov-east-1".
 - To monitor the Beijing region, type "cn-north-1".
 - To monitor the Ningxia region, type "cn-northwest-1".

Otherwise, leave this field blank.

NOTE: If you are monitoring both the Beijing and Ningxia regions, you must create a unique credential for each region.

- **Embed Value [%2]:**
 - If you are using the AWS Config service and want to discover only regions that have that service enabled, type "[AUTO]" in this field. After discovery, only regions that have AWS Config enabled will be displayed in the dynamic component map tree. Global resources will also be discovered.
 - If you are using not using the AWS Config service, type "[FILTER]" in this field so it will discover only regions that are reporting CloudWatch metrics. This will reduce the number of regions being monitored and the load on the Data Collector.

CAUTION: If you are performing discovery using [AUTO] or [FILTER] in the **Embed Value [%2]** field, the status of regions that don't meet these requirements will change to *Unavailable* and vanish if enabled.

NOTE: If you are performing discovery based on the AWS Config service and do not have any regions with the AWS Config service enabled, the Amazon Web Services PowerPack will discover all regions that have resources.

5. Click the **[Save& Close]** button.

Creating the SOAP/XML Credential for AWS in the SL1 Classic User Interface

To discover AWS using the manual discovery method, you must first define an AWS credential in SL1.

To define an AWS credential:

1. Go to the **Credential Management** page (System > Manage > Credentials).
2. Locate the **AWS Credential** sample credential and click its wrench icon (🔧). The **Credential Editor** modal page appears:

The screenshot shows the 'Credential Editor [77]' window. The title bar includes 'New' and 'Reset' buttons. The main content is divided into several sections:

- Basic Settings:** Includes fields for Profile Name (AWS Credential), Content Encoding (text/xml), Method (POST), HTTP Version (HTTP/1.1), URL (http://example.com/), HTTP Auth User (AWS Account Access Key), HTTP Auth Password, and Timeout (seconds) (2).
- Proxy Settings:** Includes fields for Hostname/IP, Port (0), User, and Password.
- CURL Options:** A list of options (CAINFO, CAPATH, CLOSEPOLICY, CONNECTTIMEOUT, COOKIE, COOKIEFILE, COOKIEJAR, COOKIELIST, CRLF, CUSTOMREQUEST, DNSCACHETIMEOUT) with left and right arrow buttons.
- Soap Options:** Includes an Embedded Password [%P] field and four Embed Value [%1] through [%4] fields.
- HTTP Headers:** A section with a '+ Add a header' button.

At the bottom, there are 'Save' and 'Save As' buttons.

3. Enter values in the following fields:

Basic Settings

- **Profile Name.** Type a new name for your AWS credential.
- **URL.** Enter a valid URL. This field is not used for this discovery method but must be populated with a valid URL for discovery to complete.
- **HTTP Auth User.** Type your **Access Key ID**.
- **HTTP Auth Password.** Type your **Secret Access Key**. The characters appear as asterisks to protect your password privacy.

Proxy Settings

NOTE: The **Proxy Settings** fields are required only if you are discovering AWS services through a proxy server. Otherwise, leave these fields blank.

- **Hostname/IP.** Type the host name or IP address of the proxy server.

- **Port.** Type the port on the proxy server to which you will connect.
- **User.** Type the username used to access the proxy server.
- **Password.** Type the password used to access the proxy server.

CAUTION: If you are creating a credential from the **AWS Credential - Proxy** example and the proxy server does not require a username and password, then the **User** and **Password** fields must both be blank. In that scenario, if you leave the "<Proxy_User>" text in the **User** field, SL1 cannot properly discover your AWS services.

SOAP Options

- **Embed Value [%1].** Do one of the following:
 - To monitor a GovCloud account, type "us-gov-west-1" or "us-gov-east-1".
 - To monitor the Beijing region, type "cn-north-1".
 - To monitor the Ningxia region, type "cn-northwest-1".

Otherwise, leave this field blank.

NOTE: If you are monitoring both the Beijing and Ningxia regions, you must create a unique credential for each region.

- **Embed Value [%2]:**
 - If you are using the AWS Config service and want to discover only regions that have that service enabled, type "[AUTO]" in this field. After discovery, only regions that have AWS Config enabled will be displayed in the dynamic component map tree. Global resources will also be discovered.
 - If you are using not using the AWS Config service, type "[FILTER]" in this field so it will discover only regions that are reporting CloudWatch metrics. This will reduce the number of regions being monitored and the load on the Data Collector.

CAUTION: If you are performing discovery using [AUTO] or [FILTER] in the **Embed Value [%2]** field, the status of regions that don't meet these requirements will change to *Unavailable* and vanish if enabled.

NOTE: If you are performing discovery based on the AWS Config service and do not have any regions with the AWS Config service enabled, the *Amazon Web Services PowerPack* will discover all regions that have resources.

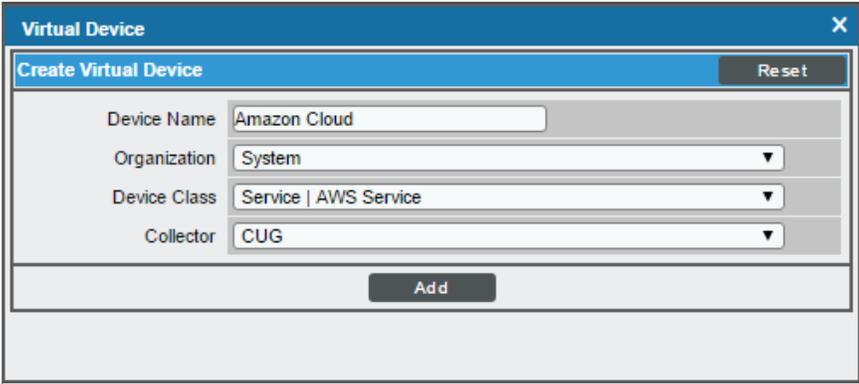
4. Click the **[Save As]** button, and then click **[OK]**.

Creating an AWS Virtual Device for Discovery in the SL1 Classic User Interface

Because the Amazon Web Service does not have a specific IP address, you cannot discover an AWS device using discovery. Instead, you must create a **virtual device** that represents the Amazon Web Service. A virtual device is a user-defined container that represents a device or service that cannot be discovered by SL1. You can use the virtual device to store information gathered by policies or Dynamic Applications.

To create a virtual device that represents your Amazon service:

1. Go to the **Device Manager** page (Devices > Device Manager or Registry > Devices > Device Manager in the SL1 classic user interface).
2. Click the **[Actions]** button, then select *Create Virtual Device*. The **Virtual Device** modal page appears:



The screenshot shows a modal window titled "Virtual Device" with a close button (X) in the top right corner. Inside the modal, there is a sub-header "Create Virtual Device" and a "Reset" button. Below this, there are four input fields: "Device Name" (text input with "Amazon Cloud"), "Organization" (dropdown menu with "System"), "Device Class" (dropdown menu with "Service | AWS Service"), and "Collector" (dropdown menu with "CUG"). At the bottom center of the form is an "Add" button.

3. Enter values in the following fields:
 - **Device Name.** Enter a name for the device. For example, you could enter "Amazon Cloud" in this field.
 - **Organization.** Select the organization for this device. The organization the device is associated with limits the users that will be able to view and edit the device.
 - **Device Class.** Select *Service | AWS Service*.
 - **Collector.** Select the collector group that will monitor the device.
4. Click the **[Add]** button to create the virtual device.

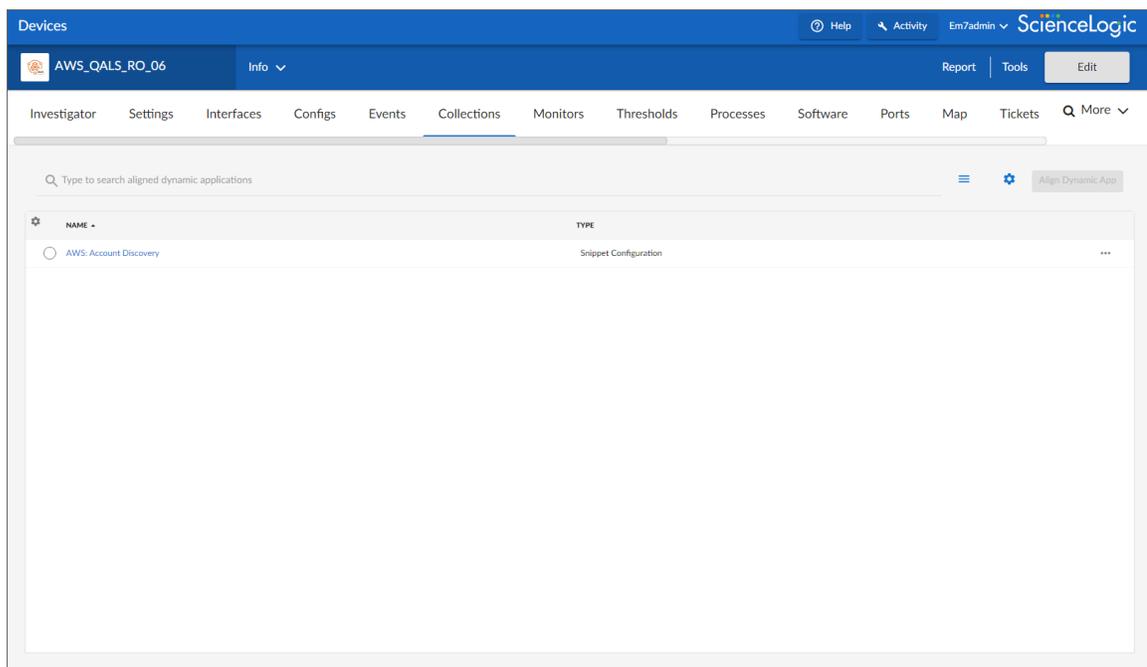
Aligning the Discovery Dynamic Application

To discover your AWS account, you must manually align the "AWS: Account Discovery" Dynamic Application with the AWS virtual device. After you do so, the other Dynamic Applications in the *Amazon Web Services PowerPack* will automatically align to discover and monitor all of the components in your AWS account.

TIP: If your AWS account includes API Gateways or Lambda services to be monitored and you want SL1 to put those component devices in a "vanished" state if the platform cannot retrieve data about them for a specified period of time, ScienceLogic recommends setting the **Component Vanish Timeout Mins.** field to at least 120 minutes. For more information, see the chapter on "Vanishing and Purging Devices" in the **Device Management** manual.

To align the "AWS: Account Discovery" Dynamic Application to your virtual device:

1. Go to the **Devices** page.
2. Click the AWS virtual device and click on it to open the **Device Investigator**.
3. In the **Device Investigator**, click the **[Collections]** tab. The **Dynamic Application Collections** page appears.
4. Click the **[Edit]** button and then click the **[Align Dynamic App]** button.
5. In the **Align Dynamic Application** page, click *Choose Dynamic Application*.
6. In the **Choose Dynamic Application** page, locate the *credential you created for your AWS service* and select it.
7. Click the **[Select]** button and then click the **[Align Dynamic App]** button.



Aligning the Discovery Dynamic Application in the SL1 Classic User Interface

To discover your AWS account, you must manually align the "AWS: Account Discovery" Dynamic Application with the AWS virtual device. After you do so, the other Dynamic Applications in the *Amazon Web Services PowerPack* will automatically align to discover and monitor all of the components in your AWS account.

TIP: If your AWS account includes API Gateways or Lambda services to be monitored and you want SL1 to put those component devices in a "vanished" state if the platform cannot retrieve data about them for a specified period of time, ScienceLogic recommends setting the **Component Vanish Timeout Mins.** field to at least 120 minutes. For more information, see the chapter on "Vanishing and Purging Devices" in the **Device Management** manual.

To align the "AWS: Account Discovery" Dynamic Application to your virtual device:

1. Go to the **Device Manager** page (Registry > Devices > Device Manager).
2. Click the wrench icon () for your virtual device.
3. In the **Device Administration** panel, click the **[Collections]** tab. The **Dynamic Application Collections** page appears.
4. Click the **[Actions]** button, and then select *Add Dynamic Application* from the menu.
5. In the **Dynamic Application Alignment** modal page, select *AWS: Account Discovery* in the **Dynamic Applications** field.
6. In the **Credentials** field, select the [credential you created for your AWS service](#).

7. Click the **[Save]** button to align the Dynamic Application.

The screenshot displays a web-based configuration interface for a monitoring application. At the top, there are several tabs: Close, Properties, Thresholds, Collections (selected), Monitors, Schedule, Logs, Toolbox, Interfaces, Relationships, Tickets, Redirects, and Notes. Below the tabs, the configuration is organized into two columns. The left column contains fields for Device Name (Amazon Cloud), ID (1651), Class (Service), Organization (System), and Device Hostname. The right column contains fields for Managed Type (Virtual Device), Category (Cloud.Service), Sub-Class (AWS Service), Uptime (0 days, 00:00:00), and Group / Collector (CUG | em7_a0). To the right of these fields is a 'Service' icon with a wrench and the text 'Amazon Cloud'. Below the configuration fields is a table titled 'Dynamic Application™ Collections | Application Added'. The table has columns for ID, Poll Frequency, Type, and Credential. A single row is visible with ID 32, Poll Frequency 5 mins, Type Snippet Configuration, and Credential Amazon Web Services Credential. At the bottom of the interface, there is a '[Select Action]' dropdown menu and a 'Go' button. A 'Save' button is located at the very bottom center of the interface.

Automated Discovery Using AssumeRole with a Single IAM Key from the AWS Master Account

Automated discovery using AssumeRole with an IAM key is the recommended approach to monitor your AWS accounts when your Data Collectors are **not** acting as EC2 instances. In this method of discovery, your organization will be discovered first and then the accounts within the organization will be created automatically.

This method of discovery has the following benefits:

- Only a single IAM key needs to be managed on SL1, instead of an IAM key for every AWS account.
- The IAM key is only used to get the information about the organization, and all the actual monitoring is done via temporary tokens, which is the recommended approach by AWS.

This method can also be used in the following scenarios:

- When a proxy server is between the Data Collector and the AWS cloud
- When Ping is not available
- In the Government cloud

NOTE: All examples shown are for commercial AWS accounts. When AWS Gov is being monitored, the JSON data that refers to ARN will need to be modified from "aws" to "aws-us-gov". For example:
Resource": "arn:aws:iam::<account number>:role/Sciencelogic-Monitor would need to be Resource": "arn:aws:iam-us-gov::<account number>:role/Sciencelogic-Monitor

To use this method of discovery, perform the following steps:

1. [Configure a user in the master billing account](#)
2. [Create a role in each account](#)
3. [Configuring the SL1 credential](#)
4. [Create and run the discovery session](#)

NOTE: If Ping is blocked, then you must follow the steps in the [Manually Create the Organization and Align the Dynamic Applications](#) section.

Configure a User in the Master Billing Account

The first step in this discovery method is to create a policy that defines the permissions needed by SL1. To do this, copy the policy below into an editor:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "organizations:ListAccounts",
        "organizations:DescribeOrganization",
        "organizations:DescribeAccount"
      ],
      "Resource": "*"
    },
    {
      "Sid": "VisualEditor1",
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": "arn:aws:iam::<account number>:role/Sciencelogic-Monitor"
    }
  ]
}
```

For each account that needs to be monitored, duplicate the "Resource": "arn:aws:iam::<Account Number>:role/Sciencelogic-Monitor" line and set the <Account Number> to the correct account number.

After editing the policy, perform the following steps in the AWS console:

1. Go to **IAM > Policies > Create Policy**. Select the **JSON** tab and copy the edited JSON text into the AWS console.
2. Click **Next: Tags** and then click **Next: Review**.
3. Type a name for the policy (for example, "SL1MasterBillingPermissions") and then select **[Create Policy]**.
4. To create a user in the master billing account, go to **IAM > Users > Add User**.
5. Type the user's name and select the option for **Programmatic Access**. Click **[Next: Permissions]**.
6. Select **Attach existing policies directly** and select the checkbox for the policy you created.
7. Select **Next: Tags > Next: Review > Create User**.

NOTE: The Access Key and Secret Key need to be saved as these will be needed when configuring the SL1 credential.

Create a Role in Each Account

In every AWS account that is to be monitored, a role with the **same name** needs to be created. The default name is "ScienceLogic-Monitor". To create the role, perform the following steps for each account that is to be monitored:

1. In the AWS console, go to **IAM > Roles** and select **Create Role**.
2. Select **Another AWS Account** and enter the account ID of the Master Billing Account. Select **Next: Permissions**.
3. Select the policy that was created in the [Minimum Permissions Needed to Monitor Your AWS Accounts](#) section.
4. Select **Next: Tags** and then **Next: Review**.
5. Enter "ScienceLogic-Monitor" in the **Role name** field and then select **[Create role]**.
6. Repeat these steps for each AWS account that you want to monitor.

Next you will need to edit the trust relationship of the role to restrict the principle to the user you created. To do this:

1. In the AWS console, go to **IAM > Roles** and select the "ScienceLogic-Monitor" role.
2. Select the **Trust Relationships** tab and click **[Edit trust relationship]**.
3. Edit the JSON to look like the following:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": [
        "AWS": "arn:aws:iam::<Master Billing Account>:user/<Master Billing Account
```

```
User>"  
  },  
  
  {  
    "Action": "sts:AssumeRole",  
    "Condition": {}  
  }  
]  
}
```

NOTE: The ARN above is the ARN of the user that was created in the previous steps.

4. Once you have updated the policy, click [**Update Trust Policy**].

Configure the SL1 Credential

You can use your master organization account to automatically discover all AWS accounts, instead of having to enter a key for each account. This process will also create a separate DCM tree for each account.

NOTE: If you are using an SL1 system prior to version 11.1.0, the new user interface does not include the **Duplicate** option for sample credential(s). ScienceLogic recommends that you use [the classic user interface and the Save As button](#) to create new credentials from sample credentials. This will prevent you from overwriting the sample credential(s).

NOTE: Ensure that you use the "AWS Credential - Master Account" credential, as this credential is valid for AssumeRole and has the correct headers for AssumeRole discovery. Do not use the classic "AWS Credential" to discover an AssumeRole pingable device, as it will not work.

NOTE: Discovery of China accounts does not support alignment using AssumeRole. For those accounts customers must continue to use manual alignment of Dynamic Applications.

To define the credential:

1. Go to the **Credentials** page (Manage > Credentials).

2. Locate the **AWS Credential - Master Account** sample credential, click its **[Actions]** icon (☰) and select **Duplicate**. A copy of the credential, called **AWS Credential - Master Account copy** appears.
3. Click the **[Actions]** icon (☰) for the **AWS Credential - Master Account copy** credential and select **Edit**. The **Edit Credential** modal page appears:

4. Enter values in the following fields:
 - **Name**. Type a new name for your AWS credential.
 - **All Organizations**. Toggle on (blue) to align the credential to all organizations, or toggle off (gray) and then select one or more specific organizations from the **What organization manages this service?** drop-down field to align the credential with those specific organizations.
 - **URL**. Type `https://organizations.us-east-1.amazonaws.com` in the field. If your administrator has configured a different region, you can change it or use the default region. To discover Gov accounts using AssumeRole, type `https://organizations.us-gov-west-1.amazonaws.com`
 - **HTTP Auth User**. Type the AWS access key ID of the user you created in the master account.
 - **HTTP Auth Password**. Type the AWS secret access key of the user created in the master account.

- **Embed Value [%2]:**
 - If you are using the AWS Config service and want to discover only regions that have that service enabled, type "[AUTO]" in this field. After discovery, only regions that have AWS Config enabled will be displayed in the dynamic component map tree. Global resources will also be discovered.
 - If you are using not using the AWS Config service, type "[FILTER]" in this field so it will discover only regions that are reporting CloudWatch metrics. This will reduce the number of regions being monitored and the load on the Data Collector.
- Under **HTTP Headers**, you can edit the following options:
 - *AssumeRole*. Type the AWS Role you created in each account. The default name is "ScienceLogic-Monitor".
 - *AssumeRoleSession*. Optional. The default value is "AssumeRoleSession:SL1".
 - *Regions*. The regions entered in this field will be discovered. For example, entering "Regions:ap-southeast-2, us-east-2" will discover two regions. If left blank, all regions will be discovered. The default value is "Regions:ALL".
 - *OrganizationCreation:NAME:ID*. Autocreates an SL1 organization for accounts using AssumeRole. You can enter one of the following options:
 - **OrganizationCreation:NAME**. The name of the organization will contain the name of the user.
 - **OrganizationCreation:ID**. The name of the organization will contain the ID of the user.
 - **OrganizationCreation:ID:NAME**. The name of the organization will contain both the ID and name of the user, in that order.
 - **OrganizationCreation:NAME:ID**. The name of the organization will contain both the name and ID of the user, in that order.

NOTE: The existing organization will be changed by this setting only if it is the default (System) organization. If this header is not included, then **all** the discovered accounts will be placed into the organization selected in the discovery session.

5. Click the **[Save & Close]** button.

NOTE: If the "AWS: Account Creation" Dynamic Application is reporting that it is unable to use your AssumeRole, double-check your trust relationships on your configured roles.

Configure the SL1 Credential in the SL1 Classic User Interface

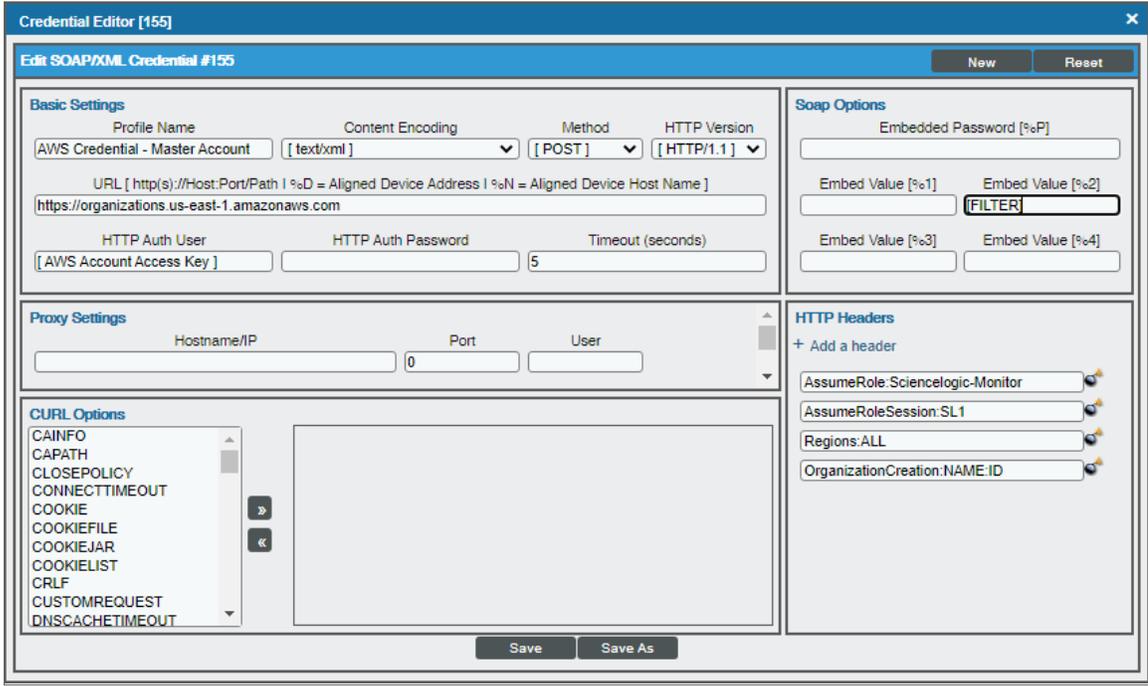
You can use your master organization account to automatically discover all AWS accounts, instead of having to enter a key for each account. This process will also create a separate DCM tree for each account.

NOTE: Ensure that you use the "AWS Credential - Master Account" credential, as this credential is valid for AssumeRole and has the correct headers for AssumeRole discovery. Do not use the classic "AWS Credential" to discover an AssumeRole pingable device, as it will not work.

NOTE: Discovery of China accounts does not support alignment using AssumeRole. For those accounts customers must continue to use manual alignment of Dynamic Applications.

To define the credential:

1. Go to the **Credential Management** page (System > Manage > Credentials).
2. Locate the **AWS Credential - Master Account** sample credential that you need and click its wrench icon (). The **Credential Editor** modal page appears:



The screenshot shows the 'Credential Editor [155]' window. The title bar indicates 'Edit SOAP/XML Credential #155'. The interface is divided into several sections:

- Basic Settings:** Profile Name (AWS Credential - Master Account), Content Encoding (text/xml), Method (POST), HTTP Version (HTTP/1.1), URL (https://organizations.us-east-1.amazonaws.com), HTTP Auth User (AWS Account Access Key), HTTP Auth Password, and Timeout (5 seconds).
- Proxy Settings:** Hostname/IP, Port (0), and User.
- CURL Options:** A list of options including CAINFO, CAPATH, CLOSEPOLICY, CONNECTTIMEOUT, COOKIE, COOKIEFILE, COOKIEJAR, COOKIELIST, CRLF, CUSTOMREQUEST, and DNSCACHETIMEOUT.
- Soap Options:** Embedded Password [%P], Embed Value [%1] (with a FILTER button), Embed Value [%2], Embed Value [%3], and Embed Value [%4].
- HTTP Headers:** A list of headers including AssumeRole: Sciencelogic-Monitor, AssumeRoleSession: SL1, Regions: ALL, and OrganizationCreation: NAME:ID.

Buttons for 'New', 'Reset', 'Save', and 'Save As' are visible at the bottom of the window.

3. Enter values in the following fields:

Basic Settings

- **Profile Name.** Type a new name for your AWS credential.
- **URL.** Type `https://organizations.us-east-1.amazonaws.com` in the field. If your administrator has configured a different region, you can change it or use the default region. To discover Gov accounts using AssumeRole, type `https://organizations.us-gov-west-1.amazonaws.com`
- **HTTP Auth User.** Type the AWS access key ID of the user you created in the master account.
- **HTTP Auth Password.** Type the AWS secret access key of the user created in the master account.

SOAP Options

- **Embed Value [%2]:**
 - If you are using the AWS Config service and want to discover only regions that have that service enabled, type "[AUTO]" in this field. After discovery, only regions that have AWS Config enabled will be displayed in the dynamic component map tree. Global resources will also be discovered.
 - If you are using not using the AWS Config service, type "[FILTER]" in this field so it will discover only regions that are reporting CloudWatch metrics. This will reduce the number of regions being monitored and the load on the Data Collector.

HTTP Headers

- Click + **Add a header** to add a header field. You can enter the following options:
 - **AssumeRole.** Type the AWS Role you created in each account. The default name is "ScienceLogic-Monitor".
 - **AssumeRoleSession.** Optional. The default value is "AssumeRoleSession:SL1".
 - **Regions.** The regions entered in this field will be discovered. For example, entering "Regions:ap-southeast-2, us-east-2" will discover two regions. If left blank, all regions will be discovered. The default value is "Regions:ALL".
 - **OrganizationCreation:NAME:ID.** Autocreates an SL1 organization for accounts using AssumeRole. You can enter one of the following options:
 - **OrganizationCreation:NAME.** The name of the organization will contain the name of the user.
 - **OrganizationCreation:ID.** The name of the organization will contain the ID of the user.
 - **OrganizationCreation:ID:NAME.** The name of the organization will contain both the ID and name of the user, in that order.
 - **OrganizationCreation:NAME:ID.** The name of the organization will contain both the name and ID of the user, in that order.

NOTE: The existing organization will be changed by this setting only if it is the default (System) organization. If this header is not included, then **all** the discovered accounts will be placed into the organization selected in the discovery session.

4. Click the **[Save As]** button, and then click **[OK]**.

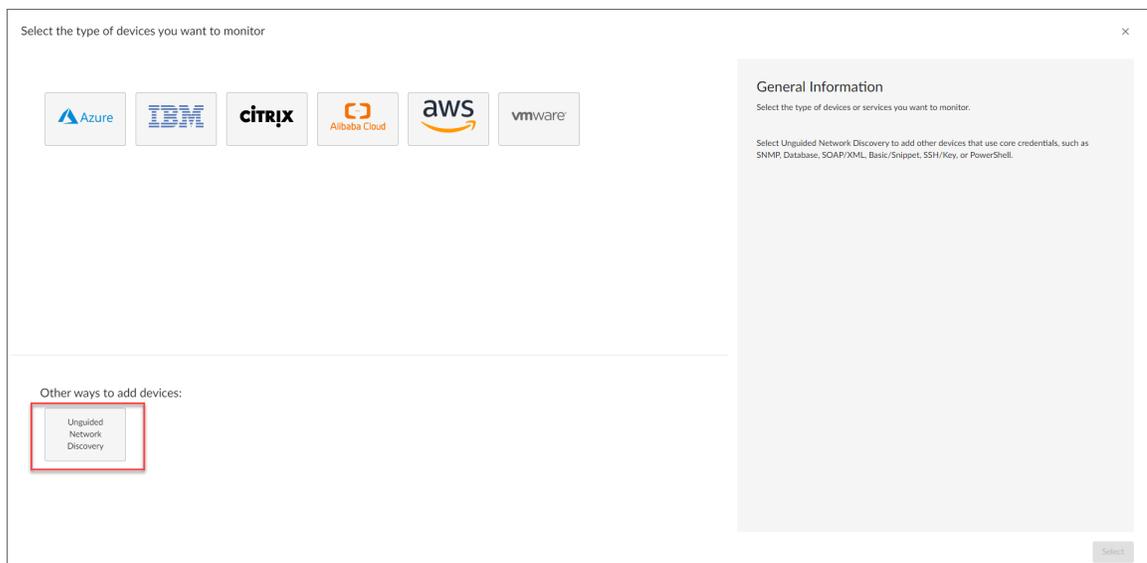
NOTE: If the "AWS: Account Creation" Dynamic Application is reporting that it is unable to use your AssumeRole, double-check your trust relationships on your configured roles.

Create and Run the Discovery Session

To discover AWS Accounts in an AWS Organization using AssumeRole, perform the following steps:

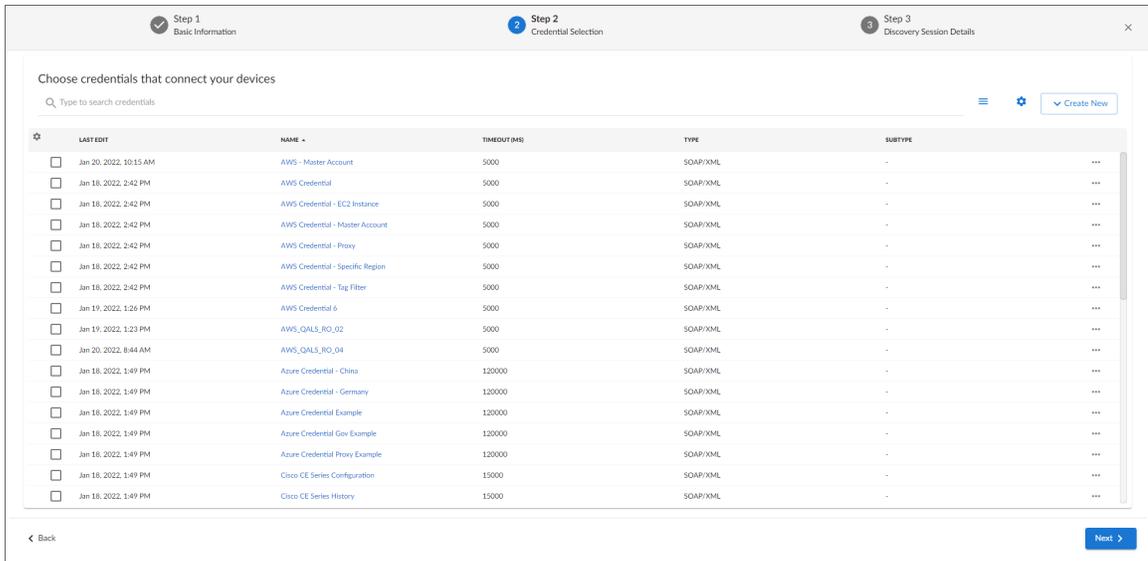
NOTE: If Ping is not supported between the Data Collector and AWS, you can skip this section and go to the [Manually Create the Organization and Align Dynamic Applications](#) section.

1. On the **Devices** page (🖨️) or the **Discovery Sessions** page (Devices > Discovery Sessions), click the **[Add Devices]** button. The **Select** page appears:

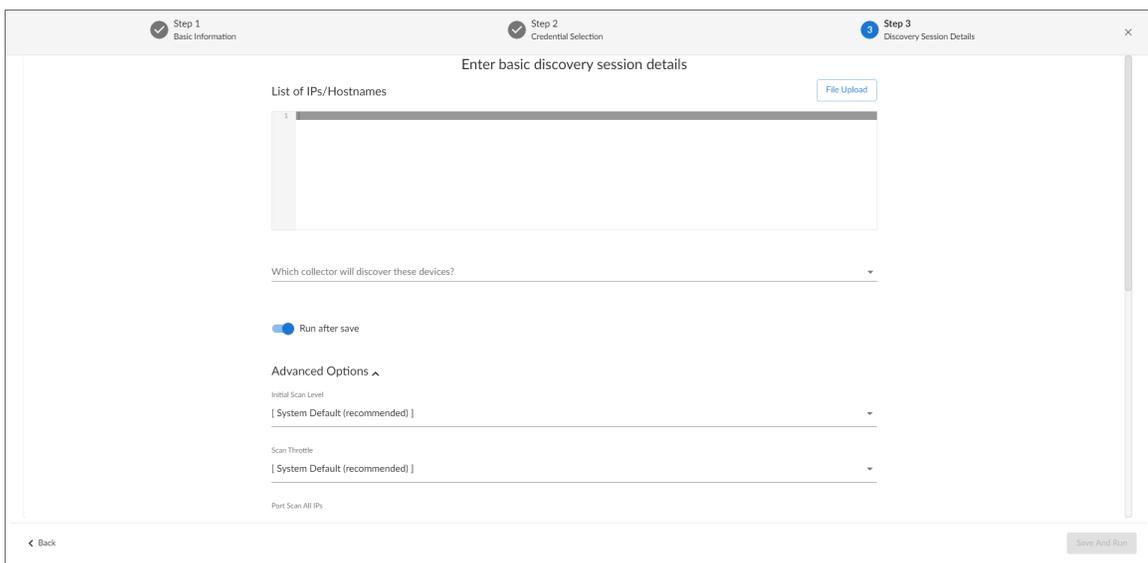


2. Click the **[Unguided Network Discovery]** button. Additional information about the requirements for discovery appears in the **General Information** pane to the right.
3. Click **[Select]**. The **Basic Information** page appears.

4. Supply values in the following fields:
 - **Name.** Type a unique name for this discovery session. This name is displayed in the list of discovery sessions on the **[Discovery Sessions]** tab.
 - **Description.** Optional. Type a short description of the discovery session. You can use the text in this description to search for the discovery session on the **[Discovery Sessions]** tab.
 - **Select the organization to add discovered devices to.** Select the name of the organization to which you want to add the discovered devices.
5. Click **[Next]**. The **Credential Selection** page of the **Add Devices** wizard appears:



6. On the **Credential Selection** page, locate and select the **credential** you created.
7. **[Next]**. The **Discovery Session Details** page of the **Add Devices** wizard appears:



8. Complete the following fields:
 - **List of IPs/Hostnames.** Type the URL of your AWS master billing account.
 - **Which collector will monitor these devices?** Required. Select an existing collector to monitor the discovered devices.
 - **Run after save.** Select this option to run this discovery session as soon as you save the session.

In the **Advanced options** section, click the down arrow icon () to complete the following fields:

 - **Discover Non-SNMP.** Enable this setting.
 - **Model Devices.** Enable this setting.
9. Click **[Save and Run]** if you enabled the Run after save setting, or **[Save and Close]** to save the discovery session. The **Discovery Sessions** page (Devices > Discovery Sessions) displays the new discovery session.
10. If you selected the **Run after save** option on this page, the discovery session runs, and the **Discovery Logs** page displays any relevant log messages. If the discovery session locates and adds any devices, the **Discovery Logs** page includes a link to the **Device Investigator** page for the discovered device.

NOTE: If you discontinue monitoring on any devices that are using the Assume Role authentication method, ScienceLogic recommends the best practice of first disabling the devices, deleting the devices from the DCM tree, and then cleaning up any AWS permissions in IAM. This will avoid any unnecessary alerts.

Create and Run the Discovery Session in the SL1 Classic User Interface

To discover AWS Accounts in an AWS Organization using AssumeRole, perform the following steps:

NOTE: If Ping is not supported between the Data Collector and AWS, you can skip this section and go to the [Manually Create the Organization and Align Dynamic Applications](#) section.

1. Go to the **Discovery Control Panel** page (System > Manage > Classic Discovery).

2. Click the **[Create]** button. The **Discovery Session Editor** page appears:

The screenshot shows the 'Discovery Session Editor | Create New' interface. It features several sections for configuring a discovery session:

- Identification Information:** Name: AWS Discovery for AssumeRole; Description: (empty).
- IP and Credentials:**
 - IP Address/Hostname Discovery List:** organizations.us-east-1.amazonaws.com
 - SNMP Credentials:** A list of credentials with 'AWS Proxy Master Account' selected.
 - Other Credentials:** Lync 2010 Credentials - Example, SQL PowerShell - Example, Windows PowerShell - Example.
- Detection and Scanning:**
 - Initial Scan Level:** System Default (recommended)
 - Scan Throttle:** System Default (recommended)
 - Port Scan All IPs:** System Default (recommended)
 - Port Scan Timeout:** System Default (recommended)
 - Detection Method & Port:** A list of protocols with 'UDP: 161 SNMP' selected.
 - Interface Inventory Timeout (ms):** 600000
 - Maximum Allowed Interfaces:** 10000
 - Bypass Interface Inventory:**
- Basic Settings:**
 - Discover Non-SNMP:**
 - Model Devices:**
 - DHCP:**
 - Device Model Cache TTL (h):** 2
 - Collection Server PID:** RS-ISO-DCU-35
 - Organization:** [System]
 - Add Devices to Device Group(s):** None, LayerX Appliances, Servers
 - Apply Device Template:** [Choose a Template]

Buttons: 'New', 'Reset', 'Save', 'Log All'.

3. Supply values in the following fields:
 - **IP Address Discovery List.** Type the URL of your AWS master billing account.
 - **Other Credentials.** Select the credential you created.
 - **Discover Non-SNMP.** Select this checkbox.
 - **Model Devices.** Select this checkbox.
4. Optionally, supply values in the other fields in this page. For a description of the fields in this page, see the **Discovery & Credentials** manual.
5. Click the **[Save]** button.
6. The **Discovery Control Panel** page will refresh. Click the lightning bolt icon (⚡) for the discovery session you just created.
7. In the pop-up window that appears, click the **[OK]** button. The page displays the progress of the discovery session.

NOTE: If you discontinue monitoring on any devices that are using the Assume Role authentication method, ScienceLogic recommends the best practice of first disabling the devices, deleting the devices from the DCM tree, and then cleaning up any AWS permissions in IAM. This will avoid any unnecessary alerts.

Manually Creating the Organization and Aligning Dynamic Applications

NOTE: The following steps are needed only if ping is **not** supported between the Data Collector and AWS.

To create a virtual device to create the organization:

1. Go to the **Device Manager** page (Devices > Device Manager or Registry > Devices > Device Manager in the SL1 classic user interface).
2. Click the **[Actions]** button, then select *Create Virtual Device*. The **Virtual Device** modal page appears:
3. Enter values in the following fields:
 - **Device Name**. Enter a name for the device. For example, you could enter "Amazon Organization" in this field.
 - **Organization**. Select the organization for this device. The organization the device is associated with limits the users that will be able to view and edit the device.
 - **Device Class**. Select *AWS | Organization*.
 - **Collector**. Select the collector group that will monitor the device.
4. Click the **[Add]** button to create the virtual device.

Next, you must manually align the "AWS: Account Creation" Dynamic Application with the AWS virtual device. After you do so, the other Dynamic Applications in the *Amazon Web Services PowerPack* will automatically align to discover and monitor all of the components in your AWS account.

To align the "AWS: Account Creation" Dynamic Application to your virtual device:

1. Go to the **Devices** page.
2. Locate your virtual device and click its name to open the **Device Investigator**.
3. In the **Device Investigator** page, click the **[Collections]** tab. The **Dynamic Application Collections** page appears.
4. Click the **[Edit]** button and then click the **[Align Dynamic App]** button.
5. In the **Align Dynamic Application** page, click *Choose Dynamic Application*.
6. In the **Choose Dynamic Application** page, select *AWS: Account Creation*.
7. In the **Align Dynamic Application** page, click *Choose Credential*.

8. In the **Choose Credential** page, select the *credential you created* and then click the **[Select]** button.
9. Click the **[Align Dynamic App]** button to align the Dynamic Application.

Automated Discovery when the Data Collector Runs as an EC2 Instance

This method of discovery is recommended for monitoring your AWS accounts within an organization when your Data Collectors are EC2 instances. In this case, a standard SL1 discovery process is created, and this mechanism will first discover your organization and then create all the accounts within the organization.

This method of discovery has the following benefits:

- No AWS credentials are needed in SL1

NOTE: All examples shown are for commercial AWS accounts. When AWS Gov is being monitored, the JSON data that refers to ARN will need to be modified from "aws" to "aws-us-gov". For example: Resource": "arn:aws:iam::<account number>:role/Sciencelogic-Monitor would need to be Resource": "arn:aws:iam-us-gov::<account number>:role/Sciencelogic-Monitor

To use this method of discovery, perform the following steps:

1. [Create an AWS role in the master billing account](#)
2. [Create an AWS role in account that the collector is in](#)
3. [Create an AWS role in each account that is to be monitored](#)
4. [Create an SL1 credential](#)
5. [Create and run the discovery session](#)

Create a Role in the Master Billing Account

The role you will create in the master billing account is assumed from the account that the EC2 instance is in. This role will enable SL1 to temporarily log in to the master billing account and discover other accounts.

Before creating the role, you must first create a policy that defines the permissions needed by SL1. To do this, copy the policy from below into an editor:

```
{ "Version": "2012-10-17",
  "Statement":
    [{"Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "organizations:ListAccounts",
        "organizations:DescribeOrganization",
        "organizations:DescribeAccount"
      ]
    }
  ],
```

```
        "Resource": "*"
    }
}
```

Next, perform the following steps:

1. Log in to the Master Billing Account via the AWS console and select **IAM > Policies > Create Policy**.
2. Select the **JSON** tab and paste the JSON text you copied above into the AWS console.
3. Click **Next: Tags** and then click **Next: Review**.
4. Type a name for the policy (for example, "SL1MasterBillingPermissions") in the **Name** field and then click **Create Policy**.

To create the role:

1. Go to **IAM > Roles > Create Role**.
2. Under **Select type of trusted entity**, select **Another AWS account**.
3. Type the account number of the account that contains the EC2 instance running on the collector in the **Account ID** field, and then click **Next: Permissions**.
4. Select the checkbox for the policy you created above.
5. Click **Next: Tags** and then click **Next: Review**.
6. Type the role name from the example above (SL1MasterAccountRole) in the **Role name** field, then click **Create role**.

The trust policy is set up by the console automatically as follows:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::581618222958:root"
      },
      "Action": "sts:AssumeRole",
      "Condition": {}
    }
  ]
}
```

7. In the console, edit the trust relationship and replace `:root` with `:role/ec2-collector`.

NOTE: "ec2-collector" is the name of the role that will be created in the account that the EC2 collector is in. This policy allows only the "ec2-collector" role to assume this role in the master billing account. If you use another name for the role, then this trust relationship must use that name instead of "ec2-collector".

Create an AWS Role in the Account your Data Collector is In

The role you create in the account your Data Collector is in will be assigned to the EC2 instances that house those Data Collectors. This role enables the SL1 Data Collector to assume a role in the master billing account, which is then used to discover the organization and retrieve the accounts associated with that organization. Once the accounts have been discovered, this role allows SL1 to assume the monitor role in each of the accounts.

First you will need to create a policy in the accounts that the Data Collectors are in. To create this policy, first cut and paste the following JSON text into an editor:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": "sts:AssumeRole",
      "Resource": [
        "arn:aws:iam::<master billing account ID>:role/SL1MasterAccountRole",
        "arn:aws:iam::<monitored account 1>:role/ScienceLogic-Monitor",
        "arn:aws:iam::<monitored account 2>:role/ScienceLogic-Monitor",
        "arn:aws:iam::<monitored account 3>:role/ScienceLogic-Monitor"
      ]
    }
  ]
}
```

Replace the "**master billing account**" with your master billing account number.

For each account to be monitored, ensure that there is a line under Resource that matches the account ID. The example above shows three accounts to be monitored.

NOTE: If the master billing account is to be monitored, it will also need a line in the Resource list.

If you did not use the example "SL1MasterAccountRole" name, replace it with the name of your role.

Next, perform the following steps:

1. Log in to the AWS console and select **IAM > Policies > Create Policy**.
2. Select the **JSON** tab and copy the JSON text you edited above into the AWS console.
3. Click **Next: Tags** and then click **Next: Review**.
4. Type a name for the policy (for example, "EC2CollectorPolicy") in the **Name** field and then click **Create Policy**.

To create the role:

NOTE: If you already have a role assigned to the Data Collector that houses the EC2 instance, then you can add the policy you just created to that existing role. Otherwise, follow the steps below to create the role.

1. Go to **IAM > Roles > Create Role**.
2. Under **Select type of trusted entity**, select **AWS service**.
3. Under **Choose a use case**, select **EC2**.
4. Click **Next: Permissions** and select the policy you created above.
5. Click **Next: Tags** and then click **Next: Review**.
6. Type the name from our example (ec2-collector) in the **Role name** field, then click **Create role**.

Next, you need to assign this instance profile to the EC2 instances that are Data Collectors. To do this:

1. Go to the AWS console and click **EC2 > Instances**.
2. Select the checkbox for each instance that is a Data Collector.
3. Click **Actions > Security > Modify IAM Role**.
4. In the drop-down field, select the role that you just created and then click **[Save]**.

Create a Role in Each Account

In every account that is to be monitored, a role with the **same name** needs to be created. The default name is ScienceLogic-Monitor. The following steps must be performed for each account that is to be monitored:

1. In the AWS console for the account and go to **IAM > Roles > Create Role**.
2. Under **Select type of trusted entity**, select **Another AWS account**.
3. Type the account number that houses the EC2 collectors in the **Account ID** field, and then click **Next: Permissions**.
4. Select the checkbox for the policy you created in the [Minimum Permissions Needed to Monitor Your AWS Accounts](#) section (called "SL1MinimumPermissions").
5. Click **Next: Tags** and then click **Next: Review**.
6. Type ScienceLogic-Monitor in the **Role name** field, then click **Create role**.
7. Click on the role that was just created and select the **Trust Relationships** tab.
8. Click the **[Edit trust relationship]** button.
9. In the **Policy Document** editor, change the Principle from "AWS": "arn:aws:iam::<ec2 collector account>:root" to "AWS": "arn:aws:iam::<collector account>:role/ec2-collector" (where ec2-collector is the name of the role created on the account housing the EC2 collector). Then click the **[Update Trust Policy]** button.
10. Repeat these steps for each account that is to be monitored.

Configuring the Credential to Discover AWS on an EC2 Collector

NOTE: If you are using an SL1 system prior to version 11.1.0, the new user interface does not include the **Duplicate** option for sample credential(s). ScienceLogic recommends that you use [the classic user interface and the Save As button](#) to create new credentials from sample credentials. This will prevent you from overwriting the sample credential(s).

To define an AWS credential to discover AWS on an EC2 collector:

1. Go to the **Credentials** page (Manage > Credentials).
2. Locate the **AWS Credential - EC2 Instance** sample credential, click its **[Actions]** icon (⋮) and select **Duplicate**. A copy of the credential, called **AWS Credential - Master Account copy** appears.
3. Click the **[Actions]** icon (⋮) for the **AWS Credential - EC2 Instance copy** credential and select **Edit**. The **Edit Credential** modal page appears:

The screenshot shows the 'Edit Credential' modal page. The form is titled 'Edit Credential' and contains several sections. The 'Name' field is 'AWS Credential - EC2 Instance copy'. There is a toggle for 'All Organizations' (currently on) and a dropdown for 'Select the organizations the credential belongs to'. The 'Timeout (ms)' is set to 5000. Below are dropdowns for 'Content Encoding' (text/xml), 'Method' (POST), and 'HTTP Version' (http/1.1). The 'URL' field contains 'https://organizations.us-east-1.amazonaws.com'. There are fields for 'HTTP Auth User' (IAM) and 'HTTP Auth Password'. There are also fields for 'Proxy Hostname/IP', 'Proxy User', 'Proxy Password', and 'Proxy Port'. At the bottom, there are four 'Embed Value' fields labeled [%1] through [%4]. On the right side, there is a 'Credential Tester' section with a 'Select Credential Test' dropdown, a 'Select Collector' dropdown (CUG | s11a01: 10.128.68.26), and an 'IP or Hostname to test' field with a 'Test Credential' button. A 'Save & Close' button is at the bottom right.

4. Enter values in the following fields:
 - **Name.** Type a new name for your AWS credential.
 - **All Organizations.** Toggle on (blue) to align the credential to all organizations, or toggle off (gray) and then select one or more specific organizations from the **What organization manages this service?** drop-down field to align the credential with those specific organizations.
 - **URL.** Type `https://organizations.us-east-1.amazonaws.com` in the field. If your administrator has configured a different region, you can change it or use the default region. **To discover Gov accounts** using AssumeRole, type `https://organizations.us-gov-west-1.amazonaws.com`.

- **HTTP Auth User.** Leave the default value "IAM" in the field.
- **HTTP Auth Password.** Leave the default value.
- **Embed Value [%2]:**
 - If you are using the AWS Config service and want to discover only regions that have that service enabled, type "[AUTO]" in this field. After discovery, only regions that have AWS Config enabled will be displayed in the dynamic component map tree. Global resources will also be discovered.
 - If you are using not using the AWS Config service, type "[FILTER]" in this field so it will discover only regions that are reporting CloudWatch metrics. This will reduce the number of regions being monitored and the load on the Data Collector.
- Under **HTTP Headers**, edit the following options:
 - **OrganizationArn.** Defines the ARN for the AssumeRole. This is the ARN of the role created in the master billing account. In the [example above](#) it was called "SL1MasterAccountRole". For example, `OrganizationArn:arn:aws:iam::<Master Billing Account>:role/SL1MasterAccountRole`
 - **AssumeRole.** Type the AWS Role you created in each account. The default name is "ScienceLogic-Monitor".
 - **AssumeRoleSession.** Optional. The default value is "AssumeRoleSession:SL1".
 - **OrganizationCreation:NAME:ID.** Autocreates an SL1 organization for accounts using AssumeRole. You can enter one of the following options:
 - **OrganizationCreation:NAME.** The name of the organization will contain the name of the user.
 - **OrganizationCreation:ID.** The name of the organization will contain the ID of the user.
 - **OrganizationCreation:ID:NAME.** The name of the organization will contain both the ID and name of the user, in that order.
 - **OrganizationCreation:NAME:ID.** The name of the organization will contain both the name and ID of the user, in that order.

NOTE: The existing organization will be changed by this setting only if it is the default (System) organization. If this header is not included, then **all** the discovered accounts will be placed into the organization selected in the discovery session.

5. Click the **[Save & Close]** button.

Configuring the Credential to Discover AWS on an EC2 Collector in the SL1 Classic User Interface

To define an AWS credential to discover AWS on an EC2 collector:

1. Go to the **Credential Management** page (System > Manage > Credentials).

- Locate the **AWS Credential - EC2 Instance** sample credential that you need and click its wrench icon (). The **Credential Editor** modal page appears:

- Enter values in the following fields:

Basic Settings

- **Profile Name.** Type a new name for your AWS credential.
- **URL.** Type `https://organizations.us-east-1.amazonaws.com` in the field. If your administrator has configured a different region, you can change it or use the default region. **To discover Gov accounts** using AssumeRole, type `https://organizations.us-gov-west-1.amazonaws.com`.
- **HTTP Auth User.** Leave the default value "IAM" in the field.

SOAP Options

- **Embed Value [%2]:**
 - If you are using the AWS Config service and want to discover only regions that have that service enabled, type "[AUTO]" in this field. After discovery, only regions that have AWS Config enabled will be displayed in the dynamic component map tree. Global resources will also be discovered.
 - If you are not using the AWS Config service, type "[FILTER]" in this field so it will discover only regions that are reporting CloudWatch metrics. This will reduce the number of regions being monitored and the load on the Data Collector.

HTTP Headers

- Click **+ Add a header** to add a header field. You can enter the following options:
 - *OrganizationArn*. Defines the ARN for the AssumeRole. This is the ARN of the role created in the master billing account. In the [example above](#) it was called "SL1MasterAccountRole". For example, `OrganizationArn:arn:aws:iam::<Master Billing Account>:role/SL1MasterAccountRole`
 - *AssumeRole*. Type the AWS Role you created in each account. The default name is "ScienceLogic-Monitor".
 - *AssumeRoleSession*. Optional. The default value is "AssumeRoleSession:SL1".
 - *OrganizationCreation:NAME:ID*. Autocreates an SL1 organization for accounts using AssumeRole. You can enter one of the following options:
 - **OrganizationCreation:NAME**. The name of the organization will contain the name of the user.
 - **OrganizationCreation:ID**. The name of the organization will contain the ID of the user.
 - **OrganizationCreation:ID:NAME**. The name of the organization will contain both the ID and name of the user, in that order.
 - **OrganizationCreation:NAME:ID**. The name of the organization will contain both the name and ID of the user, in that order.

NOTE: The existing organization will be changed by this setting only if it is the default (System) organization.

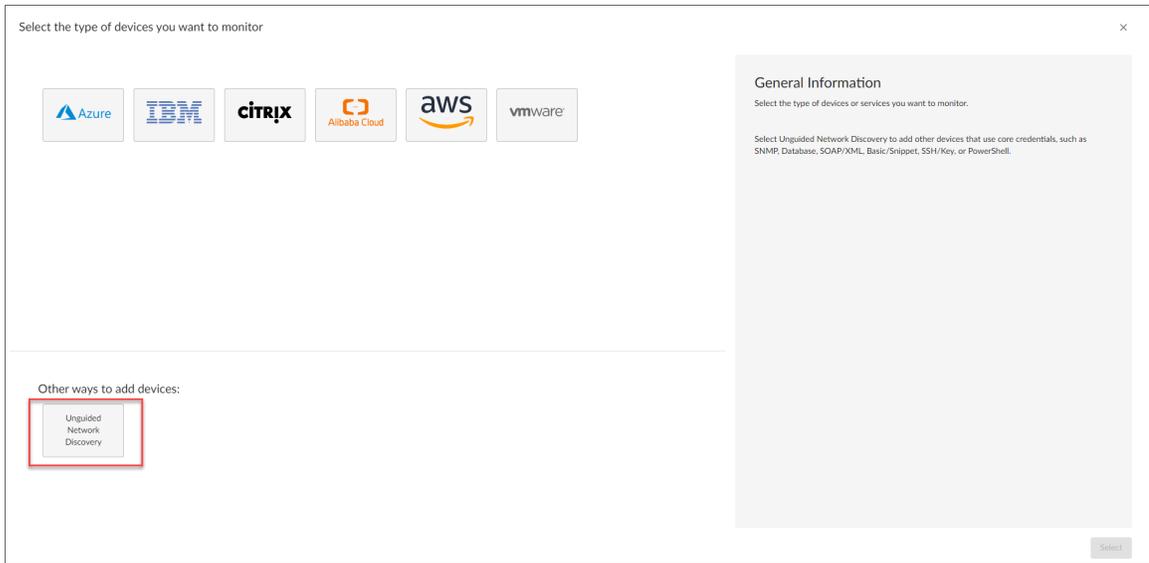
4. Click the **[Save As]** button, then click **[OK]**.

Create and Run the Discovery Session

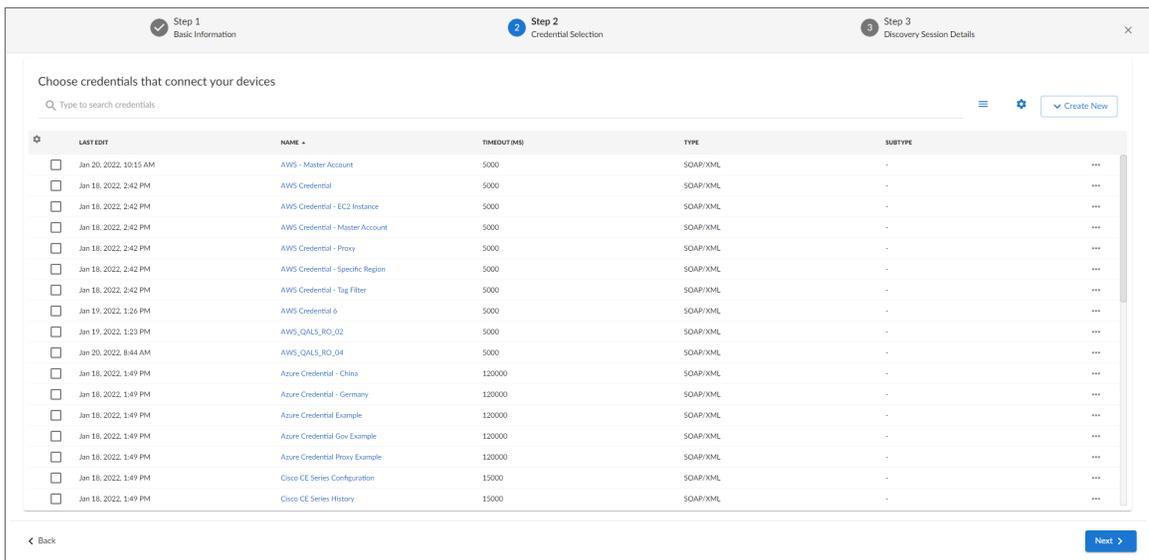
To discover AWS Accounts in an AWS Organization using AssumeRole, perform the following steps:

NOTE: If you are upgrading the PowerPack and had previously discovered accounts within an organization separately and now want to use a different discovery method, you must first disable the "AWS: Account Discovery" Dynamic Application in each account that is being upgraded.

1. On the **Devices** page () or the **Discovery Sessions** page (Devices > Discovery Sessions), click the **[Add Devices]** button. The **Select** page appears:



2. Click the **[Unguided Network Discovery]** button. Additional information about the requirements for discovery appears in the **General Information** pane to the right.
3. Click **[Select]**. The **Basic Information** page appears.
4. Supply values in the following fields:
 - **Name**. Type a unique name for this discovery session. This name is displayed in the list of discovery sessions on the **[Discovery Sessions]** tab.
 - **Description**. Optional. Type a short description of the discovery session. You can use the text in this description to search for the discovery session on the **[Discovery Sessions]** tab.
 - **Select the organization to add discovered devices to**. Select the name of the organization to which you want to add the discovered devices.
5. Click **[Next]**. The **Credential Selection** page of the **Add Devices** wizard appears:



6. On the **Credential Selection** page, locate and select the *credential* you created.
7. **[Next]**. The **Discovery Session Details** page of the **Add Devices** wizard appears:

8. Complete the following fields:
 - **List of IPs/Hostnames.** Type the URL of your AWS master billing account.
 - **Which collector will monitor these devices?** Required. Select an existing Data Collector to monitor the discovered devices.
 - **Run after save.** Select this option to run this discovery session as soon as you save the session.
 - In the **Advanced options** section, click the down arrow icon () to complete the following fields:
 - **Discover Non-SNMP.** Enable this setting.
 - **Model Devices.** Enable this setting.
9. Click **[Save and Run]** if you enabled the Run after save setting, or **[Save and Close]** to save the discovery session. The **Discovery Sessions** page (Devices > Discovery Sessions) displays the new discovery session.
10. If you selected the **Run after save** option on this page, the discovery session runs, and the **Discovery Logs** page displays any relevant log messages. If the discovery session locates and adds any devices, the **Discovery Logs** page includes a link to the **Device Investigator** page for the discovered device.

NOTE: If you discontinue monitoring on any devices that are using the Assume Role authentication method, ScienceLogic recommends the best practice of first disabling the devices, deleting the devices from the DCM tree, and then cleaning up any AWS permissions in IAM. This will avoid any unnecessary alerts.

Create and Run the Discovery Session in the SL1 Classic User Interface

To discover AWS Accounts in an AWS Organization using AssumeRole, perform the following steps:

NOTE: If you are upgrading the PowerPack and had previously discovered accounts within an organization separately and now want to use a different discovery method, you must first disable the "AWS: Account Discovery" Dynamic Application in each account that is being upgraded.

1. Go to the **Discovery Control Panel** page (System > Manage > Classic Discovery).
2. Click the **[Create]** button. The **Discovery Session Editor** page appears:

The screenshot shows the 'Discovery Session Editor | Create New' interface. It features a top navigation bar with 'New' and 'Reset' buttons. The main content area is organized into four primary sections:

- Identification Information:** Includes a 'Name' field (populated with 'AWS Discovery for AssumeRole') and a 'Description' field.
- IP and Credentials:** Contains an 'IP Address/Hostname Discovery List' with a text input field (populated with 'organizations.us-east-1.amazonaws.com') and a 'Browse for file...' button. Below this are sections for 'SNMP Credentials' (listing various protocols like Cisco, Dell EMC, and AWS) and 'Other Credentials' (listing various user types like Lync, SQL, and Windows).
- Detection and Scanning:** Features several dropdown menus for 'Initial Scan Level', 'Scan Throttle', 'Port Scan All IPs', and 'Port Scan Timeout', all currently set to 'System Default (recommended)'. It also includes a 'Detection Method & Port' list, an 'Interface Inventory Timeout (ms)' field (set to 600000), and a 'Maximum Allowed Interfaces' field (set to 10000).
- Basic Settings:** Includes checkboxes for 'Discover Non-SNMP', 'Model Devices', and 'DHCP'. It also has a 'Device Model Cache TTL (h)' field (set to 2), a 'Collection Server PID' dropdown (set to 'RS-ISO-DCU-35'), an 'Organization' dropdown (set to '[System]'), and an 'Add Devices to Device Group(s)' list.

At the bottom of the form, there is a 'Save' button and a 'Log All' checkbox.

3. Supply values in the following fields:
 - **IP Address Discovery List.** Type the URL of your AWS master billing account.
 - **Other Credentials.** Select the credential you created.
 - **Discover Non-SNMP.** Select this checkbox.
 - **Model Devices.** Select this checkbox.

4. Optionally, supply values in the other fields in this page. For a description of the fields in this page, see the **Discovery & Credentials** manual.
5. Click the **[Save]** button.
6. The **Discovery Control Panel** page will refresh. Click the lightning bolt icon () for the discovery session you just created.
7. In the pop-up window that appears, click the **[OK]** button. The page displays the progress of the discovery session.

NOTE: If you discontinue monitoring on any devices that are using the Assume Role authentication method, ScienceLogic recommends the best practice of first disabling the devices, deleting the devices from the DCM tree, and then cleaning up any AWS permissions in IAM. This will avoid any unnecessary alerts.

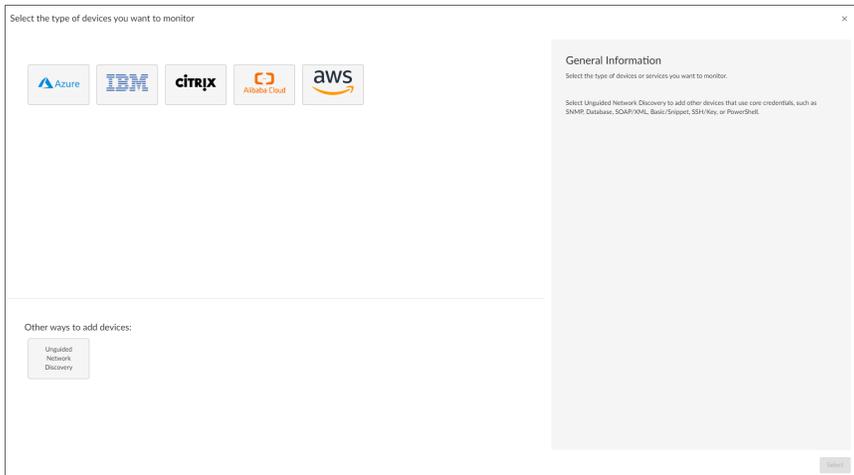
AWS Guided Discovery

You can use the Universal Discovery Framework process in SL1 that guides you through a variety of existing discovery types in addition to traditional SNMP discovery. This process, which is also called "guided discovery", lets you pick a discovery type based on the type of devices you want to monitor. The Universal Discovery workflow includes a button for Amazon Web Services.

NOTE: If you want to discover one of the third-party products that are available as an option when using the Universal Discovery workflow, you must have the corresponding PowerPack installed on your SL1 system to ensure that the appropriate Dynamic Applications, Device Classes, and other elements can be utilized for discovery. For example, if you want to discover an Amazon Web Services account, you must have the *Amazon Web Services* PowerPack installed.

To run a guided or Universal Discovery:

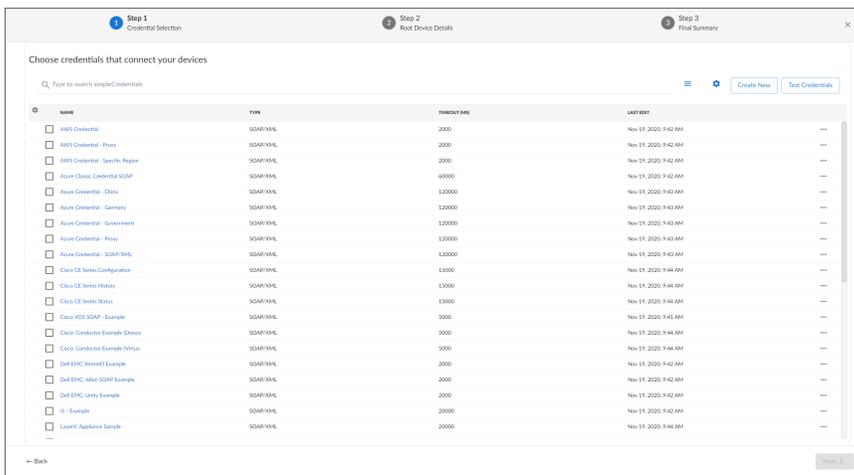
1. On the **Devices** page () or the **Discovery Sessions** page (Devices > Discovery Sessions), click the **[Add Devices]** button. The **Select** page appears.



2. Select the **Amazon Web Services** button. Additional information about the requirements for device discovery appears in the **General Information** pane to the right. If you are on SL1 11.2.0 or later, you will be prompted to select a type of AWS guided discovery from the following:

- **AWS EC2**
- **AWS IAM**
- **AWS Assume Role**

3. Click **[Select]**. The **Credential Selection** page appears:



NOTE: During the guided discovery process, you cannot click **[Next]** until the required fields are filled on the page, nor can you skip to future steps. However, you can revisit previous steps that you have already completed.

- On the **Credential Selection** page of the guided discovery process, will select the AWS credential for the guided workflow that you chose. If you are not yet on SL1 version 11.2.0, select the credential that you configured for basic guided discovery. If you have not yet configured a credential, go to the **Credentials** page (Manage > Credentials) and configure the type of credential you will need:
 - [AWS Assume Role Credential](#)
 - [AWS EC2 Credential](#)
 - [AWS IAM Credential](#)

Defining an AWS Assume Role Credential

SL1 includes an AWS Assume Role credential type that you can use to connect with the AWS service during guided discovery using the Assume Role discovery method. The Assume Role discovery method provides an automated mechanism to discover all your AWS accounts within an organization using a single IAM key. This credential type uses field names and terminology that are specific to the AWS service.

NOTE: For more information about monitoring AWS using Assume Role, see the section on [Automated Discovery Using Assume Role with a Single IAM Key from the AWS Master Account](#).

To define an AWS Assume Role credential:

- Go to the **Credentials** page (Manage > Credentials).
- Click the **[Create New]** button and then select *Create AWS Assume Role Credential*. The **Create Credential** modal page appears:

- Supply values in the following fields:
 - Name.** Type a unique name for the credential. Can be any combination of alphanumeric characters, up to 64 characters.

- **All Organizations.** Toggle on (blue) to align the credential to all organizations, or toggle off (gray) and then select one or more specific organizations from the **Select the organizations the credential belongs to** drop-down field to align the credential with those specific organizations.
- **Timeout (ms).** Type the time, in milliseconds, after which SL1 will stop trying to communicate with the device from which you want to retrieve data.
- **AWS Access Key ID.** Type the Access Key ID for an account on the AWS device to be monitored.
- **AWS Secret Access Key.** Type the Secret Access Key for an account on the AWS device to be monitored.
- **Cloud Type.** Select the AWS cloud type that will be accessed with the credential. This field is required. Choices are:
 - *Standard.* Select this option if you want to connect to a standard AWS account.
 - *GovCloud.* Select this option if you want to connect to an AWS GovCloud account.
 - *Beijing.* Select this option if you want to connect to AWS regions in China.
- **Assume Role.** Type the AWS Role you created in each account. The default name is "ScienceLogic-Monitor".
- **Assume Role Session.** Optional. The default value is "SL1".
- **Organization Creation.** Auto-creates an SL1 organization for accounts using AssumeRole. You can type one of the following options:

NOTE: Credentials created for guided discovery workflows do not need "OrganizationCreation" typed before the Name and/or ID.

- *NAME.* The name of the organization will contain the name of the user.
- *ID.* The name of the organization will contain the ID of the user.
- *ID:NAME.* The name of the organization will contain both the ID and name of the user, in that order.
- *NAME:ID.* The name of the organization will contain both the name and ID of the user, in that order.
- **Configuration.** Select the method used to control what AWS devices are discovered and monitored. Choices are:
 - *Default.* The default AWS discovery method.
 - *AwsConfig.* Select this option if your accounts have the AWS Config service enabled.
 - *AwsCloudwatch.* Select this option to discover only the AWS regions that are reporting CloudWatch metrics.
- **Regions.** Type the AWS regions that you want to discover. For example, entering "ap-southeast-2, us-east-2" will discover two regions. If left blank, all regions will be discovered. The default value is "ALL".

- **Filter by Tags.** To discover AWS devices and filter them by tags, type the tag operation, tag key, and tag value, in the following format: <operation>#<tag name>#<tag value>. For example, if you want to filter by Tag Name, you would type the following:

```
Tags:equals#Name#Example
```

Valid operations include:

- equals
- notEquals
- contains
- notContains

You can chain together multiple filters separating them by a comma. For example:

```
Tags:equals#Name#Example,contains#Owner#Someone
```

- **Proxy Hostname/IP.** Type the host name or IP address of the proxy server.
- **Proxy Port.** Type the port number on the proxy server to which you will connect.
- **Proxy User.** Type the username to use to access the proxy server.
- **Proxy Password.** Type the password to use to access the proxy server.

NOTE: If you use a proxy server in front of the AWS devices you want to communicate with, enter values in the proxy fields. Otherwise, you can skip these fields.

4. Click **[Save & Close]**.

NOTE: If you would like to test your credential using the Credential Tester panel, click **[Save & Test]**. For detailed instructions on using the Credential Tester panel, see the [Testing the AWS Credential](#) section.

Defining an AWS EC2 Credential

SL1 includes an AWS EC2 credential type that you can use to connect with the AWS service during guided discovery when your Data Collectors are EC2 instances. This credential type uses field names and terminology that are specific to the AWS service.

NOTE: For more information about monitoring AWS accounts within an organization when your Data Collectors are EC2 instances, see the section on [Automated Discovery when the Data Collector Runs as an EC2 Instance](#).

To define an EC2 credential:

1. Go to the **Credentials** page (Manage > Credentials).
2. Click the **[Create New]** button and then select *Create AWS EC2 Credential*. The **Create Credential** modal page appears:

The screenshot shows the 'Create Credential' modal window. It has a title bar with a close button. The main content area is split into two columns. The left column has a 'Name *' input field, an 'All Organizations' toggle (currently blue), a 'Timeout (ms)' input field with '1500', a 'Cloud Type' dropdown (set to 'Standard'), an 'Organization Arn *' input field with 'arn:aws:iam::987654321', an 'Assume Role *' input field with 'Sciencelogic-Monitor', an 'Assume Role Session *' input field with 'SL1', and an 'Organization Creation (optional)' input field with 'NAME:ID'. The right column is titled 'Credential Tester' and has a 'Select Credential Test' dropdown, a 'Select Collector' dropdown with 'CUG | rcarnigan-aio-102821: 10.2.8.21', and an 'IP or Hostname to test *' input field. A 'Test Credential' button is positioned below the IP field. A 'Close' button is at the bottom right of the modal.

3. Supply values in the following fields:

- **Name**. Type a unique name for the credential. Can be any combination of alphanumeric characters, up to 64 characters.
- **All Organizations**. Toggle on (blue) to align the credential to all organizations, or toggle off (gray) and then select one or more specific organizations from the **Select the organizations the credential belongs to** drop-down field to align the credential with those specific organizations.
- **Timeout (ms)**. Type the time, in milliseconds, after which SL1 will stop trying to communicate with the device from which you want to retrieve data.
- **Cloud Type**. Select the AWS cloud type that will be accessed with the credential. This field is required. Choices are:
 - *Standard*. Select this option if you want to connect to a standard AWS account.
 - *GovCloud*. Select this option if you want to connect to an AWS GovCloud account.

NOTE: To use a Government account, the EC2 or All-in-One Data Collector should be created on one GovCloud child account.

- *Beijing*. Select this option if you want to connect to AWS regions in China.
- **Organization Arn**. Type the Amazon Resource Name (ARN) for the Assume Role. This is the ARN of the role created in the master billing account.

- **Assume Role.** Type the AWS Role you created in each account. The default name is "ScienceLogic-Monitor".
- **Assume Role Session.** Optional. The default value is "SL1".
- **Organization Creation.** Auto-creates an SL1 organization for accounts using AssumeRole. You can type one of the following options:

NOTE: Credentials created for guided discovery workflows do not need "OrganizationCreation" typed before the Name and/or ID.

- *NAME.* The name of the organization will contain the name of the user.
 - *ID.* The name of the organization will contain the ID of the user.
 - *ID:NAME.* The name of the organization will contain both the ID and name of the user, in that order.
 - *NAME:ID.* The name of the organization will contain both the name and ID of the user, in that order.
- **Configuration.** Select the type of method used to control what AWS devices are discovered and monitored. Choices are:
 - *Default.* The default AWS discovery method.
 - *AwsConfig.* Select this option if your accounts have the AWS Config service enabled.
 - *AwsCloudwatch.* Select this option to discover only the AWS regions that are reporting CloudWatch metrics.
 - **Regions.** Type the AWS regions that you want to discover. For example, entering "ap-southeast-2, us-east-2" will discover two regions. If left blank, all regions will be discovered. The default value is "ALL".
 - **Filter by Tags.** To discover AWS devices and filter them by tags, type the tag operation, tag key, and tag value, in the following format: <operation>#<tag name>#<tag value>. For example, if you want to filter by Tag Name, you would type the following:

Tags:equals#Name#Example

Valid operations include:

- equals
- notEquals
- contains
- notContains

You can chain together multiple filters separating them by a comma. For example:

Tags:equals#Name#Example,contains#Owner#Someone

- **Proxy Hostname/IP.** Type the host name or IP address of the proxy server.
- **Proxy Port.** Type the port number on the proxy server to which you will connect.
- **Proxy User.** Type the username to use to access the proxy server.
- **Proxy Password.** Type the password to use to access the proxy server.

NOTE: If you use a proxy server in front of the AWS devices you want to communicate with, enter values in the proxy fields. Otherwise, you can skip these fields.

4. Click **[Save & Close]**.

NOTE: If you would like to test your credential using the Credential Tester panel, click **[Save & Test]**. For detailed instructions on using the Credential Tester panel, see the [Testing the AWS Credential](#) section.

Defining an AWS IAM Credential

You can use IAM policies in AWS to restrict which regions and services SL1 will monitor. To do this, you can create another IAM policy and apply that along with the SL1 monitoring policy to the applicable user or role(s).

SL1 includes an AWS IAM credential type that you can use to connect with the AWS service during guided discovery using the IAM discovery method. This credential type uses field names and terminology that are specific to the AWS service.

NOTE: For more information about monitoring AWS using IAM permissions, see the section on [Using IAM Permissions to Restrict SL1 Access to Specific Regions and Services](#).

To define an AWS IAM credential:

1. Go to the **Credentials** page (Manage > Credentials).
2. Click the **[Create New]** button and then select *Create AWS IAM Credential*. The **Create Credential** modal page appears:

3. Supply values in the following fields:

- **Name**. Type a unique name for the credential. Can be any combination of alphanumeric characters, up to 64 characters.
- **All Organizations**. Toggle on (blue) to align the credential to all organizations, or toggle off (gray) and then select one or more specific organizations from the **Select the organizations the credential belongs to** drop-down field to align the credential with those specific organizations.
- **Timeout (ms)**. Type the time, in milliseconds, after which SL1 will stop trying to communicate with the device from which you want to retrieve data.
- **AWS Access Key ID**. Type the Access Key ID for an account on the AWS device to be monitored.
- **AWS Secret Access Key**. Type the Secret Access Key for an account on the AWS device to be monitored.
- **Cloud Type**. Select the AWS cloud type that will be accessed with the credential. This field is required. Choices are:
 - *Standard*. Select this option if you want to connect to a standard AWS account.
 - *GovCloud*. Select this option if you want to connect to an AWS GovCloud account.
 - *Beijing*. Select this option if you want to connect to AWS regions in China.
- **Configuration**. Select the method used to control what AWS devices are discovered and monitored. Choices are:

- *Default*. The default AWS discovery method.
- *AwsConfig*. Select this option if your accounts have the AWS Config service enabled.
- *AwsCloudwatch*. Select this option to discover only the AWS regions that are reporting CloudWatch metrics.
- **Regions**. Type the AWS regions that you want to discover. For example, entering "ap-southeast-2, us-east-2" will discover two regions. If left blank, all regions will be discovered. The default value is "ALL".
- **Filter by Tags**. To discover AWS devices and filter them by tags, type the tag operation, tag key, and tag value, in the following format: <operation>#<tag name>#<tag value>. For example, if you want to filter by Tag Name, you would type the following:

```
Tags:equals#Name#Example
```

Valid operations include:

- equals
- notEquals
- contains
- notContains

You can chain together multiple filters separating them by a comma. For example:

```
Tags:equals#Name#Example,contains#Owner#Someone
```

- **Proxy Hostname/IP**. Type the host name or IP address of the proxy server.
- **Proxy Port**. Type the port number on the proxy server to which you will connect.
- **Proxy User**. Type the username to use to access the proxy server.
- **Proxy Password**. Type the password to use to access the proxy server.

NOTE: If you use a proxy server in front of the AWS devices you want to communicate with, enter values in the proxy fields. Otherwise, you can skip these fields.

4. Click **[Save & Close]**.

NOTE: If you would like to test your credential using the Credential Tester panel, click **[Save & Test]**. For detailed instructions on using the Credential Tester panel, see the [Testing the AWS Credential](#) section.

Completing the Discovery Session

5. Once you have finished creating or configuring your credential and have selected it in the **Credential Selection** page, click **[Next]**. The **Discovery Session Name** page appears.

6. Complete the following fields:

- **Discovery Session Name.** Type a name for the discovery session.
- **Select the organization to add discovered devices to.** Select the name of the organization to which you want to add the discovered device.
- **Collector Group Name.** Select an existing collector group to communicate with the discovered device. This field is required.

NOTE: When assigning devices to a collector group, SL1's multi-tenancy rules will validate that the collector group you select belongs to the organization you selected in the previous field. If you attempt to run a discovery session where the devices, collector group, and credentials do not all belong to the same organization, you will receive an error message and will not be able to save or execute the discovery session.

7. Click **[Next]**. SL1 creates the AWS root device with the appropriate Device Class assigned to it and aligns the relevant Dynamic Applications. The **Device Discovery Completed** page appears, which is the third and final step of the guided discovery session. In SL1 version 11.2.0 and later, as SL1 discovers your devices, system messages relating to the discovery appear on the page under the heading "Discovery Logs".

NOTE: If SL1 cannot determine the appropriate Device Class, it will assign the device to the Generic SNMP Device Class.

8. Click **[Close]**.

NOTE: The results of a guided discovery do not display on the **Discovery Sessions** page (Devices > Discovery Sessions). However, you can retrieve details of saved Guided Discovery Sessions with the `guidedDiscoverySessions` GraphQL query. Details for discovery sessions that create a virtual root device are not currently displayed in the user interface.

The AWS Credential Test and Viewing Component Devices

Overview

The following sections describe how to use the AWS credential test, understanding AWS Dynamic Applications, and how to view AWS component devices: Amazon Web Services PowerPack

<i>Testing the AWS Credential</i>	100
<i>Testing the AWS Credential in the SL1 Classic User Interface</i>	102
<i>Viewing AWS Component Devices</i>	103
<i>Relationships Between Component Devices</i>	105
<i>Vanishing Component Devices</i>	107

Testing the AWS Credential

NOTE: The Credential Test is for use with the [Manual Discovery](#) method only.

SL1 includes a Credential Test for Amazon Web Services. Credential Tests define a series of steps that SL1 can execute on demand to validate whether a credential works as expected.

The AWS Credential Test can be used to test a SOAP/XML credential for monitoring AWS using the Dynamic Applications in the *Amazon Web Services PowerPack*. The AWS Credential Test performs the following steps:

- **Test Reachability.** Performs an ICMP ping request to the URL for the EC2 service in the region specified in the credential. If a region is not specified in the credential, the us-east-1 region is used.

- **Test Port Availability.** Performs an NMAP request to TCP port 443 on the URL for the EC2 service in the region specified in the credential. If a region is not specified in the credential, the us-east-1 region is used.
- **Test Name Resolution.** Performs an nslookup request on the URL for the EC2 service in the region specified in the credential. If a region is not specified in the credential, the us-east-1 region is used.
- **Make connection to AWS account.** Attempts to connect to the AWS service using the account specified in the credential.
- **Scan AWS services.** Verifies that the account specified in the credential has access to the services.

NOTE: The AWS Credential Test does not support the testing of credentials that connect to AWS through a proxy server.

To test the AWS credential:

1. Go to the **Credentials** page (Manage > Credentials).
2. Locate the credential you wish to test, select the **Actions** button (⋮) next to it and click *Test*.
3. The **Credential Test Form** modal page appears. Fill out the following fields on this page:
 - **Credential.** This field is read-only and displays the name of the credential you selected.
 - **Select Credential Test.** Select **AWS Credential Test**.
 - **Collector.** Select the All-In-One Appliance or Data Collector that will run the test.
 - **IP or Hostname to Test.** Enter a valid IP address.
4. Click the **[Run Test]** button to run the credential test. The **Testing Credential** window appears:

STEP	DESCRIPTION	LOG MESSAGE	STATUS
Test Reachability	Check to see if the device is reachable using ICMP	The device is reachable using ICMP. The average response time is 2...	Passed
Test Port Availability	Check to see if the appropriate port is open	Port 443 is open	Passed
Test Name Resolution	Check to see if nslookup can resolve the IP and hostname	Name resolution failed: Reverse failed, Forward failed	Failed
Make rIRI Request	Check to see if a rIRI request succeeds	rIRI request failed: HTTP 400	Failed

The **Testing Credential** window displays a log entry for each step in the credential test. The steps performed are different for each credential test. The log entry for each step includes the following information:

- **Step.** The name of the step.
- **Description.** A description of the action performed during the step.
- **Log Message.** The result of the step for this execution of the credential test.

- **Status.** Whether the result of this step indicates the credential and/or the network environment is configured correctly (Passed) or incorrectly (Failed).
- **Step Tip.** Mouse over the question mark icon (?) to display the tip text. The tip text recommends what to do to change the credential and/or the network environment if the step has a status of "Failed".

Testing the AWS Credential in the SL1 Classic User Interface

NOTE: The Credential Test is for use with the *Manual Discovery* method only.

SL1 includes a Credential Test for Amazon Web Services. Credential Tests define a series of steps that SL1 can execute on demand to validate whether a credential works as expected.

The AWS Credential Test can be used to test a SOAP/XML credential for monitoring AWS using the Dynamic Applications in the *Amazon Web Services PowerPack*. The AWS Credential Test performs the following steps:

- **Test Reachability.** Performs an ICMP ping request to the URL for the EC2 service in the region specified in the credential. If a region is not specified in the credential, the us-east-1 region is used.
- **Test Port Availability.** Performs an NMAP request to TCP port 443 on the URL for the EC2 service in the region specified in the credential. If a region is not specified in the credential, the us-east-1 region is used.
- **Test Name Resolution.** Performs an nslookup request on the URL for the EC2 service in the region specified in the credential. If a region is not specified in the credential, the us-east-1 region is used.
- **Make connection to AWS account.** Attempts to connect to the AWS service using the account specified in the credential.
- **Scan AWS services.** Verifies that the account specified in the credential has access to the services.

NOTE: The AWS Credential Test does not support the testing of credentials that connect to AWS through a proxy server.

To test the AWS credential:

1. Go to the **Credential Test Management** page (System > Customize > Credential Tests).
2. Locate the **AWS Credential Test** and click its lightning bolt icon (⚡). The **Credential Tester** modal page appears:

3. Supply values in the following fields:

- **Test Type.** This field is pre-populated with the credential test you selected.
- **Credential.** Select the credential to test. This drop-down list includes only credentials that you have access to that can be tested using the selected credential test.
- **Hostname/IP.** Leave this field blank.
- **Collector.** Select the All-In-One Appliance or Data Collector that will run the test.

4. Click the **[Run Test]** button to run the credential test. The **Test Credential** window appears:

Step	Description	Log Message	Status
1 Test Reachability	Check to see if the EC2 service is reachable using ICMP	The EC2 service is reachable using ICMP. The average response time is 3.400ms	Passed
2 Test Port Availability	Check to see if the EC2 HTTPS port is open	Port 443 is open	Passed
3 Test Name Resolution	Check to see if nslookup can resolve the EC2 Service	Name resolution succeeded: Forward returned 1 result	Passed
4 Make connection to AWS account	Check to see if an AWS account can be connected to and queried	AWS connection succeeded	Passed
5 Scan AWS Services	Verify services are available to specified account.	AWS service scan succeeded	Passed

The **Test Credential** window displays a log entry for each step in the credential test. The steps performed are different for each credential test. The log entry for each step includes the following information:

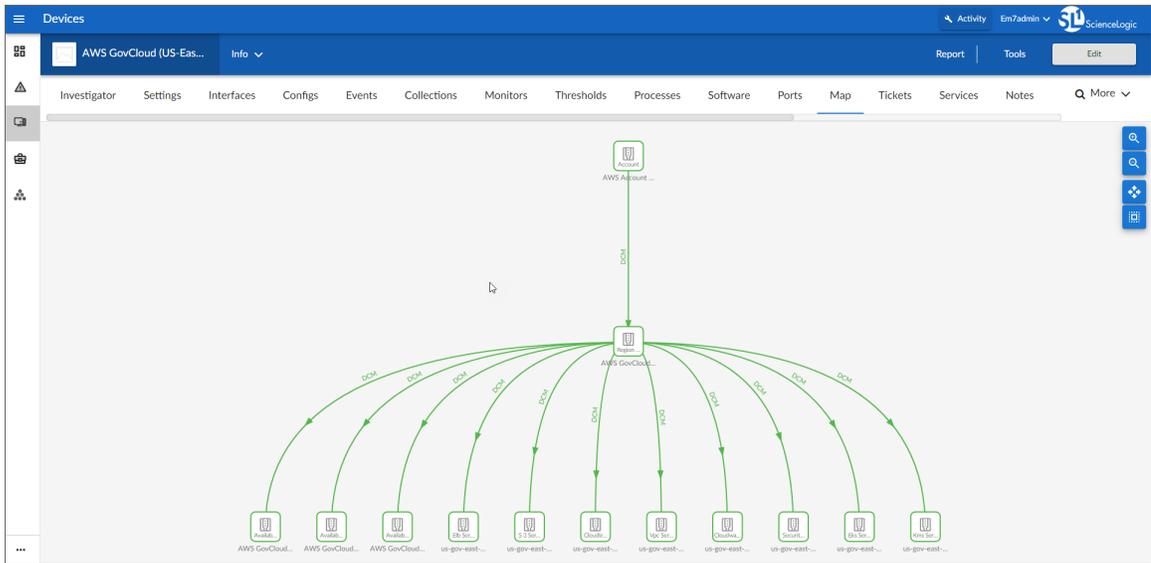
- **Step.** The name of the step.
- **Description.** A description of the action performed during the step.
- **Log Message.** The result of the step for this credential test.
- **Status.** Whether the result of this step indicates the credential or the network environment is configured correctly (Passed) or incorrectly (Failed).
- **Step Tip.** Mouse over the question mark icon (?) to display the tip text. The tip text recommends what to do to change the credential or the network environment if the step has a status of "Failed".

Viewing AWS Component Devices

When SL1 performs collection for the AWS virtual device, SL1 will create component devices that represent each element in your AWS infrastructure and align other Dynamic Applications to those component devices. Some of the Dynamic Applications aligned to the component devices will also be used to create additional component devices. All component devices appear in the **Devices** page.

In addition to the **Devices** page, you can view the AWS service and all associated component devices in the following places in the user interface:

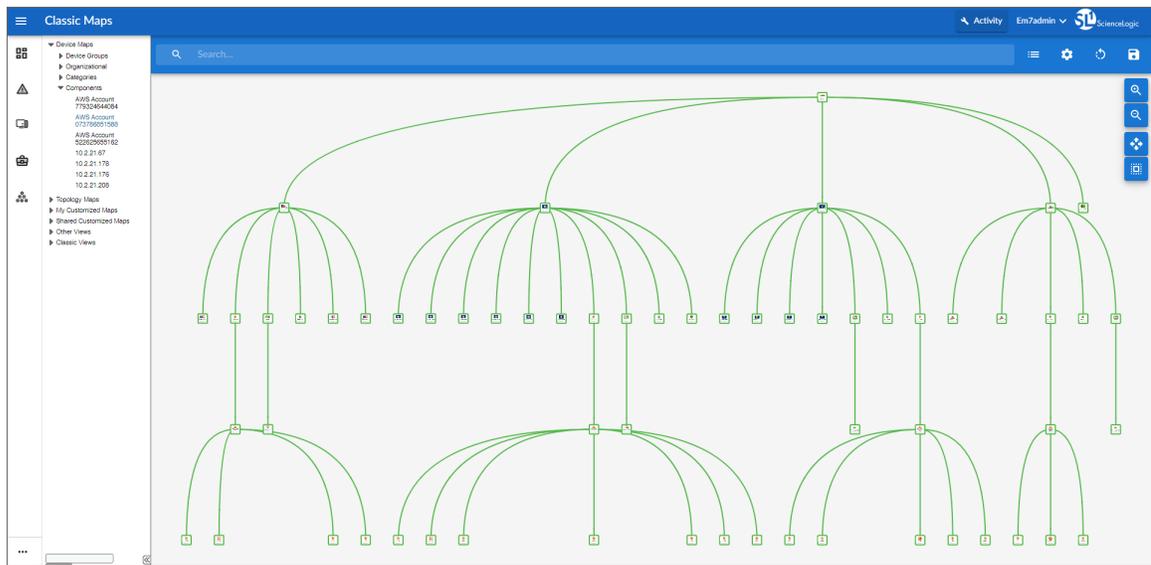
- The **Device Investigator** Map page (click **Map** in the **Device Investigator** page) displays a map of a particular device and all of the devices with which it has parent-child relationships. Double-clicking any of the listed devices reloads the page to make the selected device the primary device



- The **Device Components** page (Devices > Device Components) displays a list of all root devices and component devices discovered by SL1 in an indented view, so you can easily view the hierarchy and relationships between child devices, parent devices, and root devices. To view the component devices associated with an AWS service, find the AWS virtual device and click its plus icon (+).

Device Name	IP Address	Device Category	Device Class / Sub-class	DID	Organization	Current State	Collection State	Collection Date
1 - AWS Account 911618229564	--	Account	AWS Account	3291	System	Healthy	CUG	Active
1 + AWS GovCloud (US-East: us-gov-east-1)	--	Region	AWS Region GovCloud US East	3293	System	Healthy	CUG	Active
2 - AWS GovCloud (US-West: us-gov-west-1)	--	Region	AWS Region GovCloud US West	3294	System	Healthy	CUG	Active
1 + AWS GovCloud (US-West: us-gov-west-1a)	--	AvailabilityZone	AWS AvailabilityZone - GovCloud US West	3311	System	Healthy	CUG	Active
2 + AWS GovCloud (US-West: us-gov-west-1b)	--	AvailabilityZone	AWS AvailabilityZone - GovCloud US West	3310	System	Healthy	CUG	Active
3 + AWS GovCloud (US-West: us-gov-west-1c)	--	AvailabilityZone	AWS AvailabilityZone - GovCloud US West	3312	System	Healthy	CUG	Active
4 + us-gov-west-1 CloudTrail Service	--	Service	AWS CloudTrail Service	3316	System	Healthy	CUG	Active
5 us-gov-west-1 CloudWatch Service	--	Service	AWS CloudWatch Service	3319	System	Healthy	CUG	Active
6 + us-gov-west-1 EFS Service	--	Service	AWS EFS Service	3324	System	Healthy	CUG	Active
7 + us-gov-west-1 Elastic Beanstalk Service	--	Service	AWS Elastic Beanstalk Service	3321	System	Healthy	CUG	Active
8 + us-gov-west-1 ELB Service	--	Service	AWS ELB Service	3313	System	Healthy	CUG	Active
9 + us-gov-west-1 Glacier Service	--	Service	AWS Glacier Service	3317	System	Healthy	CUG	Active
10 + us-gov-west-1 IoT Service	--	Service	AWS IoT Service	3326	System	Healthy	CUG	Active
11 us-gov-west-1 KMS Service	--	Service	AWS KMS Service	3325	System	Healthy	CUG	Active
12 + us-gov-west-1 Lambda Service	--	Service	AWS Lambda Service	3322	System	Healthy	CUG	Active
13 + us-gov-west-1 S3 Service	--	Service	AWS S3 Service	3315	System	Healthy	CUG	Active
14 + us-gov-west-1 Security	--	Network	AWS Security	3320	System	Healthy	CUG	Active
15 + us-gov-west-1 SNS Service	--	Service	AWS SNS Service	3308	System	Healthy	CUG	Active
16 + us-gov-west-1 SQS Service	--	Service	AWS SQS Service	3314	System	Healthy	CUG	Active
17 + us-gov-west-1 VPC Service	--	Service	AWS VPC Service	3318	System	Healthy	CUG	Active
18 + us-gov-west-1 WAF Regional Service	--	Service	AWS WAF Regional Service	3323	System	Healthy	CUG	Active
2 - AWS Account 84864678992	--	Account	AWS Account	3292	System	Healthy	CUG	Active

- The **Component Map** page (Classic Maps > Device Maps > Components) allows you to view devices by root node and view the relationships between root nodes, parent components, and child components in a map. This makes it easy to visualize and manage root nodes and their components. SL1 automatically updates the **Component Map** as new component devices are discovered. SL1 also updates each map with the latest status and event information. To view the map for an AWS service, go to Classic Maps > Device Maps > Components, and select the map from the list in the left NavBar. To learn more about the **Component Map** page, see the **Maps** manual.



Relationships Between Component Devices

In addition to the parent/child relationships between component devices, relationships are automatically created by the Dynamic Applications in the *Amazon Web Services PowerPack* between the following component devices:

- AWS API Gateway Services and AWS Network Load Balancers
- AWS API Instances and AWS Lambda Functions
- AWS Application ELBs and AWS Availability Zones
- AWS Application ELBs and AWS Route 53-Hosted Zones
- AWS Application ELBs and AWS Security Groups
- AWS Application ELBs and AWS Target Groups
- AWS Application ELBs and AWS VPC Instances
- AWS Auto Scale Groups and AWS Auto Scale Launch Configurations
- AWS Direct Connect Virtual Instances and AWS Virtual Private Gateways
- AWS ECS Instances and AWS EC2 Instances
- AWS ECS Services and AWS Classic Load Balancers
- AWS ECS Services and AWS Security Groups
- AWS ECS Services and AWS Target Groups
- AWS ECS Services and AWS VPC Instances
- AWS ECS Services and AWS VPC Subnets
- AWS EC2 Instances and AWS Auto Scale Groups
- AWS EC2 Instances and AWS EBS Volumes
- AWS EC2 Instances and AWS Elastic Beanstalk Applications

- AWS EC2 Instances and AWS ELB Instances
- AWS EC2 Instances and AWS EMR Instances
- AWS EC2 Instances and AWS OpsWorks Instances
- AWS EC2 Instances and AWS Security Groups
- AWS EC2 Instances and AWS Target Groups
- AWS EC2 Instances and AWS VPC Instances
- AWS EC2 Instances and AWS VPC Subnets
- AWS EC2 Instances and the Cisco Cloud Center application
- AWS Lambda Functions and AWS Security Groups
- AWS Lambda Functions and AWS Simple Notification Services (SNS)
- AWS Lambda Functions and AWS Simple Queue Services (SQS)
- AWS Lambda Functions and AWS VPC Instances
- AWS Lambda Functions and AWS VPC Subnets
- AWS Lambda Function Qualified Services and AWS Security Groups
- AWS Lambda Function Qualified Services and AWS VPC Instances
- AWS Lambda Function Qualified Services and AWS VPC Subnets
- AWS Lambda Function Replicas and their parent AWS Lambda Function Versions
- AWS Network ELBs and AWS Availability Zones
- AWS Network ELBs and AWS Route 53-Hosted Zones
- AWS Network ELBs and AWS Target Groups
- AWS Network ELBs and AWS VPC Instances
- AWS Organizations and AWS Accounts
- AWS RDS Aurora Clusters and AWS RDS DB Instances
- AWS Redshift Instances and AWS Security Groups
- AWS Redshift Instances and AWS VPC Instances
- AWS Route Tables and AWS Virtual Private Gateways
- AWS Route Tables and AWS VPC Subnets
- AWS S3 Instances and AWS CloudTrail Instances
- AWS Security Groups and AWS VPC Instances
- AWS SNS Instances and AWS CloudTrail Instances
- AWS SNS Instances and AWS Glacier Instances
- AWS Transit Gateways and AWS VPC Instances
- AWS VPC Instances and AWS ELB Instances
- AWS VPC Instances and AWS Target Groups
- AWS VPC Instances and other intra-account AWS VPC Instances

- AWS WorkSpaces and AWS VPC Subnets

Vanishing Component Devices

If SL1 cannot retrieve information about a component device for the amount of time specified in the **Component Vanish Timeout** field (in either the **Global Threshold Settings** page, the **Device Thresholds** page for the component device, or the **Device Thresholds** page for a device higher in the component tree), SL1 sets the device to "vanished".

When a device is set to "vanished", SL1 stops trying to collect data about the component device. The vanished device will not appear in reports or views. The vanished device will appear only in the **Vanished Device Manager** page. When a device is set to "vanished", all children of that device are also set to "vanished".

NOTE: This section describes the standard device vanishing behavior that **does not** use the "AWS: Vanish Terminated EC2 Instances" Run Book Action and Automation policies. If you use the "AWS: Vanish Terminated EC2 Instances" Run Book Action and Automation policies, see the chapter on **"AWS Run Book Actions and Automations"** for more information about device vanishing.

Most AWS component devices operate using the standard SL1 vanishing logic: If the device is terminated in AWS, it then becomes unavailable in SL1. If the device is unavailable for the amount of time specified in the **Component Vanish Timeout** field, then that device is vanished.

However, two AWS component device types operate using slightly different logic:

- **EC2.** EC2 instances that are deleted in AWS still appear in the AWS portal for one to two hours in a *terminated* state. If SL1 polls that device and receives a response from AWS that the EC2 is terminated, SL1 will classify the device as unavailable. If the **Component Vanish Timeout** setting has been enabled, then SL1 will vanish this device automatically. If, however, the EC2 instance has merely been *stopped* rather than terminated, SL1 will not vanish the device, even if the **Component Vanish Timeout** setting has been enabled.
- **RDS.** RDS instances that have a status of *stopped* or *stopping* in AWS will be classified as unavailable in SL1. If the **Component Vanish Timeout** setting has been enabled, then SL1 will vanish this device automatically.

ScienceLogic recommends setting the **Component Vanish Timeout** to *120 minutes* when monitoring AWS accounts.

For more information about vanishing devices, see the chapter on "Vanishing & Purging Devices" in the **Device Management** manual.

Configuring Inbound CloudWatch Alarms

Overview

The following sections describe the CloudWatch alarm Event Policies that are included in the *Amazon Web Services PowerPack* and information about configuring CloudWatch and SL1 to generate events based on CloudWatch alarms:

<i>CloudWatch Alarm Event Policies</i>	108
<i>Creating Custom CloudWatch Metrics</i>	110
<i>Configuring CloudWatch to Send Alarms for a Metric</i>	113
<i>Enabling Custom Metrics Collection in SL1</i>	115
<i>Configuring the "AWS: CloudWatch Alarms Performance" Dynamic Application</i>	115
<i>Enabling CloudWatch Alarm Events in SL1</i>	118
<i>Preserving CloudWatch Alarm Event Changes</i>	119

CloudWatch Alarm Event Policies

Amazon CloudWatch is a service that allows you to monitor your AWS resources and applications in near real-time. You can use CloudWatch to collect and track metrics, and use CloudWatch alarms to send notifications or automatically trigger changes to the resources being monitored based on rules that you define.

In addition to SL1 collecting metrics for AWS instances, you can configure CloudWatch to send alarm information to SL1 via API. SL1 can then generate an event for each alarm.

The Amazon Web Services PowerPack includes an "AWS :CloudWatch Alarms Performance" Dynamic Application. This Dynamic Application monitors CloudWatch alarms and associates the alarms with the appropriate AWS component devices, if applicable. If an appropriate component device does not exist in SL1 or cannot be determined, the alarm is instead associated with the component device for the AWS account.

CAUTION: The performance data collected by the "AWS: CloudWatch Alarms Performance" Dynamic Application is metadata intended to give general insight into the alarm activity the Dynamic Application is processing. This metadata can help identify overall trends, but users should be cautioned that the data presented can be imprecise in certain scenarios, such as when the Dynamic Application is being run in debug mode while data is still being collected.

The Amazon Web Services PowerPack also includes several pre-defined event policies for CloudWatch alarms:

Alarm Type	Alarm State	Event Policy Name	Description	Event Source	Severity
Action	Failed	AWS: CloudWatchAlarm_Action_Failed	An Amazon CloudWatch alarm action has failed.	API	Major
Action	InProgress	AWS: CloudWatchAlarm_Action_InProgress	An Amazon CloudWatch alarm action is in progress.	API	Notice
Action	Succeeded	AWS: CloudWatchAlarm_Action_Succeeded	An Amazon CloudWatch alarm action has succeeded.	API	Notice
Configuration Update	Configuration Update	AWS: CloudWatchAlarm_ConfigurationUpdate	A ConfigurationUpdate alarm type is received.	API	Notice
Status Update	Alarm	AWS: CloudWatchAlarm_StateUpdate_Alarm	A CloudWatch alarm transitions to an "Alarm" state.	API	Major
Status Update	Insufficient Data	AWS: CloudWatchAlarm_StateUpdate_InsufficientData	A CloudWatch alarm transitions to an "Insufficient Data" state.	API	Notice
Status Update	OK	AWS: CloudWatchAlarm_StateUpdate_OK	A CloudWatch alarm transitions to an "OK" state.	API	Healthy

These events are aligned to AWS Account component devices in the following way:

- If the CloudWatch alarm is configured on a device that is discovered in SL1, then the event in SL1 will be aligned with the component device for that instance.
- If the CloudWatch alarm is configured on a device that is either not discovered or not supported by CloudWatch, or if SL1 cannot determine a correct component device, then that alarm will be aligned to the Account component device.

The "AWS: CloudWatch Alarms Performance" Dynamic Application and related Event Policies are disabled by default. If you want SL1 to monitor CloudWatch alarms and generate events about them, you must enable the Dynamic Application and Event Policies. You must also configure the Dynamic Application to specify which types of alarms you want to monitor.

For more information about enabling and configuring the "AWS: CloudWatch Alarms Performance" Dynamic Application, see the [Configuring the "AWS: CloudWatch Alarms Performance" Dynamic Application](#) section. For more information about enabling the CloudWatch alarms Event Policies, see the [Enabling CloudWatch Alarm Events in the ScienceLogic Platform](#) section.

NOTE: Because the AWS services make new data points available at varying time intervals, there might be a difference in the data points collected by SL1 when compared to data presented in CloudWatch at a given time. The difference between SL1 and CloudWatch is typically less than 1%.

NOTE: If an event expires and the CloudWatch alarm in AWS is still in an "Alarm" state, SL1 will not generate any additional CloudWatch events unless that CloudWatch alarm changes states in AWS.

Creating Custom CloudWatch Metrics

A CloudWatch alarm watches a single metric and performs one or more actions based on the value of the metric relative to a threshold over a number of time periods. A CloudWatch metric consists of the following elements:

- A **namespace**, such as *AWS/EC2*
- A **metric name**, such as *CPUUtilization*
- A **value**, such as *42*
- A **dimension** that identifies a particular resource instance, such as `{'Name': 'InstanceId', 'Value': 'i-0a6a989bb8d57b074'}`

NOTE: For a complete list of supported CloudWatch Metrics and Dimensions, see https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/CW_Support_For_AWS.html.

The Amazon Web Services PowerPack uses the metric dimensions identified in an alarm to associate the alarm message to a particular ScienceLogic component device. The following table lists the services that are currently supported and the dimensions used to associate an alarm to a component device:

AWS Service	Dimension
API Gateway	'ApiName' 'ApiName Stage' NOTE: ScienceLogic recommends that you create API Gateways with unique names within the same region.
ApplicationELB	'LoadBalancer' 'TargetGroup'
CloudFront	'DistributionId'
Direct Connect	'ConnectionID'
DynamoDB	'TableName'
EBS	'VolumeId'
ECS	'ClusterName' 'ServiceName'
EC2	'InstanceId' 'AutoScalingGroupName'
EKS Cluster	'ClusterName'
ElasticBeanstalk	'EnvironmentName'
ElastiCache	'CacheClusterId' NOTE: Alarms for this service will be associated with the component device for the AWS account.
ElasticMapReduce	'JobFlowId'
ELB	'LoadBalancerName'
Glacier	'VaultId' NOTE: This service is not supported by CloudWatch. You must define a custom metric and publish the metric to the CloudWatch service using an agent toolkit or the AWS command-line interface.
Lambda	'FunctionName', 'Resource', 'Version', 'Alias', 'Executed Version' NOTE: Alarms "across all functions" for this service will be associated with the component device for the AWS account. Alarms "by function name" will be aligned to a specific Lambda function.
NetworkELB	'LoadBalancer' 'TargetGroup'
OpsWorks	'StackId' 'InstanceId'
RDS	'DBInstanceIdentifier' NOTE: Alarms for this service will be associated with the component device for the AWS account.

AWS Service	Dimension
Redshift	'ClusterIdentifier' NOTE: Alarms for this service will be associated with the component device for the AWS account.
Route53	'HealthCheckId'
Shield	'ShieldService' NOTE: CloudWatch alarms are available only for Shield Advanced Services.
SNS	'TopicName'
SQS	'QueueName'
StorageGateway	'GatewayId' 'VolumId'
S3	'BucketName'
WAF	'WebACLId'

AWS enables users to create custom metrics for these services and then publish those metrics to CloudWatch using the AWS command-line interface (CLI) or an application programming interface (API). The Dynamic Applications in the *Amazon Web Services PowerPack* can then collect data for these custom AWS metrics (which are not in the "AWS" cloud namespace).

NOTE: For the *Amazon Web Services PowerPack* to collect data for these custom metrics, you must enable certain Dynamic Applications that are disabled by default. For more information, see the [Enabling Custom Metrics Collection in the ScienceLogic Platform](#) section.

When creating a custom metric, it is important that the metric is correctly formed. For SL1 to align a custom metric to a particular ScienceLogic component device, the following must be true:

- The metric namespace must include the service being tracked.

For example, *MyVendorName/EC2* would be a valid namespace that the *Amazon Web Services PowerPack* could use to identify the EC2 service for a tracked metric.

- The dimension must include one or more of the dimensions listed in the preceding table. The dimension enables SL1 to identify which device to associate with the alarm.

For example, if the dimension included `{'Name': 'InstanceId', 'Value': 'i-0a6a989bb8d57b074'}`, this would identify the EC2 component. Other dimensions are permitted, but 'InstanceId' is necessary to locate the EC2 instance.

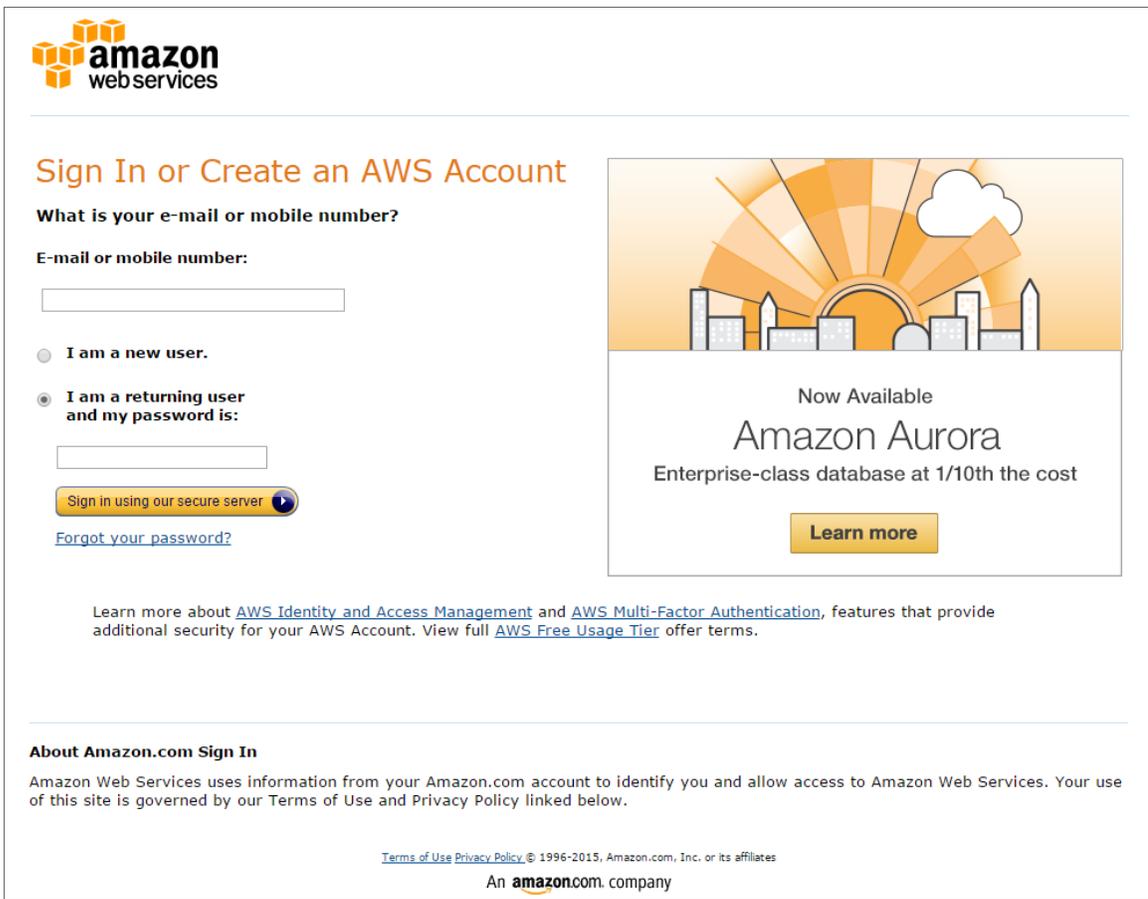
If the component device was an AutoScaleGroup component that is also under the EC2 service, then the dimension might look like this: `{'Name': 'AutoScalingGroupName', 'Value': 'Y1Z55ZJ390UP'}`.

NOTE: If the CloudWatch event cannot align to a particular ScienceLogic component device, it will instead align to the component device for the AWS account.

Configuring CloudWatch to Send Alarms for a Metric

To configure CloudWatch to send alarms to SL1 for a metric, perform the following steps:

1. Open a browser session and go to aws.amazon.com.
2. Click **[My Account]** and then select *AWS Management Console*. If you are not currently logged in to the AWS site, you will be prompted to log in:



The screenshot shows the Amazon Web Services sign-in page. At the top left is the Amazon Web Services logo. The main heading is "Sign In or Create an AWS Account". Below this, it asks "What is your e-mail or mobile number?" and provides an input field. There are two radio button options: "I am a new user." and "I am a returning user and my password is:". The second option is selected. Below the radio buttons is another input field for the password. A "Sign in using our secure server" button is present, along with a link for "Forgot your password?". To the right of the sign-in form is a promotional banner for "Amazon Aurora" with the text "Now Available Amazon Aurora Enterprise-class database at 1/10th the cost" and a "Learn more" button. At the bottom of the page, there is a section titled "About Amazon.com Sign In" with a brief explanation of how Amazon Web Services uses account information. At the very bottom, there are links for "Terms of Use" and "Privacy Policy" and the text "© 1996-2015, Amazon.com, Inc. or its affiliates" and "An amazon.com company".

3. In the **AWS Management Console**, under the **Management Tools** heading, click **[CloudWatch]**.
4. Click the **[Browse Metrics]** button.
5. Select the metric for which you want CloudWatch to send alarms.
6. Select the instances for which you want CloudWatch to send alarms for this metric.

7. Click the **[Create Alarm]** button. The **Create Alarm** page is displayed:

Create Alarm [X]

1. [Select Metric](#) 2. **Define Alarm**

Alarm Threshold

Provide the details and threshold for your alarm. Use the graph on the right to help set the appropriate threshold.

Name:

Description:

Whenever: CPUUtilization

is:

for: consecutive period(s)

Alarm Preview

This alarm will trigger when the blue line goes up to or above the red line for a duration of 5 minutes

CPUUtilization >= 0

Namespace: AWS/EC2

InstanceId:

InstanceName: student13

Metric Name:

Period:

Statistic:

Actions

Define what actions are taken when your alarm changes state.

Notification Delete

Whenever this alarm:

Send notification to: [New list](#) [Enter list](#) ⓘ

[Cancel](#)

8. Specify a Name and Description for the alarm.
9. If you have previously configured an alarm for SL1, select the notification list for SL1 in the **Send notification to** field. Otherwise, select the **[New list]** link to the right of the **Send notification to** field and supply values in the following fields:
 - **Send notification to.** Enter a name for the new notification list. If you add additional alarms, you can select the name you enter in this field instead of re-entering the email address.
 - **Email list.** Enter the email address to which you want CloudWatch notifications sent.
10. Supply values in the other fields in this page as desired.
11. Click the **[Create Alarm]** button.
12. Log in to the email account you configured to receive email from the email alias.
13. Open the confirmation email from Amazon and click the **[Confirm subscription]** link.

Enabling Custom Metrics Collection in SL1

AWS enables users to publish their own custom metrics to CloudWatch using the AWS command-line interface (CLI) or an application programming interface (API). The *Amazon Web Services PowerPack* includes Dynamic Applications that collect data for custom AWS metrics (which are not in the "AWS" cloud namespace). However, these Dynamic Applications are disabled by default and must be enabled for use.

To enable these Dynamic Applications:

1. Go to the **Dynamic Applications Manager** page (System > Manage > Applications).
2. Click the wrench icon () for the "AWS: Custom Metrics" Dynamic Application. The **Dynamic Applications Properties Editor** page appears.
3. In the **Operational State** field, select *Enabled*.
4. Click the **[Save]** button.
5. Repeat steps 1 - 4 for the "AWS: Custom Metrics Cache" Dynamic Application.

Configuring the "AWS: CloudWatch Alarms Performance" Dynamic Application

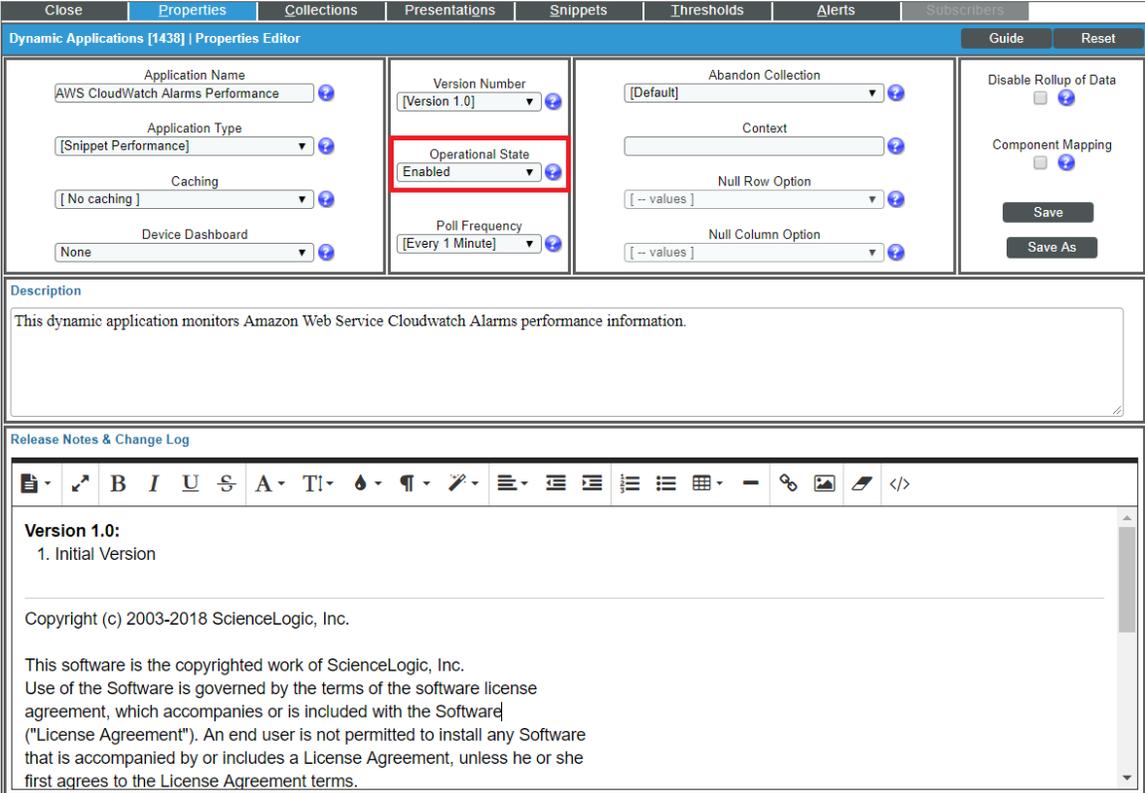
The *Amazon Web Services PowerPack* includes an "AWS: CloudWatch Alarms Performance" Dynamic Application that monitors CloudWatch alarms and associates the alarms with the appropriate AWS component devices, if applicable. This Dynamic Application must be enabled if you want SL1 to generate CloudWatch alarm events.

NOTE: If an appropriate component device does not exist in SL1 or cannot be determined, the alarm is instead associated with the "Account" component device.

To enable the "AWS: CloudWatch Alarms Performance" Dynamic Application:

1. Go to the **Dynamic Applications Manager** page (System > Manage > Applications).

2. Locate the "AWS: CloudWatch Alarms Performance" Dynamic Application and then click its wrench icon (). The **Dynamic Applications Properties Editor** page appears.



The screenshot shows the 'Dynamic Applications Properties Editor' for the application 'AWS: CloudWatch Alarms Performance'. The interface includes several configuration sections:

- Application Name:** AWS CloudWatch Alarms Performance
- Application Type:** [Snippet Performance]
- Caching:** [No caching]
- Device Dashboard:** None
- Version Number:** [Version 1.0]
- Operational State:** Enabled (highlighted with a red box)
- Poll Frequency:** [Every 1 Minute]
- Abandon Collection:** [Default]
- Context:** [Empty field]
- Null Row Option:** [-- values]
- Null Column Option:** [-- values]
- Disable Rollup of Data:**
- Component Mapping:**

Below the configuration fields, there is a **Description** field containing the text: "This dynamic application monitors Amazon Web Service Cloudwatch Alarms performance information." and a **Release Notes & Change Log** section with a rich text editor. The release notes include:

Version 1.0:
1. Initial Version

Copyright (c) 2003-2018 ScienceLogic, Inc.

This software is the copyrighted work of ScienceLogic, Inc. Use of the Software is governed by the terms of the software license agreement, which accompanies or is included with the Software ("License Agreement"). An end user is not permitted to install any Software that is accompanied by or includes a License Agreement, unless he or she first agrees to the License Agreement terms.

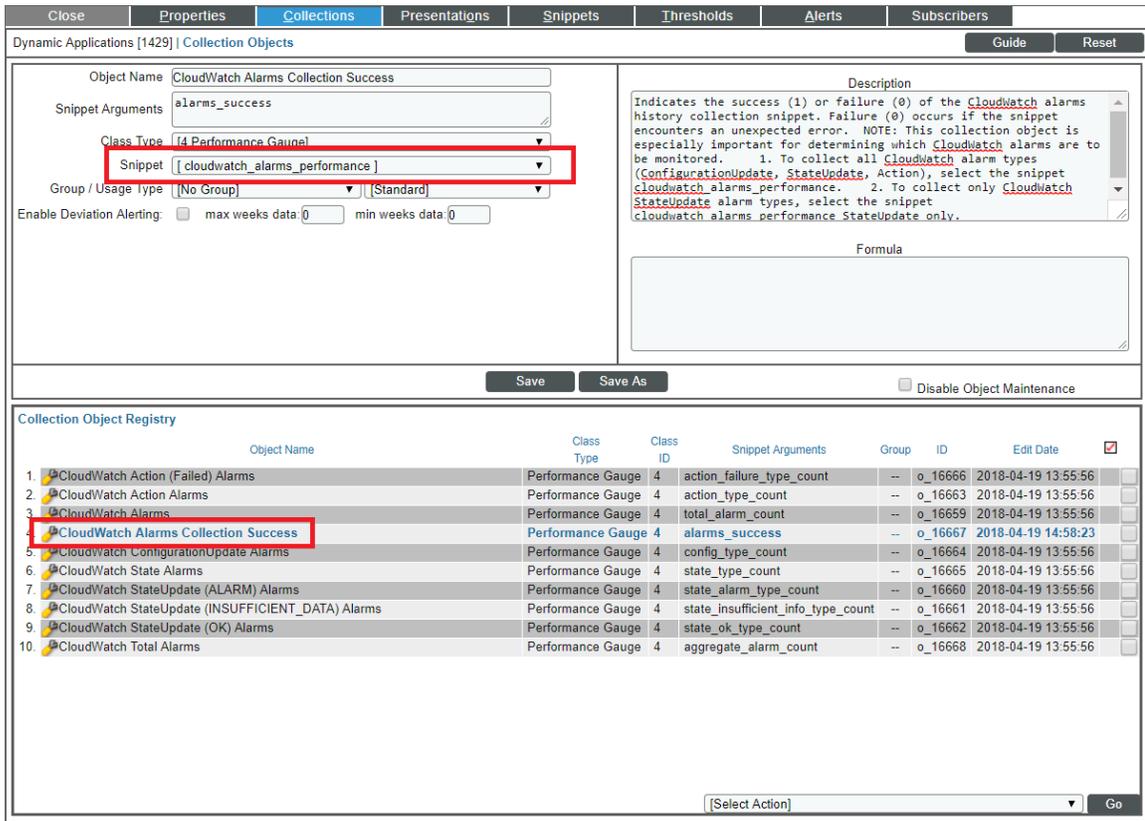
3. In the **Operational State** field, select *Enabled*.
4. Click **[Save]**.

By default, the "AWS: CloudWatch Alarms Performance" Dynamic Application monitors only the "StateUpdate" type of CloudWatch alarms. If you want the Dynamic Application to also monitor "Action" and "ConfigurationUpdate" alarm types, you must configure the Dynamic Application to do so.

To configure the "AWS: CloudWatch Alarms Performance" Dynamic Application to monitor all CloudWatch alarm types:

1. Go to the **Dynamic Applications Manager** page (System > Manage > Applications).
2. Locate the "AWS: CloudWatch Alarms Performance" Dynamic Application and then click its wrench icon (). The **Dynamic Applications Properties Editor** page appears.
3. Click the **[Collections]** tab. The **Collection Objects** page appears.

- On the **Collection Objects** page, locate the "CloudWatch Alarms Collection Success" collection object and then click its wrench icon ().



Dynamic Applications [1429] | Collection Objects

Object Name: CloudWatch Alarms Collection Success

Snippet Arguments: alarms_success

Class Type: Performance Gauge

Snippet: cloudwatch_alarms_performance

Group / Usage type: (No Group) / (Standard)

Enable Deviation Alerting: max weeks data: 0 min weeks data: 0

Description: Indicates the success (1) or failure (0) of the CloudWatch alarms history collection snippet. Failure (0) occurs if the snippet encounters an unexpected error. NOTE: This collection object is especially important for determining which CloudWatch alarms are to be monitored. 1. To collect all CloudWatch alarm types (ConfigurationUpdate, StateUpdate, Action), select the snippet cloudwatch_alarms_performance. 2. To collect only CloudWatch StateUpdate alarm types, select the snippet cloudwatch_alarms_performance_StateUpdate_only.

Formula:

Save Save As Disable Object Maintenance

Collection Object Registry

	Object Name	Class Type	Class ID	Snippet Arguments	Group	ID	Edit Date	
1.	CloudWatch Action (Failed) Alarms	Performance Gauge	4	action_failure_type_count	--	o_16666	2018-04-19 13:55:56	<input type="checkbox"/>
2.	CloudWatch Action Alarms	Performance Gauge	4	action_type_count	--	o_16663	2018-04-19 13:55:56	<input type="checkbox"/>
3.	CloudWatch Alarms	Performance Gauge	4	total_alarm_count	--	o_16659	2018-04-19 13:55:56	<input type="checkbox"/>
4.	CloudWatch Alarms Collection Success	Performance Gauge	4	alarms_success	--	o_16667	2018-04-19 14:58:23	<input type="checkbox"/>
5.	CloudWatch ConfigurationUpdate Alarms	Performance Gauge	4	config_type_count	--	o_16664	2018-04-19 13:55:56	<input type="checkbox"/>
6.	CloudWatch State Alarms	Performance Gauge	4	state_type_count	--	o_16665	2018-04-19 13:55:56	<input type="checkbox"/>
7.	CloudWatch StateUpdate (ALARM) Alarms	Performance Gauge	4	state_alarm_type_count	--	o_16660	2018-04-19 13:55:56	<input type="checkbox"/>
8.	CloudWatch StateUpdate (INSUFFICIENT_DATA) Alarms	Performance Gauge	4	state_insufficient_info_type_count	--	o_16661	2018-04-19 13:55:56	<input type="checkbox"/>
9.	CloudWatch StateUpdate (OK) Alarms	Performance Gauge	4	state_ok_type_count	--	o_16662	2018-04-19 13:55:56	<input type="checkbox"/>
10.	CloudWatch Total Alarms	Performance Gauge	4	aggregate_alarm_count	--	o_16668	2018-04-19 13:55:56	<input type="checkbox"/>

[Select Action] Go

- In the **Snippet** field, select one of the following options:

- `cloudwatch_alarms_performance`. This option is selected by default. This snippet triggers notifications if any alarm configuration is modified.
- `cloudwatch_alarms_performance_StateUpdate_only`. This snippet will only trigger events for State Update alarms.
- `cloudwatch_alarms_statistics`. This snippet will trigger events for all CloudWatch alarm types (Action, Configuration Update, and State Update).

NOTE: If you want to revert back to monitoring only the "StateUpdate" CloudWatch alarms, then select `cloudwatch_alarms_performance_StateUpdate_only` in the **Snippet** field.

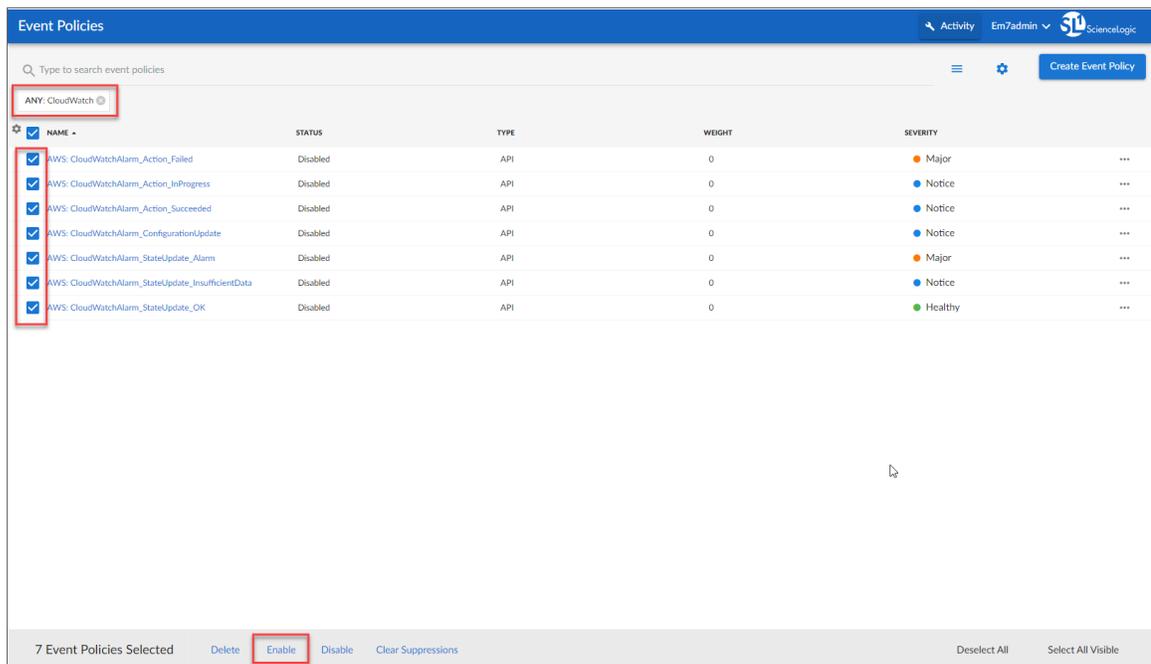
- Click **[Save]**. This Dynamic Application will be executed every 1 minute.

Enabling CloudWatch Alarm Events in SL1

The Amazon Web Services PowerPack also includes several pre-defined event policies for CloudWatch alarms. These Event Policies must be enabled if you want SL1 to generate CloudWatch alarm events.

To enable the CloudWatch alarms Event Policies:

1. Go to the **Event Policies** page (Events > Event Policies).
2. Perform a search for "CloudWatch".



3. Select the check boxes for the events you want to enable.
4. Select **Enable** at the bottom of the screen.

To enable the CloudWatch alarms Event Policies in the SL1 classic user interface:

1. Go to the **Event Policy Manager** page (Registry > Events > Event Manager).

- In the **Event Policy Name** filter-while-you-type field, type "CloudWatch".

The screenshot shows the 'Event Policy Manager' interface with a table of policies. The filter 'Event Policy Name' is set to 'CloudWatch'. The table lists seven policies, all of which are enabled. A dropdown menu is open, showing the following options: '[Select Action]', 'Administration:', 'DELETE these Event Policies', 'ENABLE these Event Policies' (highlighted), 'DISABLE these Event Policies', 'CLEAR the Suppression List', and '[Select Action]'. A 'Go' button is visible at the bottom right of the dropdown.

	Event Policy Name	Type	State	P-Pack	Severity	Weight	ID	Expiry	Time	Thresh	Edited By	Last Edited	External ID	Ext. Category	
1.	AWS: CloudWatchAlarm_Action_Failed	API	Enabled	Yes	Major	0	4234	90 Min.	0 Min.	0	em7admin	2018-04-17 09:56:26	--	--	<input checked="" type="checkbox"/>
2.	AWS: CloudWatchAlarm_Action_InProgress	API	Enabled	Yes	Notice	0	4236	30 Min.	0 Min.	0	em7admin	2018-04-17 09:56:26	--	--	<input checked="" type="checkbox"/>
3.	AWS: CloudWatchAlarm_Action_Succeeded	API	Enabled	Yes	Notice	0	4233	30 Min.	0 Min.	0	em7admin	2018-04-17 09:56:26	--	--	<input checked="" type="checkbox"/>
4.	AWS: CloudWatchAlarm_ConfigurationUpdate	API	Enabled	Yes	Notice	0	4235	30 Min.	0 Min.	0	em7admin	2018-04-17 09:56:26	--	--	<input checked="" type="checkbox"/>
5.	AWS: CloudWatchAlarm_StateUpdate_Alarm	API	Enabled	Yes	Major	0	4230	90 Min.	0 Min.	0	em7admin	2018-04-17 09:56:26	--	--	<input checked="" type="checkbox"/>
6.	AWS: CloudWatchAlarm_StateUpdate_InsufficientData	API	Enabled	Yes	Notice	0	4231	30 Min.	0 Min.	0	em7admin	2018-04-17 09:56:27	--	--	<input checked="" type="checkbox"/>
7.	AWS: CloudWatchAlarm_StateUpdate_OK	API	Enabled	Yes	Healthy	0	4232	15 Min.	0 Min.	0	em7admin	2018-04-17 09:56:27	--	--	<input checked="" type="checkbox"/>

- Select the check boxes for the events you want to enable.
- In the **Select Action** drop-down field, select *ENABLE these Event Policies*.
- Click **[Go]**.

Preserving CloudWatch Alarm Event Changes

If you have modified CloudWatch alarm event policies that are included in the *Amazon Web Services PowerPack*, those changes will be overwritten when the PowerPack is updated in your system. If you have modified event policies that are included in the PowerPack, you can:

- Re-implement those changes after each update of the *Amazon Web Services PowerPack*.
- Remove the content from the PowerPack on your system. When the *Amazon Web Services PowerPack* is updated in your system, updated versions of this content will not be installed on your system and your local changes will be preserved.

To remove event policies from the *Amazon Web Services PowerPack* on your system:

- Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
- Click the wrench icon (🔧) for the *Amazon Web Services PowerPack*. The **Editing PowerPack** page appears.
- In the left NavBar of the **Editing PowerPack** page, click **[Event Policies]**. The **Embedded Event Policies** and **Available Event Policies** panes appear.
- In the upper pane, click the bomb icon (💣) for each event policy that you want to remove from the *Amazon Web Services PowerPack* on your system.

Chapter

8

Reports

Overview

The following sections describe the reports that are included in the *Amazon Web Services PowerPack*:

<i>AWS Billing Report</i>	121
<i>AWS Inventory Report</i>	123
<i>AWS Running Config Report</i>	125

AWS Billing Report

This report displays service costs for Amazon Web Services. The report includes Total, Monthly, Quarterly, and Annual costs.



AWS Billing Report – Total Service Costs

Report Start Date: 2014/04
 Report Duration: To present
 * Billing data may be inaccurate due to missed polls.

Account: (none)		
Service	# Instances	Total Cost
	0	\$0.00
Total for Account: (none)	0	\$0.00
Account: AIDAJ5CRUCDWA7CRUTMS [14115]		
Service	# Instances	Total Cost
SQS	2	\$0.00
EC2	72	\$0.00
SNS	15	\$0.00
Total for Account: AIDAJ5CRUCDWA7	89	\$0.00
Overall Totals:	89	\$0.00

Generated on: 2015/04/17 07:46:56



Monthly Costs

AWS Billing Report – Monthly Costs

Account: (none)													
Region	Service	Apr 2014	May 2014	Jun 2014	Jul 2014	Aug 2014	Sep 2014	Oct 2014	Nov 2014	Dec 2014	Jan 2015	Feb 2015	Mar 2015
Total for Account: (none)		\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00
Account: AIDAJ5CRUCDWA7CRUTMS [14115]													
Region	Service	Apr 2014	May 2014	Jun 2014	Jul 2014	Aug 2014	Sep 2014	Oct 2014	Nov 2014	Dec 2014	Jan 2015	Feb 2015	Mar 2015
Frankfurt-central-1 [eu-central-1]	SQS	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00
Frankfurt-central-1 [eu-central-1]	EC2	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00
Frankfurt-central-1 [eu-central-1]	SNS	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00
Total for Account: AIDAJ5CRUCDWA7CRUTMS [14115]		\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00
Overall Totals:		\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00	\$0.00

Generated on: 2015/04/17 07:46:56



AWS Billing Report – Quarterly Costs

Account: (none)					
Region	Service	Q2 2014	Q3 2014	Q4 2014	Q1 2015
		\$0.00	\$0.00	\$0.00	\$0.00
Total for Account: (none)		\$0.00	\$0.00	\$0.00	\$0.00
Account: AIDAJ5CRUCDWAU7CRUTMS [14115]					
Region	Service	Q2 2014	Q3 2014	Q4 2014	Q1 2015
Frankfurt: eu-central-1 [14444]	SQS	\$0.00	\$0.00	\$0.00	\$0.00
Frankfurt: eu-central-1 [14444]	EC2	\$0.00	\$0.00	\$0.00	\$0.00
	SNS	\$0.00	\$0.00	\$0.00	\$0.00
Total for Account: AIDAJ5CRUCDWAU7CRUTMS [14115]		\$0.00	\$0.00	\$0.00	\$0.00
Overall Totals:		\$0.00	\$0.00	\$0.00	\$0.00

Generated on: 2015/04/17 07:46:56



AWS Billing Report – Annual Costs

Account: (none)			
Region	Service	2014	2015
		\$0.00	\$0.00
Total for Account: (none)		\$0.00	\$0.00
Account: AIDAJ5CRUCDWAU7CRUTMS [14115]			
Region	Service	2014	2015
Frankfurt: eu-central-1 [14444]	SQS	\$0.00	\$0.00
Frankfurt: eu-central-1 [14444]	EC2	\$0.00	\$0.00
	SNS	\$0.00	\$0.00
Total for Account: AIDAJ5CRUCDWAU7CRUTMS [14115]		\$0.00	\$0.00
Overall Totals:		\$0.00	\$0.00

Generated on: 2015/04/17 07:46:56



AWS Billing Report – Control

Description:	AWS Billing
Report Version:	1.1
Generated On:	2015/04/17 07:46:56
AWS Accounts:	All
Start Date:	2014/04
Duration:	To present

Generated on: 2015/04/17 07:46:56

The following input options are available when generating the report (Reports > Run Report > Cloud > AWS Billing):

- **AWS Accounts.** Select the AWS Account(s) for which you want to generate the report. The *All Accounts* checkbox is selected by default. De-selecting this checkbox allows you to select one or more specific accounts for which to generate a report.
- **Report Span.** Select a span from one to 36 months for the report, or specify a specific starting date for the report.

This description covers the latest version of this report as shipped by ScienceLogic. This report might have been modified on your SL1 system.

AWS Inventory Report

This report displays an inventory of AWS instance counts. The report includes the number of each kind of instance in every zone associated with the chosen accounts. It also includes a count of each EC2 instance size in each zone.



AWS Inventory Report – Instance Counts

Organization: Pittock [193]																	
Account: AIDAJ5CRUCDWA7CRUTMS [14115]																	
Level1: CloudFront Service [14120]																	
Zone	Glacier	Launch Con AS Group	Web Dist	udFront Ori	CloudTrail	ELB	Subnet	SNS	EC2	RDS	3 Health Ch3	Hosted Zo	S3	SQS	EBS	VPC	
d12nhk6qht264.cloudfront.net [14150]	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	
Totals for Level1: CloudFront Service [14120]	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	
Level1: Frankfurt: eu-central-1 [14444]																	
Zone	Glacier	Launch Con AS Group	Web Dist	udFront Ori	CloudTrail	ELB	Subnet	SNS	EC2	RDS	3 Health Ch3	Hosted Zo	S3	SQS	EBS	VPC	
eu-central-1 Glacier Service [14467]	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
eu-central-1 VPC Service [14447]	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	1	
eu-central-1a [14446]	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	
Totals for Level1: Frankfurt: eu-central-1 [14444]	1	0	0	0	0	0	0	2	0	1	0	0	0	0	0	1	
Level1: Ireland: eu-west-1 [14117]																	
Zone	Glacier	Launch Con AS Group	Web Dist	udFront Ori	CloudTrail	ELB	Subnet	SNS	EC2	RDS	3 Health Ch3	Hosted Zo	S3	SQS	EBS	VPC	
eu-west-1 Glacier Service [14129]	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	
eu-west-1 CloudTrail Service [14346]	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	
eu-west-1 ELB Service [14124]	0	0	0	0	0	1	0	0	7	0	0	0	0	0	0	0	
eu-west-1 SNS Service [14123]	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	
eu-west-1 VPC Service [14130]	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	2	
Totals for Level1: Ireland: eu-west-1 [14117]	1	0	0	0	0	1	1	1	7	0	0	0	1	0	0	2	
Level1: N. Virginia: us-east-1 [14118]																	
Zone	Glacier	Launch Con AS Group	Web Dist	udFront Ori	CloudTrail	ELB	Subnet	SNS	EC2	RDS	3 Health Ch3	Hosted Zo	S3	SQS	EBS	VPC	
us-east-1 Auto Scale Service [14138]	0	2	1	0	0	2	0	0	38	0	0	0	0	0	0	0	
us-east-1 CloudTrail Service [14139]	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	
us-east-1b [14133]	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	
us-standard S3 Service [14137]	0	0	0	0	0	0	0	0	0	0	0	0	5	0	41	0	
us-east-1 SQS Service [14340]	0	0	0	0	0	0	0	8	0	0	0	0	0	1	0	0	
us-east-1 VPC Service [14141]	0	0	0	0	0	0	0	8	0	0	0	0	0	0	0	6	
Totals for Level1: N. Virginia: us-east-1 [14118]	0	2	1	0	0	1	2	8	38	3	0	0	5	1	41	6	
Level1: Oregon: us-west-2 [14119]																	
Zone	Glacier	Launch Con AS Group	Web Dist	udFront Ori	CloudTrail	ELB	Subnet	SNS	EC2	RDS	3 Health Ch3	Hosted Zo	S3	SQS	EBS	VPC	
us-west-2 Auto Scale Service [14147]	0	1	1	0	0	0	0	0	9	0	0	0	0	0	0	0	
us-west-2 CloudTrail Service [14148]	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	
us-west-2 S3 Service [14146]	0	0	0	0	0	0	0	0	0	0	0	0	3	0	6	0	
us-west-2 SQS Service [14336]	0	0	0	0	0	0	0	4	0	0	0	0	0	1	0	0	
us-west-2 VPC Service [14149]	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	1	
Totals for Level1: Oregon: us-west-2 [14119]	0	1	1	0	0	1	0	3	4	0	0	0	3	1	6	1	
Level1: Route 53 Service [14116]																	
Zone	Glacier	Launch Con AS Group	Web Dist	udFront Ori	CloudTrail	ELB	Subnet	SNS	EC2	RDS	3 Health Ch3	Hosted Zo	S3	SQS	EBS	VPC	
mapmycloud.net [14121]	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	
Totals for Level1: Route 53 Service [14116]	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	
Totals for Account: AIDAJ5CRUCDWA7CRUTMS [14115]																	
	2	3	2	1	1	3	3	22	13	55	3	1	1	9	2	56	10
Totals for Organization: Pittock [193]																	
	2	3	2	1	1	3	3	22	13	55	3	1	1	9	2	56	10
Overall Totals:																	
	2	3	2	1	1	3	3	22	13	55	3	1	1	9	2	56	10

Generated on: April 17th, 2015 at 7:46am



AWS Inventory Report – EC2 Instance Details

Organization: Pittcock [193]										
Account: AIDAJ5CRUCDWAW7CRUTMS [14115]										
Level1: Frankfurt: eu-central-1 [14444]										
Zone	M1.small	M3.large	T1.micro	T2.small	T2.micro	C3.large	M3.xlarge	M3.medium	M1.medium	
eu-central-1a [14446]	0	0	0	0	1	0	0	0	0	
Totals for Level1: Frankfurt: eu-central-1 [14444]	0	0	0	0	1	0	0	0	0	0
Level1: Ireland: eu-west-1 [14117]										
Zone	M1.small	M3.large	T1.micro	T2.small	T2.micro	C3.large	M3.xlarge	M3.medium	M1.medium	
eu-west-1a [14126]	0	1	2	0	0	0	0	0	0	
eu-west-1c [14127]	0	0	2	0	0	0	0	0	0	
eu-west-1b [14125]	0	0	2	0	0	0	0	0	0	
Totals for Level1: Ireland: eu-west-1 [14117]	0	1	6	0	0	0	0	0	0	0
Level1: N. Virginia: us-east-1 [14118]										
Zone	M1.small	M3.large	T1.micro	T2.small	T2.micro	C3.large	M3.xlarge	M3.medium	M1.medium	
us-east-1a [14134]	4	4	3	11	1	0	0	0	0	
us-east-1e [14135]	0	0	0	3	0	0	1	0	0	
us-east-1b [14133]	1	0	4	0	0	0	0	0	1	
us-east-1c [14136]	2	0	2	0	0	1	0	0	0	
Totals for Level1: N. Virginia: us-east-1 [14118]	7	4	9	11	4	1	1	0	1	1
Level1: Oregon: us-west-2 [14119]										
Zone	M1.small	M3.large	T1.micro	T2.small	T2.micro	C3.large	M3.xlarge	M3.medium	M1.medium	
us-west-2c [14145]	0	0	4	0	0	0	0	1	0	
us-west-2a [14144]	0	0	3	0	0	0	0	0	0	
us-west-2b [14143]	0	0	0	0	0	0	0	0	1	
Totals for Level1: Oregon: us-west-2 [14119]	0	0	7	0	0	0	0	2	0	0
Totals for Account: AIDAJ5CRUCDWAW7CRUTMS [14115]	7	5	22	11	5	1	1	2	1	1
Totals for Organization: Pittcock [193]	7	5	22	11	5	1	1	2	1	1
Overall Totals:	7	5	22	11	5	1	1	2	1	1

Generated on: April 17th, 2015 at 7:46am

The following input options are available when generating the report (Reports > Run Report > Others > AWS Inventory):

- **Organizations.** Select the organization for which you want to generate the report. The *All Organizations* checkbox is selected by default. De-selecting this checkbox allows you to select one or more specific organizations for which to generate a report.
- **AWS Accounts.** Select the AWS Account(s) for which you want to generate the report. The *All Accounts* checkbox is selected by default. De-selecting this checkbox allows you to select one or more specific accounts for which to generate a report.
- **Filter on EC2 Instance Config Data.** Select the EC2 instances that will be included in the report based on the configuration data reported for each EC2 instance:
 - Choose up to four configuration parameters for EC2 instances.
 - For each selected configuration parameter, enter a value to match against and select how that value should be matched.
 - In the **Comparison Operator** field, select whether an EC2 instance must match all configuration parameters (*and*) or only one configuration parameter (*or*) to be included on the report.
- **Report Options.** Select the *Include Terminated Instances* checkbox to include all terminated instances.

This description covers the latest version of this report as shipped by ScienceLogic. This report might have been modified on your SL1 system.

AWS Running Config Report

This report displays the running config of all AWS instances for one to all organizations across a number of AWS billing accounts.

Key	Value
**** Application ****: *** AWS CloudFront Origin Discovery ***	
Distinguished Name:	am:aws:cloudfront:789135808643:distribution/E1KPRUBCK0YU3E
Exists:	1
Id:	cloudfront E1KPRUBCK0YU3E@s3.amazonaws.com
Name:	s3cloudtrail.s3.amazonaws.com
**** Application ****: *** AWS CloudFront Web Distribution ***	
Trusted Signers:	
Id:	cloudfront E1KPRUBCK0YU3E
State:	True
Distinguished Name:	am:aws:cloudfront:789135808643:distribution/E1KPRUBCK0YU3E
Comment:	
Delivery Method:	Web
Price Class:	Not Available
Name:	d12ibk6qbt264.cloudfront.net
Last Modified:	2014-09-18T03:25:03.777Z
CNames:	
Status:	Deployed
**** Application ****: *** AWS CloudFront Restriction Discovery ***	
Exists:	1
**** Application ****: *** AWS CloudFront Error Page Discovery ***	
Exists:	1
**** Application ****: *** AWS CloudFront Behavior Discovery ***	
Exists:	1

The following input options are available when generating the report (Reports > Run Report > Others > AWS Running Config):

- **Organizations.** Select one, multiple, or all organizations to include in the report.
 - *All Organizations.* This checkbox is selected by default. De-selecting this checkbox allows you to select one or more specific organizations for the report.
 - *Organizations.* If you unchecked the **All Organizations** checkbox, select one or more organizations to include in the report.
- **AWS Accounts.** Select one, multiple, or all AWS Accounts to include in the report.
 - *All Accounts.* This checkbox is selected by default. De-selecting this checkbox allows you to select one or more specific AWS accounts for the report.
 - *Accounts.* If you unchecked the **All Accounts** checkbox, select one or more AWS Accounts to include in the report.

- **Filter on EC2 Instance Config Data.** Select the EC2 instances that will be included on the report based on the configuration data reported for each EC2 instance:
 - Choose up to four configuration parameters for EC2 instances.
 - For each selected configuration parameter, enter a value to match against and select how that value should be matched.
 - In the **Comparison Operator** field, select whether an EC2 instance must match all configuration parameters (*and*) or only one configuration parameter (*or*) to be included in the report.
- **Report Options.** Select the *Include Terminated Instances* checkbox to include all terminated instances.

This description covers the latest version of this report as shipped by ScienceLogic. This report might have been modified on your SL1 system.

Chapter

9

Dashboards

Overview

The following sections describe how to install the *Amazon Web Services: Dashboards* PowerPack and a description of each dashboard that is included in the PowerPack:

<i>Installing the Amazon Web Services: Dashboards PowerPack</i>	127
<i>AWS Account Billing Dashboard</i>	128
<i>AWS Health Status Dashboard</i>	129
<i>Configuring the AWS Dashboards</i>	130
<i>AWS Service Instance Performance Dashboards</i>	131

Installing the Amazon Web Services: Dashboards PowerPack

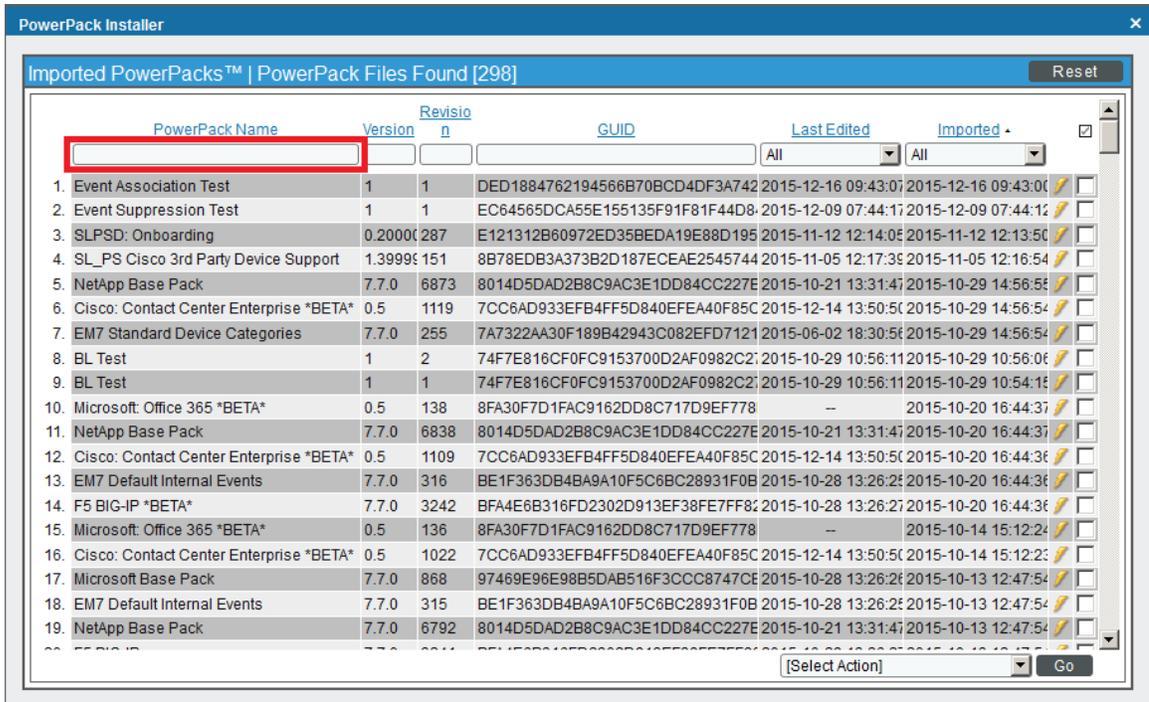
To view the Amazon Web Services dashboards in SL1, you must first install the *Amazon Web Services: Dashboards* PowerPack.

NOTE: The AWS dashboards have a default **Access Control** setting of "Private", which means they can be viewed only by an administrator. For more information about dashboard access settings, see the *Dashboards* manual.

To install the PowerPack:

1. Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
2. Click the **[Actions]** button, then select *Install PowerPack*. The **Imported PowerPacks** modal page appears.

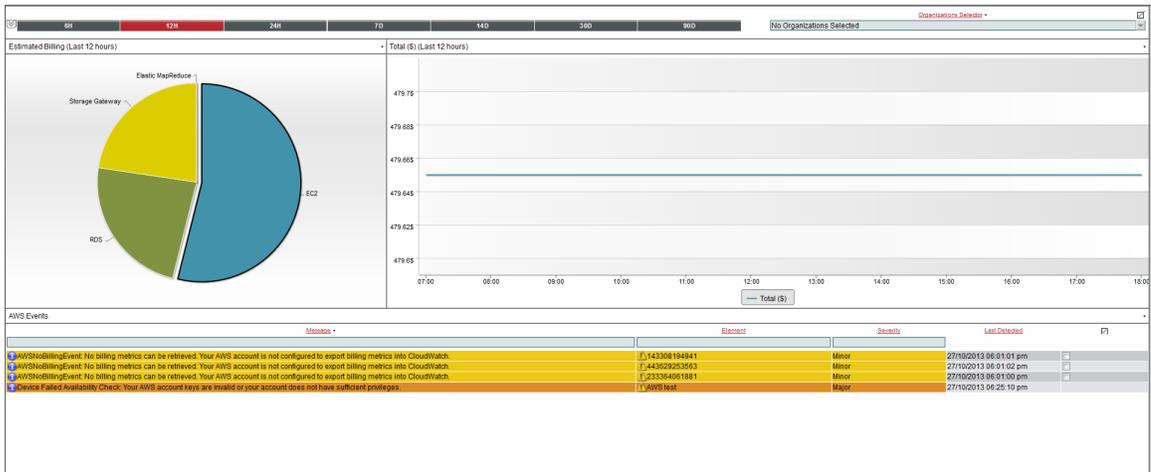
- Use the search filter in the **PowerPack Name** column heading to locate the PowerPack you want to install. To do so, enter text to match, including special characters, and the **Imported PowerPacks** modal page displays only PowerPacks that have a matching name.



- Click the lightning-bolt icon (⚡) for the PowerPack that you want to install.
- The **Install PowerPack** modal page appears. To install the PowerPack, click **[Install]**.
- The PowerPack now appears in the **PowerPack Manager** page. The contents of the PowerPack are automatically installed in your SL1 System.

AWS Account Billing Dashboard

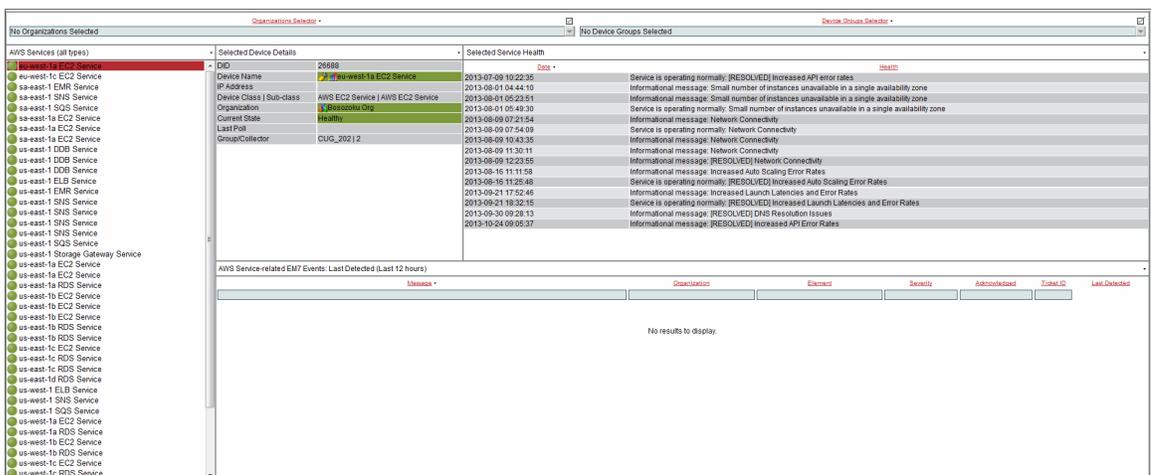
The AWS Account Billing Dashboard displays:



- A pie chart that shows the estimated billing amount for each service over the selected time period.
- A performance graph that shows the estimated billing amount for the selected service, over time. To select a service, click on the pie-chart segment for that service.
- A table that shows the currently active AWS events.
- A time span selector that controls the amount of data shown in the pie chart and the performance graph.
- An organization selector that limits the data in the pie chart and performance graph to include only instances associated with the selected organizations.

AWS Health Status Dashboard

The AWS Health Status Dashboard displays:



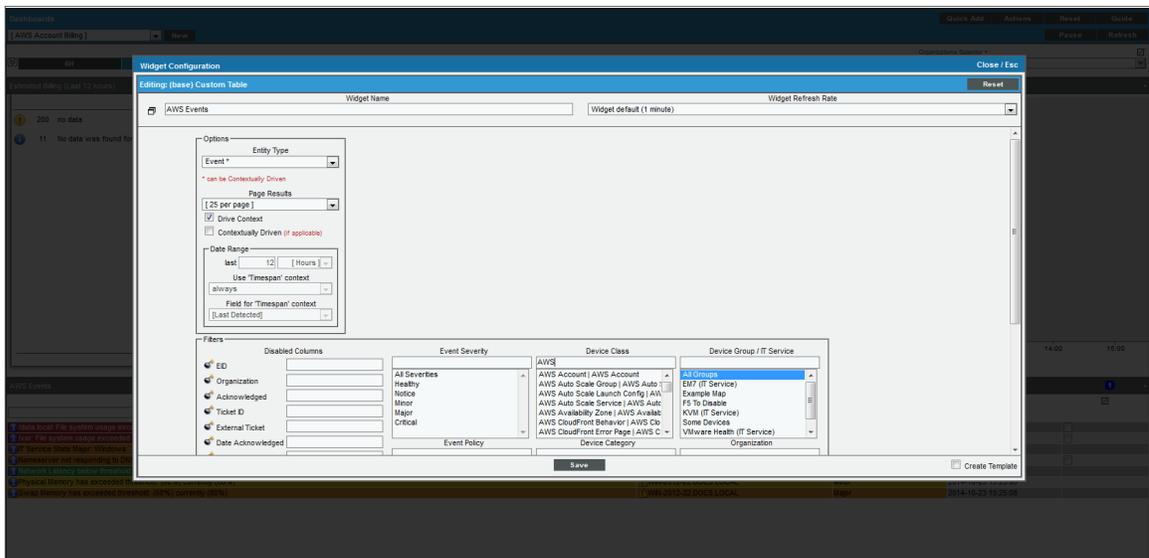
- A traffic light widget that displays a list of AWS services. To populate the other widgets in this dashboard, select a service.

- A tear-sheet widget that displays information and links for the selected service.
- A service health widget, that displays log messages about the health of the service.
- A table that displays currently active events for the service.
- An organization selector and a device group selector that control which services are shown in the traffic light widget.

Configuring the AWS Dashboards

The AWS Account Billing and AWS Health Status dashboards must have their (base) Custom Table widgets manually configured to filter only AWS service-specific events. To do this:

1. Go to Dashboards > Classic Dashboards and select AWS Account Billing, or in the SL1 classic user interface go to Dashboards and select AWS Account Billing.
2. Click the down-arrow in the upper-right of the AWS Events widget, and then select *Configure* from the **Options** menu. The **Widget Configuration** modal page appears.
3. In the **Device Class** filter, enter "AWS" to show only AWS device classes:



4. Control-click on the following items in the **Device Class** field:
 - AWS DDB Service
 - AWS EC2 Service
 - AWS ELB Service
 - AWS EMR Service
 - AWS RDS Service
 - AWS SNS Service

- AWS SQS Service
- AWS Storage Gateway Service

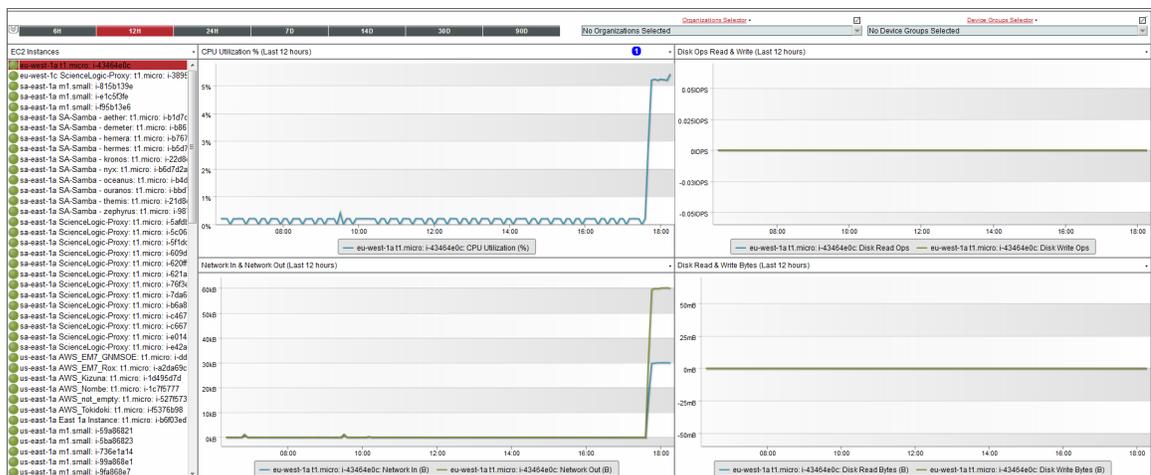
5. Click the **[Save]** button.
6. Repeat steps 1 - 5 for the AWS Health Status dashboard.

AWS Service Instance Performance Dashboards

The *Amazon Web Services: Dashboards PowerPack* includes a dashboard for each service type. Each dashboard displays performance metrics for instances of an AWS service. The following dashboards are included:

- AWS Application ELB Performance
- AWS Classic ELB Performance
- AWS DDB Performance
- AWS EBS Performance
- AWS EC2 Performance
- AWS EMR Performance
- AWS Network ELB Performance
- AWS RDS Performance
- AWS SQS Performance
- AWS Storage Gateway Performance

Each performance dashboard includes:



- A traffic light widget that shows the status of all instances for the service.
- Four performance graphs that show applicable metrics when you select an instance from the traffic light widget.

- A time span selector that controls the amount of data shown in the performance graphs.
- An organization selector and device group selector that control which instances are shown in the traffic light widget.

Chapter

10

Run Book Actions and Automations

Overview

The following sections describe the Run Book Action and Automation policies that are included in the *Amazon Web Services PowerPack* and how to use them:

<i>About the Run Book Actions and Automations</i>	134
<i>Disabling EC2 and EBS Instances by EC2 Tag</i>	135
<i>Modifying the Parameters of the Automation Actions</i>	136
<i>Enabling the Component Device Record Created Event Policy</i>	137
<i>Enabling the Automation Policies</i>	137
<i>Preserving Automation Changes</i>	137
<i>Discovering EC2 Instances by Public or Private IP Address</i>	138
<i>Modifying the Parameters of the Automation Actions</i>	139
<i>Enabling the Component Device Record Created Event Policy</i>	141
<i>Enabling the Device Record Created Event Policy</i>	142
<i>Enabling the Automation Policies</i>	142
<i>Preserving Automation Changes</i>	143
<i>Aligning AWS Regions to the AWS Region Device Class</i>	143
<i>Vanishing Terminated or Terminating EC2 Instances</i>	144
<i>Enabling the Automation Policies</i>	145
<i>Preserving Automation Changes</i>	145

About the Run Book Actions and Automations

The Amazon Web Services PowerPack includes Run Book Action and Automation policies that can be used to:

- Automatically disable EC2 and EBS devices based on EC2 tags collected from AWS
- Automatically create and start a discovery session for the public or private IP address of an EC2 instance after a component and physical device are merged
- Automatically move an EC2 instance to a vanished state if the EC2 instance is in a terminating or terminated state
- Align AWS region device classes with the correct AWS Region

The following table describes the automation policies and what they do:

Policy Name	Result
AWS: Account Creation	SL1 creates a virtual device for an AWS account.
AWS: Disable EBS Instances by EC2 Tag	If a component device belongs to the AWS EBS Volumes device group and has an EC2 tag, SL1 disables the device.
AWS: Disable EC2 and EBS Instances by EC2 Tag	If a component device belongs to either the AWS EBS Volumes or AWS EC2 Instances device group and has an EC2 tag, SL1 disables the device.
AWS: Disable or Discover EC2 Instances	SL1 automatically discovers EC2 instances by public or private IP address. Additionally, if a component device belongs to the AWS EC2 Instances device group and has an EC2 tag, SL1 disables the device.
AWS: Discover EC2 Instances	SL1 automatically discovers EC2 instances by public or private IP address.
AWS: EKS Cluster Creation	SL1 automatically discovers EKS Clusters when an AWS EKS Cluster is configured.
AWS: Merge with EC2	If SL1 determines that the IP address of a physical device matches a custom attribute added to an EC2 Instance component device, SL1 merges the devices.
AWS: Organization Creation	SL1 creates a virtual device for an AWS organization.
AWS: RDS DB Instance Device Class Alignment	SL1 aligns the correct RDS device class the RDS Instance.
AWS: Region Device Class Alignment	If a Region is aligned to an incorrect Region device class, SL1 will align the Region to the correct device class.

Policy Name	Result
AWS: Vanish Terminated EC2 Instances	If a device belongs to the AWS EC2 Instances device group and is in a terminated or terminating state, SL1 un-merges the EC2 Instance and physical device (if applicable), clears the device's associated events, and then moves the device to a vanished state.

NOTE: The automation policies in the Amazon Web Services PowerPack are disabled by default. To use these automations, you must enable the automation policies and optionally modify the parameters in the automation actions.

NOTE: To use the automation policies in the Amazon Web Services PowerPack, the AWS EBS Volumes and AWS EC2 Instances device groups must already be created and populated.

Disabling EC2 and EBS Instances by EC2 Tag

The automation described in this section disables EC2 and EBS devices based on EC2 tags. This can be set up in the "AWS: Disable Instance by Tag" Run Book Automation, so if an EBS or EC2 instance has the tag(s) you specify, SL1 will disable the device.

The automation for disabling EC2 and EBS instances includes two automation actions that are executed in the following order:

- **AWS: Get EC2 Instance Configuration.** This action requests information from the AWS API about the EC2 instance that triggered the automation action or the EC2 instance associated with the EBS instance that triggered the automation action. Information about the EC2 instance associated with an EBS instance is returned only if one EC2 instance is associated with the EBS instance.
- **AWS: Disable Instance By Tag.** This action compares the information collected by the **AWS: Get EC2 Instance Configuration** automation action with a pre-defined list of key/value pairs. If an AWS tag matches a key/value pair, the triggering device is disabled.

The Amazon Web Services PowerPack includes three automation policies that trigger these actions:

- **AWS: Disable EC2 and EBS Instances by EC2 Tag.** If enabled, this automation policy can trigger for any device with which the "AWS: EC2 Instance Configuration" or the "AWS: EBS Instance Configuration" Dynamic Applications are aligned (the members of the AWS EC2 Instances and AWS EBS Volumes device groups). The automation policy triggers when the "Component Device Record Created" event is active on the matching devices, immediately after the devices are discovered in the system. Enable this automation policy if you want to disable EC2 and EBS instances by EC2 tag, but do not want to enable automated discovery of EC2 instances by public or private IP address.

- **AWS: Disable or Discover EC2 Instances.** If enabled, this automation policy can trigger for any device with which the "AWS: EC2 Instance Configuration" Dynamic Application is aligned (the members of the AWS EC2 Instances). The automation policy triggers when the "Component Device Record Created" event is active on the matching devices, immediately after the devices are discovered in the system. Enable this automation policy if you want to disable EC2 instances by EC2 tag *and* want to enable automated discovery of EC2 instances by public or private IP address. This automation policy is configured to run both processes in the correct order for EC2 instances. If you enable this automation policy and want to automatically disable associated EBS instances, you must also enable the **AWS: Disable EBS Instances by EC2 Tag** automation policy.
- **AWS: Disable EBS Instances by EC2 Tag.** If enabled, this automation policy can trigger for any device with which the "AWS: EC2 Instance Configuration" Dynamic Application is aligned (the members of the AWS EC2 Instances). The automation policy triggers when the "Component Device Record Created" event is active on the matching devices, immediately after the devices are discovered in the system. Enable this automation policy if you want to disable EC2 instances by EC2 tag, want to enable automated discovery of EC2 instances by public or private IP address, and want to disable EBS instances by EC2 tag.

To use this automation, you must:

- [Modify the parameters of the automation actions \(optional\)](#)
- [Enable the Component Device Record Created event policy](#)
- [Enable the automation policies](#)
- [Configure your system to preserve these changes](#)

Modifying the Parameters of the Automation Actions

The snippet for the **AWS: Disable Instance by Tag** automation action includes the pre-defined list of key/value pairs with which the tags collected from the AWS API are compared. You must modify this list to include the key/value pairs that you want to use to disable EC2 instances.

To modify the parameters for the **AWS: Disable Instance by Tag** automation action:

1. Go to the **Action Policy Manager** page (Registry > Run Book > Actions).
2. Click the wrench icon () for the **AWS: Disable Instance By Tag** automation action.
3. In the **Snippet Code** field, locate and edit the following line:

```
DISABLE_TAGS = [('ExampleKey', 'ExampleValue')]
```

The line must be in the following format, with each key and each value inside single-quotes and each key/value pair comma-separated inside parentheses, with commas separating each key/value pair.

```
DISABLE_TAGS = [('Key', 'Value'), ('Key', 'Value'), ..., ('Key', 'Value')]
```

For example, suppose you want to disable an EC2 instance where the "Environment" key is either "dev" or "test" or the "Owner" key is "Sales". You would update the line so it looks like this:

```
DISABLE_TAGS = [('Environment', 'dev'), ('Environment', 'test'), ('Owner', 'Sales')]
```

4. Click the **[Save]** button.

Enabling the Component Device Record Created Event Policy

To enable the "Component Device Record Created" event policy:

1. Go to the **Event Policies** page (Events > Event Policies).
2. Click the Actions menu () for the "Component Device Record Created" event policy and select *Edit*.
3. In the **Event Policy Editor** page, click on the **Enable Event Policy** toggle to enable the event policy.
4. Click **[Save]**.

To enable the "Component Device Record Created" event policy in the SL1 classic user interface:

1. Go to the **Event Policy Manager** page (Registry > Events > Event Manager).
2. Click the wrench icon () for the "Component Device Record Created" event policy.
3. In the **Operational State** field, select *Enabled*.
4. Click **[Save]**.

To prevent this change from being overwritten when the PowerPacks installed on the system are updated, you can enable the **Selective PowerPack Field Protection** option. To enable this option:

1. Go to the **Behavior Settings** page (System > Settings > Behavior).
2. Check the **Enable Selective PowerPack Field Protection** checkbox.
3. Click **[Save]**.

Enabling the Automation Policies

To enable one or more automation policies in the *Amazon Web Services* PowerPack:

1. Go to the **Automation Policy Manager** page (Registry > Run Book > Automation).
2. Click the wrench icon () for the automation policy you want to enable.
3. In the **Policy State** field, select *Enabled*.
4. Click **[Save]**.

Preserving Automation Changes

If you have modified automation actions and policies that are included in the *Amazon Web Services* PowerPack, those changes will be overwritten when the PowerPack is updated in your system. If you have modified automation actions and policies that are included in the PowerPack, you can:

- Re-implement those changes after each update of the *Amazon Web Services* PowerPack.
- Remove the content from the PowerPack on your system before you update it. When the *Amazon Web Services* PowerPack is updated in your system, updated versions of this content will not be installed on your system and your local changes will be preserved.

To remove automation actions or automation policies content from the *Amazon Web Services PowerPack* on your system:

1. Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
2. Click the wrench icon () for the *Amazon Web Services PowerPack*. The **Editing PowerPack** page appears.
3. In the left NavBar of the **Editing PowerPack** page, select the type of content you want to remove:
 - To remove an automation action, click **Run Book Actions**. The **Embedded Run Book Actions** and **Available Run Book Actions** panes appear.
 - To remove an automation policy, click **Run Book Policies**. The **Embedded Run Book Policies** and **Available Run Book Policies** panes appear.
4. In the upper pane, click the bomb icon () for each automation action or automation policy that you want to remove from the *Amazon Web Services PowerPack* on your system.

Discovering EC2 Instances by Public or Private IP Address

The automation in this section automatically creates and starts a discovery session for the public or private IP address of an EC2 instance after a component and physical device are merged. If SL1 determines that the IP address of a physical device matches a custom attribute added to an EC2 instance component device, SL1 merges the devices.

The automation for discovering EC2 instances by public or private IP addresses includes three automation actions that are executed in the following order:

- **AWS: Get EC2 Instance Configuration**. This action requests information from the AWS API about the EC2 instance that triggered the automation action.
- **AWS: Discover from EC2 IP**. This action uses the IP address and port information in the response from the AWS API to create and run a discovery session. This action also adds a custom attribute to the EC2 component device record that can be used to match a newly discovered device to the EC2 instance.
- **AWS: Merge Physical with Component**. This action matches the IP address of a physical device with the custom attribute added to EC2 component devices by the **AWS: Discover from EC2 IP** automation action. If a match is found, the matching EC2 component device is merged with the physical device.

The *Amazon Web Services PowerPack* includes three automation policies that trigger these actions:

- **AWS: Discover EC2 Instances**. If enabled, this automation policy can trigger for any device with which the "AWS: EC2 Instance Configuration" Dynamic Application is aligned (the members of the AWS EC2 Instances). The automation policy triggers when the "Component Device Record Created" event is active on the matching devices, immediately after the devices are discovered in the system. Enable this automation policy if you want to enable automated discovery of EC2 instances by public or private IP address but do not want disable EC2 and EBS instances by EC2 tag.

- **AWS: Disable or Discover EC2 Instances.** If enabled, this automation policy can trigger for any device with which the "AWS: EC2 Instance Configuration" Dynamic Application is aligned (the members of the AWS EC2 Instances). The automation policy triggers when the "Component Device Record Created" event is active on the matching devices, immediately after the devices are discovered in the system. Enable this automation policy if you want to disable EC2 instances by EC2 tag **and** want to enable automated discovery of EC2 instances by public or private IP address. This automation policy is configured to run both in the correct order for EC2 instances.
- **AWS: Merge with EC2.** If enabled, this automation policy can trigger for any device. The automation policy triggers when the "Device Record Created" event is active on the matching devices, immediately after the devices are discovered in the system. Enable this automation policy if you want to enable automated discovery of EC2 instances by public or private IP address.

To use this automation, you must:

- [Modify the parameters of the automation actions \(optional\)](#)
- [Enable the Component Device Record Created event policy](#)
- [Enable the Device Record Created event policy](#)
- [Enable the automation policies](#)
- [Configure your system to preserve these changes](#)

Modifying the Parameters of the Automation Actions

The snippet for the **AWS: Discover from EC2 IP** automation action includes parameters that define how the automation action creates discovery sessions. You can edit the following lines in the **Snippet Code** field of the **AWS: Discover from EC2 IP** automation action to change these parameters:

- `EC2_IP_ATTRIBUTE = 'PrivateIpAddress'`

The attribute returned by the AWS API for EC2 instances that contains the IP address to use in the discovery session. By default, the private IP address is used. To use the public IP address of the EC2 instance, change this line to:

```
EC2_IP_ATTRIBUTE = 'PublicIpAddress'
```

- `EXTRA_SCAN_PORTS = ["21", "22", "23", "25", "80", "443", "5985", "5986"]`

The list of TCP ports used in the discovery session includes any TCP ports that are specified explicitly in the security group associated with the EC2 instance, plus any TCP ports included in the `EXTRA_SCAN_PORTS` parameter. You can add or remove ports from this default list. For example, if you wanted to remove TCP port 21 from this list and add TCP port 53, you would change this line to:

```
EXTRA_SCAN_PORTS = ["22", "23", "25", "53", "80", "443", "5985", "5986"]
```

NOTE: The `EXTRA_SCAN_PORTS` parameter must be populated if there are no rules for specific ports in the security group associated with the EC2 instance.

- `AUTO_INCLUDE_CREDS = True`

If the `AUTO_INCLUDE_CREDS` parameter is "True", the automation will automatically add credentials to the discovery session. A credential will be added automatically if it meets one of the following requirements:

- The credential is an SNMP credential, the Security Group associated with the EC2 instance includes a rule that allows access to UDP port 161, and the credential is explicitly aligned within the organization of the EC2 instance.
- The credential is an SNMP credential, the Security Group associated with the EC2 instance includes a rule that allows access to UDP port 161, the credential is associated with all organizations in the system, and the `INCLUDE_ALL_ORG_CREDS` parameter is "True".
- The credential is not an SNMP credential or an LDAP/AD credential, the TCP port used by the credential is included in the list of TCP ports for the discovery session (the credential is specified explicitly in the security group associated with the EC2 instance or is included in the `EXTRA_SCAN_PORTS` parameter), and the credential is explicitly aligned with in the organization of the EC2 instance.
- The credential is not an SNMP credential or an LDAP/AD credential, the TCP port used by the credential is included in the list of TCP ports for the discovery session (the credential is specified explicitly in the security group associated with the EC2 instance or is included in the `EXTRA_SCAN_PORTS` parameter), and the `INCLUDE_ALL_ORG_CREDS` parameter is "True".

To disable the automatic alignment of credentials to the discovery session, change this line to:

```
AUTO_INCLUDE_CREDS = False
```

- `INCLUDE_ALL_ORG_CREDS = True`

If `INCLUDE_ALL_ORG_CREDS` is "True" and the `AUTO_INCLUDE_CREDS` parameter is "True", credentials that are aligned with all organizations (credentials that do not have an explicit organization alignment) are automatically included in the discovery session when that credential meets the other requirements for being automatically included in the discovery session.

- `EXTRA_CREDS = ""`

In addition to the credentials that are automatically included in the discovery sessions based on open ports, you can optionally specify a string of comma-separated credential IDs for credentials that will be included in every discovery session created by this automation. For example, if you wanted to include credentials with IDs 10 and 13 in every discovery session created by this automation, you would change this line to:

```
EXTRA_CREDS = "10,13"
```

- `DISCOVER_NON_SNMP = "1"`

If `DISCOVER_NON_SNMP` is set to "1", discovery sessions created by this automation will be configured to discover non-SNMP devices. If you want the discovery sessions created by this automation to discover only SNMP devices, change this line to:

```
DISCOVER_NON_SNMP = "0"
```

- `TEMPLATE_NAME = ""`

If you specify a device template name in the `TEMPLATE_NAME` parameter, that device template will be automatically aligned with all discovery sessions created by this automation. For example, if you wanted to align a device template called "Standard Device Template" to every discovery session created by this automation, you would change this line to:

```
TEMPLATE_NAME = "Standard Device Template"
```

To modify the parameters for the **AWS: Discover from EC2 IP** automation action, perform the following steps:

1. Go to the **Action Policy Manager** page (Registry > Run Book > Actions).
2. Click the wrench icon () for the **AWS: Discover from EC2 IP** automation action.
3. In the **Snippet Code** field, locate and edit the line(s) for the parameter(s) you want to change:
4. Click the **[Save]** button.

If you modified the `EC2_IP_ATTRIBUTE` parameter in the **AWS: Discover from EC2 IP** automation action, you must perform the following steps to update the **AWS: Merge Physical with Component** automation action:

To modify the parameters for the **AWS: Discover from EC2 IP** automation action, perform the following steps:

1. Go to the **Action Policy Manager** page (Registry > Run Book > Actions).
2. Click the wrench icon () for the **AWS: Discover from EC2 IP** automation action.
3. In the **Snippet Code** field, locate and edit the following line:

```
IP_ATTRIBUTE = 'c-EC2_PrivateIpAddress'
```

If you changed the `EC2_IP_ATTRIBUTE` parameter in the **AWS: Discover from EC2 IP** automation action to 'PublicIpAddress', change this line to:

```
IP_ATTRIBUTE = 'c-EC2_PublicIpAddress'
```

4. Click the **[Save]** button.

Enabling the Component Device Record Created Event Policy

To enable the "Component Device Record Created" event policy:

1. Go to the **Event Policies** page (Events > Event Policies).
2. Click the Actions menu () for the "Component Device Record Created" event policy and select *Edit*.
3. In the **Event Policy Editor** page, click on the **Enable Event Policy** toggle to enable the event policy.
4. Click **[Save]**.

To enable the "Component Device Record Created" event policy in the SL1 classic user interface:

1. Go to the **Event Policy Manager** page (Registry > Events > Event Manager).
2. Click the wrench icon () for the "Component Device Record Created" event policy.

3. In the **Operational State** field, select *Enabled*.
4. Click **[Save]**.

To prevent this change from being overwritten when the PowerPacks installed on the system are updated, you can enable the **Selective PowerPack Field Protection** option. To enable this option:

1. Go to the **Behavior Settings** page (System > Settings > Behavior).
2. Check the **Enable Selective PowerPack Field Protection** checkbox.
3. Click **[Save]**.

Enabling the Device Record Created Event Policy

To enable the "Device Record Created" event policy:

1. Go to the **Event Policies** page (Events > Event Policies).
2. Click the Actions menu () for the "Device Record Created" event policy and select *Edit*.
3. In the **Event Policy Editor** page, click on the **Enable Event Policy** toggle to enable the event policy.
4. Click **[Save]**.

To enable the "Device Record Created" event policy in the SL1 classic user interface:

1. Go to the **Event Policy Manager** page (Registry > Events > Event Manager).
2. Click the wrench icon () for the "Device Record Created" event policy.
3. In the **Operational State** field, select *Enabled*.
4. Click **[Save]**.

To prevent this change from being overwritten when the PowerPacks installed on the system are updated, you can enable the **Selective PowerPack Field Protection** option. To enable this option:

1. Go to the **Behavior Settings** page (System > Settings > Behavior).
2. Check the **Enable Selective PowerPack Field Protection** checkbox.
3. Click **[Save]**.

Enabling the Automation Policies

To enable one or more automation policies in the Amazon Web Services PowerPack:

1. Go to the **Automation Policy Manager** page (Registry > Run Book > Automation).
2. Click the wrench icon () for the automation policy you want to enable.
3. In the **Policy State** field, select *Enabled*.
4. Click **[Save]**.

Preserving Automation Changes

If you have modified automation actions and policies that are included in the *Amazon Web Services PowerPack*, those changes will be overwritten when the PowerPack is updated in your system. If you have modified automation actions and policies that are included in the PowerPack, you can:

- Re-implement those changes after each update of the *Amazon Web Services PowerPack*.
- Remove the content from the PowerPack on your system before you update it. When the *Amazon Web Services PowerPack* is updated in your system, updated versions of this content will not be installed on your system and your local changes will be preserved.

To remove automation actions or automation policies content from the *Amazon Web Services PowerPack* on your system:

1. Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
2. Click the wrench icon () for the *Amazon Web Services PowerPack*. The **Editing PowerPack** page appears.
3. In the left NavBar of the **Editing PowerPack** page, select the type of content you want to remove:
 - To remove an automation action, click **Run Book Actions**. The **Embedded Run Book Actions** and **Available Run Book Actions** panes appear.
 - To remove an automation policy, click **Run Book Policies**. The **Embedded Run Book Policies** and **Available Run Book Policies** panes appear.
4. In the upper pane, click the bomb icon () for each automation action or automation policy that you want to remove from the *Amazon Web Services PowerPack* on your system.

Aligning AWS Regions to the AWS Region Device Class

The automation for aligning an AWS Region to the correct AWS Region device class includes one automation action:

- **AWS: Region Device Class Alignment**. This action updates the AWS device class to the correct AWS Region.

NOTE: Device classes for AWS Regions are updated in the second cycle of the "AWS: Region Device Class Discovery" Dynamic Application. Regions will be updated after 24 hours.

The *Amazon Web Services PowerPack* includes an automation policy that triggers this action:

- **AWS: Region Device Class Alignment.** If enabled, this automation policy can trigger for any device with which the "AWS: Region Device Class Discovery" Dynamic Application is aligned. The automation policy triggers when the "AWS: Device Class Change" event is active on the matching devices, and the automation policy will repeat every 10 minutes until that event is no longer active.

Vanishing Terminated or Terminating EC2 Instances

The automation in this section automatically moves an EC2 instance to a vanished state if the EC2 instance is in a terminating or terminated state. SL1 unmerges the EC2 instance and physical device, clearing the associated events, and moves the devices to a vanished state.

The automation for vanishing terminated EC2 instances includes one automation action:

- **AWS: Vanish Terminated EC2 Instances.** If an EC2 instance has been terminated in Amazon, its corresponding device in SL1 becomes unavailable. This action then requests information from the AWS API about the EC2 instance that triggered the automation action. If the response from the AWS API indicates that the EC2 instance that triggered the automation action is in a terminated or terminating state, the action performs the following steps:
 - If the automation triggers for a physical device that is merged with an EC2 instance, the devices are un-merged.
 - If the automation triggers for a physical device that is merged with an EC2 instance, after being un-merged the physical device is moved to a virtual collector group.
 - If the automation triggers for a physical device that is merged with an EC2 instance, after being unmerged, all events associated with the physical device are cleared.
 - All events associated with the component device are cleared.
 - The component device is vanished.

NOTE: If an EC2 instance is stopped in AWS rather than terminated, then the "AWS Vanish Terminated EC2 Instances" action is not triggered.

The *Amazon Web Services PowerPack* includes an automation policy that triggers this action:

- **AWS: Vanish Terminated EC2 Instances.** If enabled, this automation policy can trigger for any device with which the "AWS: EC2 Instance Configuration" Dynamic Application is aligned (the members of the AWS EC2 Instances). The automation policy triggers when the "Availability Check Failed" event is active on the matching devices, and the automation policy will repeat every 10 minutes until that event is no longer active.

To use this automation, you must:

- [Enable the AWS: Vanish Terminated EC2 Instances automation policy](#)
- [Configure your system to preserve this change](#)

Enabling the Automation Policies

To enable one or more automation policies in the *Amazon Web Services* PowerPack:

1. Go to the **Automation Policy Manager** page (Registry > Run Book > Automation).
2. Click the wrench icon () for the automation policy you want to enable.
3. In the **Policy State** field, select *Enabled*.
4. Click **[Save]**.

Preserving Automation Changes

If you have modified automation actions and policies that are included in the *Amazon Web Services* PowerPack, those changes will be overwritten when the PowerPack is updated in your system. If you have modified automation actions and policies that are included in the PowerPack, you can:

- Re-implement those changes after each update of the *Amazon Web Services* PowerPack.
- Remove the content from the PowerPack on your system before you update it. When the *Amazon Web Services* PowerPack is updated in your system, updated versions of this content will not be installed on your system and your local changes will be preserved.

To remove automation actions or automation policies content from the *Amazon Web Services* PowerPack on your system:

1. Go to the **PowerPack Manager** page (System > Manage > PowerPacks).
2. Click the wrench icon () for the *Amazon Web Services* PowerPack. The **Editing PowerPack** page appears.
3. In the left NavBar of the **Editing PowerPack** page, select the type of content you want to remove:
 - To remove an automation action, click **Run Book Actions**. The **Embedded Run Book Actions** and **Available Run Book Actions** panes appear.
 - To remove an automation policy, click **Run Book Policies**. The **Embedded Run Book Policies** and **Available Run Book Policies** panes appear.
4. In the upper pane, click the bomb icon () for each automation action or automation policy that you want to remove from the *Amazon Web Services* PowerPack on your system.

11

Key Metrics Collected by the PowerPack

Overview

This section lists the key metrics for AWS services that the Amazon Web Services PowerPack collects by Dynamic Application.

<i>AWS API Gateway Service</i>	147
<i>AWS Application ELB Service</i>	148
<i>AWS Auto Scale Service</i>	150
<i>AWS CloudFront Service</i>	152
<i>AWS CloudTrail Service</i>	157
<i>AWS CloudWatch Service</i>	158
<i>AWS DDB Service</i>	159
<i>AWS Direct Connect Service</i>	160
<i>AWS DynamoDB Service</i>	165
<i>AWS EBS Service</i>	166
<i>AWS EC2 Service</i>	168
<i>AWS ECS Service</i>	175
<i>AWS EFS Service</i>	182
<i>AWS EKS Service</i>	184
<i>AWS Elastic Beanstalk Service</i>	186
<i>AWS ElastiCache Service</i>	190
<i>AWS ELB Service</i>	196
<i>AWS EMR Service</i>	200
<i>AWS Glacier Service</i>	204
<i>AWS IoT Service</i>	205

<i>AWS KMS Service</i>	206
<i>AWS Lambda Service</i>	207
<i>AWS LightSail Service</i>	213
<i>AWS Network ELB Service</i>	216
<i>AWS OpsWorks Service</i>	217
<i>AWS RDS Service</i>	219
<i>AWS Redshift Service</i>	230
<i>AWS Route 53 Service</i>	233
<i>AWS S3 Service</i>	235
<i>AWS SES Service</i>	237
<i>AWS Shield Standard Service</i>	238
<i>AWS SNS Service</i>	239
<i>AWS SQS Service</i>	241
<i>AWS Storage Gateway Service</i>	243
<i>AWS STS Service</i>	245
<i>AWS Transit Gateway Service</i>	245
<i>AWS VPC Service</i>	247
<i>AWS WAF Global Service</i>	254
<i>AWS Workspaces Service</i>	256

AWS API Gateway Service

AWS: API Gateway Service Configuration	
Object Name	Object Description
ID	The identifier of a Usage Plan resource.
Name	The name of a usage plan.
Rate	The API request rate limit.
AWS API Gateway Service/Network Load Balancer	The ELB associated with the API Gateway Service.
Burst	The API request burst limit.
ID	The identifier of the VPC Link.
Name	The name of the VPC link.

Network Load Balancer	The name of ELB associated with the API Gateway.
Quota	The maximum number of requests that can be made in a given time period.
Stages	The associated API stages of a usage plan.
Status	The status of the VPC link.
Target ARNs	The ARNs of network load balancers of the VPC targeted by the VPC link.

AWS: API Gateway Service Health

Object Name	Object Description
Date	The timestamp of this health status update string.
Health	A text description of this AWS API Gateway Service health status.

AWS Application ELB Service

AWS: Application ELB Instance Configuration

Object Name	Object Description
AWS Application ELB/Availability Zones	The availability zones for the load balancer.
AWS Application ELB/Security Groups	The unique identifiers of the security groups for the load balancer.
AWS Application ELB/Target Groups	The unique identifiers of the target groups for the application load balancer.
AWS Application ELB/VPC	The unique identifier of the VPC for the load balancer.
DNS Name	The public DNS name of the application load balancer.
Key	The key of the tag, that belongs to the application load balancer.
Listener ARN	The Amazon Resource Name (ARN) of the listener.
Listener Port	The port on which the load balancer is listening.
Listener Protocol	The protocol for connections from clients to the load balancer.

Load Balancer ARN	The Amazon Resource Name (ARN) of the load balancer.
Load Balancer Name	The name of the load balancer.
Load Balancer State	The state of the load balancer.
Scheme	The DNS name of an Internet-facing load balancer is publicly resolvable to the public IP addresses of the nodes.
Type	The type of load balancer.
Value	The value of the tag, that belongs to the application load balancer.

AWS: Application ELB Instance Performance

Object Name	Object Description
Active Connection Count	The total number of concurrent TCP connections active from clients to the load balancer and from the load balancer to targets.
Client TLS Negotiation Error Count	The number of TLS connections initiated by the client that did not establish a session with the load balancer. Possible causes include a mismatch of ciphers or protocols.
HTTPCode_ELB_4XX_Count	The number of HTTP 4XX client error codes that originate from the load balancer. Client errors are generated when requests are malformed or incomplete. These requests have not been received by the target. This count does not include any response codes generated by the targets.
HTTPCode_ELB_5XX_Count	The number of HTTP 5XX server error codes that originate from the load balancer. This count does not include any response codes generated by the targets.
HTTPCode_Target_2XX_Count	The number of HTTP response codes generated by the targets. This does not include any response codes generated by the load balancer.
HTTPCode_Target_3XX_Count	The number of HTTP response codes generated by the targets. This does not include any response codes generated by the load balancer.
HTTPCode_Target_4XX_Count	The number of HTTP response codes generated by the targets. This does not include any response codes generated by the load balancer.

HTTPCode_Target_5XX_Count	The number of HTTP response codes generated by the targets. This does not include any response codes generated by the load balancer.
Rejection Connection Count	The number of connections that were rejected because the load balancer had reached its maximum number of connections.
Request Count	The number of requests received by the load balancer. This includes requests over IPv4 and IPv6.
Target Connection Error Count	The number of connections that were not successfully established between the load balancer and target.
Target Response Time	The time elapsed, in seconds, after the request leaves the load balancer until a response from the target is received. This is equivalent to the target_processing_time field in the access logs.

AWS Auto Scale Service

AWS: Auto Scale Group Instance Configuration	
Object Name	Object Description
.Metric Name	One or more metrics. If you omit this parameter, all metrics are enabled. GroupMinSize GroupMaxSize GroupDesiredCapacity GroupInServiceInstances GroupPendingInstances GroupStandbyInstances GroupTerminatingInstances GroupTotalInstances Note that the GroupStandbyInstances metric is not enabled by default. You must explicitly request this metric.
ARN	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
AWS Auto Scale Group/EC2	The unique identifier for the AutoScale-EC2 relationship.
AWS Auto Scale Group/Launch Configuration	The identifier for this group's LaunchConfig.
Creation Date	The datetime this group was created.
Default Cooldown	The number of seconds allowed for the cooldown period.
Desired Capacity	The desired size of the group.
Health Check Grace Period	The polling interval between health checks.

Health Check Type	The health check status type for the group. This is usually an EC2.
Id	The unique identifier of the Auto Scale Group.
Launch Configuration ARN	The identifier for this group's LaunchConfig.
Load Balancers	A list of load balancers for the group.
Max	The maximum size of the group.
Metric Granularity	The granularity to associate with the metrics to collect. The only valid value is 1Minute.
Min	The minimum size of the group.
Name	The name of the Auto Scale group.
Placement Group	The name of the placement group, if any, for this group.
Subnets	The subnet ID of the VPC where the Group resides.
Suspended Processes	A list of suspended process names.
Termination Policies	Pre-launch environment variables
Zones	All deployable zones in this region.

AWS: Auto Scale Group Performance

Object Name	Object Description
Group Desired Capacity	The number of instances that the Auto Scaling group attempts to maintain.
Group In Service Instances	The number of instances that are running as part of the Auto Scaling group. This metric does not include instances that are pending or terminating.
Group Max Size	The maximum size of the Auto Scaling group.
Group Min Size	The minimum size of the Auto Scaling group.
Group Pending Instances	The number of instances that are pending. A pending instance is not yet in service. This metric does not include instances that are in service or terminating.
Group Standby Instances	The number of instances that are in a Standby state. Instances in this state are still running but are not actively in service.

Group Terminating Instances	The number of instances that are in the process of terminating. This metric does not include instances that are in service or pending.
Group Total Instances	The total number of instances in the Auto Scaling group. This metric identifies the number of instances that are in service, pending, and terminating.

AWS: Auto Scale Launch Config Instance Configuration

Object Name	Object Description
ARN	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Block Devices	Attached EBS filesystems at boot.
Created	The datetime this configuration was created.
EBS Optimized	If this configuration is EBS Optimized.
IAM Instance Profile	The IAM role name associated with this configuration.
Id	The unique identifier of the launch configuration.
Instance Type	The EC2 type of the instances in this configuration.
Kernel ID	The unique identifier for the kernel of this configuration.
Key Name	The key-pair associated with this configuration.
Monitoring	Whether or not detailed monitoring is set for this configuration.
Name	The name of the launch configuration.
RAM Disk Id	The unique ID, if any, for the RAM disk of this configuration.
Security Groups	A list of security groups for this launch configuration.
Spot Price	The spot price in USD/Hour for this configuration.

AWS CloudFront Service

AWS: CloudFront Behavior Configuration

Object Name	Object Description
Forwarded Query Strings	The Forwarded Query Strings of the Behavior.
Id	The unique identifier of the Behavior.
Name	The name of the Behavior.
Origin	The name of the Origin associated with this Behavior.
Path Pattern	A path pattern (for example, images/*.jpg) specifies which requests you want this cache behavior to apply to. When CloudFront receives an end-user request, the requested path is compared with path patterns in the order in which cache behaviors are listed in the distribution.
Trusted Signers	Trusted signers are the AWS accounts that can create signed URLs and signed cookies for a distribution. By default, no account, not even the account that created the distribution, is allowed to create signed URLs or signed cookies. To specify the AWS accounts that you want to use as trusted signers, add the accounts to your distribution (Web distributions and RTMP distributions)
Viewer Protocol Policy	Configure the Viewer Protocol Policy for some or all of your CloudFront cache behaviors either to redirect HTTP requests to HTTPS requests or to require that viewers use only the HTTPS protocol to access your objects in the CloudFront cache.

AWS: CloudFront Error Page Configuration

Object Name	Object Description
Error Caching Minimum TTL	The smallest TTL (in seconds) allowed for this cached element.
HTTP Error Code	The HTTP status code associated with this Error Page.
HTTP Response Code	The HTTP status code used in the response header of the Error Page.
Response Page Path	The URL path associated with the response resource of the Error Page.

AWS: CloudFront Invalidation Configuration

Object Name	Object Description
-------------	--------------------

Date	The date when an invalidation was created.
Id	The unique identifier of the Invalidation.
Object Paths	The object paths of the Invalidation.
Status	The status of the invalidation.

AWS: CloudFront Origin Configuration

Object Name	Object Description
Access Identity	The access identity hash value.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
HTTP Port	The HTTP CloudFront uses only HTTP to access the origin.
HTTPS Port	The HTTPS CloudFront uses only HTTPS to access the origin.
Id	The unique identifier of the Origin.
Name	The name of the Origin.
Origin Protocol Policy	The cache behaviors are routing requests to the origins for which you have configured an Origin Protocol Policy of HTTPS Only or Match Viewer, if applicable.
Type	The type of origin storage.

AWS: CloudFront Restriction Configuration

Object Name	Object Description
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Id	The unique identifier of the Restriction.
Name	The name of the Restriction.
Status	Whether or not this restriction is enabled.
Type	Whether the Restriction is a white-list or a black-list.

AWS: CloudFront RTMP Distribution Configuration

Object Name	Object Description
CNames	Alternative DNS entry names and sub-domains.
Comment	A comment about the specific distribution.
Delivery Method	The transfer protocol of the distribution.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Id	The unique identifier of the distribution.
Last Modified	The most recent date and time when the configuration of this distribution was modified.
Log Bucket	The S3 bucket, if any, used to store the access logs of the distribution.
Log Prefix	The log prefix, if any, used for the S3 path of the Log Bucket. Cloudfrontlog is the default value.
Name	The name of the distribution.
Origin	The fully-qualified S3 bucket name for the distribution.
Price Class	The price tiers, if any, for restrictions on the distribution.
State	Whether or not the distribution is enabled for deployment.
Status	Whether or not the distribution is deployed.
Trusted Signers	A list of AWS Account IDs.

AWS: CloudFront Service Health

Object Name	Object Description
Date	A timestamp when the health status of this service was originally written.
Health	A text description of the health status of this AWS service.

AWS: CloudFront Web Distribution Configuration

Object Name	Object Description
-------------	--------------------

AWS CloudFront Web Distribution/WAF Web ACL	The Web ACL associated with the CloudFront.
CNames	Alternative DNS entry names and sub-domains.
Comment	A comment about the specific distribution.
Delivery Method	The transfer protocol of the distribution.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Id	The unique identifier of the distribution.
Last Modified	The most recent date and time when the configuration of this distribution was modified.
Log Bucket	The S3 bucket, if any, used to store the access logs of the distribution.
Log Prefix	The log prefix, if any, used for the S3 path of the Log Bucket. Cloudfrontlog is the default value.
Name	The name of the distribution.
Price Class	The price tiers, if any, for restrictions on the distribution.
State	Whether or not the distribution is enabled for deployment.
Status	Whether or not the distribution is deployed.

AWS: CloudFront Web Distribution Performance

Object Name	Object Description
4xx Error Rate	The percentage of all requests for which the HTTP status code is 4xx.
5xx Error Rate	The percentage of all requests for which the HTTP status code is 5xx.
Bytes Downloaded	The number of bytes downloaded by viewers for GET, HEAD, and OPTIONS requests.
Bytes Uploaded	The number of bytes uploaded to your origin with CloudFront using POST and PUT requests.
Requests	The number of requests for all HTTP methods and for both HTTP and HTTPS requests.
Total Error Rate	The percentage of all requests for which the HTTP status code is 4xx or 5xx.

AWS CloudTrail Service

AWS: CloudTrail Instance Configuration	
Object Name	Object Description
AWS CloudTrail/S3	The identifier for the related S3.
AWS CloudTrail/SNS	The unique identifier for the CloudTrail-SNS relationship.
AWS: S3 Relationship	The group label for the CloudTrail-S3 relationship.
AWS: SNS Relationship	The group label for the CloudTrail-SNS relationship.
Bucket Name	The name of the S3 bucket which stores the trail.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Latest Delivery Attempt Succeeded	The datetime of the most recent successful SNS delivery.
Latest Delivery Attempt Time	The datetime of the most recent attempt to deliver cloud trail logs from SNS.
Latest Delivery Error	The error type of the most recent, if any, SNS delivery error.
Latest Notification Attempt Succeeded	The most recent datetime when an SNS delivery succeeded.
Latest Notification Attempt Time	The datetime of the most recent SNS delivery attempt whether or not it succeeded.
Latest Notification Error	The most recent notification error.
Log Global Service Events	Whether or not service status events are logged.
Logging	Whether or not the trail is actively logging.
Logging Prefix	The prefix string, if any, attached to the S3 path where trails are stored.
Name	The name of the trail.
SNS Publishing	Whether or not trail logging is published to SNS.
SNS Topic Name	The SNS topic which publishes trail updates.
Time Logging Started	The datetime when logging most recently started.
Time Logging Stopped	The most recent datetime when logging stopped.

AWS CloudWatch Service

AWS: CloudWatch Alarms Performance	
Object Name	Object Description
CloudWatch Action (Failed) Alarms	The total number of Action alarms with actionState Failure in the latest polling interval.
CloudWatch Action Alarms	The total number of Action alarms received in the latest polling interval.
CloudWatch Alarms	The total number of alarms received in the latest polling interval.
CloudWatch Alarms Collection Success	Indicates the success (1) or failure (0) of the CloudWatch alarms history collection snippet. Failure (0) occurs if the snippet encounters an unexpected error. NOTE: This collection object is especially important for determining which CloudWatch alarms are to be monitored. 1. To collect all CloudWatch alarm types (ConfigurationUpdate, StateUpdate, Action), select the snippet cloudwatch_alarms_performance. 2. To collect only CloudWatch StateUpdate alarm types, select the snippet cloudwatch_alarms_performance_StateUpdate_only.
CloudWatch ConfigurationUpdate Alarms	The total number of ConfigurationUpdate alarms received in the latest polling interval.
CloudWatch State Alarms	The total number of StateUpdate alarms received in the latest polling interval.
CloudWatch StateUpdate (ALARM) Alarms	The total number of Alarm StateUpdate alarms in the latest polling interval.
CloudWatch StateUpdate (INSUFFICIENT_DATA) Alarms	The total number of INSUFFICIENT_DATA StateUpdate alarms in the latest polling interval.
CloudWatch StateUpdate (OK) Alarms	The total number of OK StateUpdate alarms in the latest polling interval.
CloudWatch Total Alarms	A running total of all AWS CloudWatch alarms seen since start of collection.

AWS: CloudWatch Service Health

Object Name	Object Description
Date	A timestamp when this service's health status was originally written.
Health	A text description of this AWS Service's health status.

AWS DDB Service

AWS: DDB Instance Configuration	
Object Name	Object Description
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Read Capacity	The provisioned read capacity of the table
Table Name	The name of the DDB table.
Write Capacity	The provisioned write capacity of the table.

AWS: DDB Instance Performance	
Object Name	Object Description
Avg Consumed Read Capacity Units	The average consumed read capacity of the deployment configuration for this table.
Avg Consumed Write Capacity Units	The average consumed write capacity for the table.
Avg Returned Item Count	The average returned item count for the table.
Avg Successful Request Latency	The average latency for successful table operations.
Consumed Read Capacity Units Avg. Prov. Throughput	The average provisioned throughput (sum @ 5 min poll/300) read capacity of the deployment configuration for this table.
Consumed Write Capacity Units Avg. Prov. Throughput	The average provisioned throughput of consumed write capacity for the table.
Max Consumed Read Capacity Units	The maximum consumed read capacity of the deployment configuration for this table.
Max Consumed Write Capacity Units	The maximum consumed write capacity for the table.
Max Returned Item Count	The maximum returned item count for the table.

Max Successful Request Latency	The maximum latency for a successful table operation.
Min Consumed Read Capacity Units	The minimum consumed read capacity of the deployment configuration for this table.
Min Consumed Write Capacity Units	The minimum consumed write capacity for the table.
Min Returned Item Count	The minimum returned item count for the table.
Min Successful Request Latency	The minimum latency for successful table operations.
Returned Item Count	The datum counts of returned items for the table.
Successful Request Latency Counts	The number of successful table operations.
Summed Consumed Read Capacity Units	The sum total consumed read capacity of the deployment configuration for this table.
Summed Consumed Write Capacity Units	The sum total consumed write capacity for the table.
Summed Returned Item Count	The sum total of returned items for the table.
Summed System Errors	The sum of all system error counts.
Summed Throttled Requests	The sum of all table throttling requests.
Summed User Errors	The sum of all user error counts.
System Errors Counts	The datum count of all system errors.
Throttled Requests Count	The datum counts for all table throttling requests.
User Errors Counts	The datum counts for all user errors.

AWS: DDB Service Health

Object Name	Object Description
Date	The timestamp of the health update of this service.
Health	A text description of the health status of this AWS service.

AWS Direct Connect Service

AWS: Direct Connect Instance Configuration

Object Name	Object Description
AWS Device	The Direct Connection endpoint which the physical connection terminates on.

AWS Device v2	The Direct Connect endpoint on which the physical connection terminates.
Bandwidth	Bandwidth of the connection. Example: 1 Gbps (for regular connections), or 500Mbps (for hosted connections). A connection represents the physical network connection between the AWS Direct Connect location and the customer.
Connection ID	The ID of the connection.
Connection Name	The name of the connection. A connection represents the physical network connection between the AWS Direct Connect location and the customer.
Has Logical Redundancy	Indicates whether the connection supports a secondary BGP peer in the same address family (IPv4/IPv6). Possible values are yes, no or unknown.
Jumbo Frame Capable	Indicates whether jumbo frames (9001 MTU) are supported. Possible values are True or False.
Key	The key of the tag.
LAG ID	The ID of the LAG.
Location	Where the connection is located. A connection represents the physical network connection between the AWS Direct Connect location and the customer.
Owner Account	The account ID of the owner of the connection. A connection represents the physical network connection between the AWS Direct Connect location and the customer.
Provided By	AWS Direct Connect partner who is a member of the AWS Partner Network (APN). This is optional if your network is colocated with an existing AWS Direct Connect location.
Region	The AWS Region where the connection is located.

State	State of the connection. A connection represents the physical network connection between the AWS Direct Connect location and the customer. Valid Values: ordering requested pending available down deleting deleted rejected Ordering: The initial state of a hosted connection provisioned on an interconnect. The connection stays in the ordering state until the owner of the hosted connection confirms or declines the connection order. Requested: The initial state of a standard connection. The connection stays in the requested state until the Letter of Authorization (LOA) is sent to the customer. Pending: The connection has been approved, and is being initialized. Available: The network link is up, and the connection is ready for use. Down: The network link is down. Deleted: The connection has been deleted. Rejected: A hosted connection in the 'Ordering' state will enter the 'Rejected' state if it is deleted by the end customer.
Tags	Describes the tags associated with the specified Direct Connect resources.
Time DescribeLoa	The time of the most recent call to DescribeLoa for this connection.
Value	The value of the tag.
VLAN	The ID of the VLAN.

AWS: Direct Connect Instance Performance	
Object Name	Object Description
Bit Rate Ingress Data	The bit rate for inbound data to the AWS side of the connection.
Bit Rate Outbound Data	The bit rate for outbound data from the AWS side of the connection.
Connection State	The state of the connection. 0 indicates DOWN and 1 indicates UP.
CRC Error	The number of times cyclic redundancy check (CRC) errors are observed for the data received at the connection.
Health Inbound Traffic	"Indicates the health of the fiber connection for ingress (inbound) traffic to the AWS side of the connection. This metric is available for connections with 10 Gbps port speeds only."

Health Outbound Traffic	"Indicates the health of the fiber connection for egress (outbound) traffic from the AWS side of the connection. This metric is available for connections with 10 Gbps port speeds only."
Packet Rate Ingress Data	The packet rate for inbound data to the AWS side of the connection.
Packet Rate Outbound Data	The packet rate for outbound data from the AWS side of the connection.

AWS: Direct Connect Virtual Interface Configuration

Object Name	Object Description
Account	The AWS account that will own the new virtual interface.
Address Family	Indicates the address family for the BGP peer.
Address Family BGP	Indicates the address family for the BGP peer.
Amazon IP	Indicates the address family for the BGP peer.
Amazon Peer IP	IP address assigned to the Amazon interface. Example: 192.168.1.1/30
Amazon Side ASN	The autonomous system number (ASN) for the Amazon side of the connection.
Authentication Key	The authentication key for BGP configuration.
AWS Device	The Direct Connection endpoint which the virtual interface terminates on.
AWS Direct Connect/Virtual Private Gateway	The unique identifier for the Direct Connect-Virtual Private Gateway relationship.
BGP ASN	Autonomous system (AS) number for Border Gateway Protocol (BGP) configuration.
BGP Authentication Key	Authentication key for BGP configuration.
BGP peers	A list of the BGP peers configured on this virtual interface.
BGP Peers ASN	The autonomous system (AS) number for Border Gateway Protocol (BGP) configuration.
CIDR	CIDR notation for the advertised route.
Connection	The ID of the connection.
Customer IP	IP address assigned to the customer interface.

Device	The Direct Connection endpoint which the BGP peer terminates on.
Direct Connect Gateway	The ID of the direct connect gateway.
ID	The AWS resource ID for the virtual interface.
Jumbo Frame Capable	Indicates whether jumbo frames (9001 MTU) are supported. Possible values are True or False.
Key	The key of the tag.
Location	Where the connection is located.
MTU	The maximum transmission unit (MTU), in bytes. The supported values are 1500 and 9001. The default value is 1500.
Name	The name of the virtual interface assigned by the customer. Example: "My VPC"
Region	The AWS Region where the virtual interface is located.
Router Configuration	Information for generating the customer router configuration.
Routes	A list of routes to be advertised to the AWS network in this region (public virtual interface).
State	State of the virtual interface. One of: Confirming: The creation of the virtual interface is pending confirmation from the virtual interface owner. If the owner of the virtual interface is different from the owner of the connection on which it is provisioned, then the virtual interface will remain in this state until it is confirmed by the virtual interface owner. Verifying: This state only applies to public virtual interfaces. Each public virtual interface needs validation before the virtual interface can be created. Pending: A virtual interface is in this state from the time that it is created until the virtual interface is ready to forward traffic. Available: A virtual interface that is able to forward traffic. Down: A virtual interface that is BGP down. Deleting: A virtual interface is in this state immediately after calling DeleteVirtualInterface until it can no longer forward traffic. Deleted: A virtual interface that cannot forward traffic. Rejected: The virtual interface owner has declined creation of the virtual interface. If a virtual interface in the 'Confirming' state is deleted by the virtual interface owner, the virtual interface will enter the 'Rejected' state.
State BGP	The state of the BGP peer.

Tags	Describes the tags associated with the specified Direct Connect resources.
Type	The type of virtual interface. Example: private (Amazon VPC) or public (Amazon S3, Amazon DynamoDB, and so on.)
Up/Down State	The Up/Down state of the BGP peer
Value	The value of the tag.
Virtual Gateway	The ID of the virtual private gateway to a VPC. This only applies to private virtual interfaces.
VLAN	The VLAN ID. Example: 101
Your Peer IP	IP address assigned to the customer interface. Example: 192.168.1.2/30

AWS DynamoDB Service

AWS: DynamoDB Performance	
Object Name	Object Description
Conditional Check Failed Requests	The number of failed attempts to perform conditional writes.
Online Index Consumed Write Capacity	The number of write capacity units consumed when adding a new global secondary index to a table.
Online Index Percentage Progress	The percentage of completion when a new global secondary index is being added to a table.
Online Index Throttle Events	The number of write throttle events that occur when adding a new global secondary index to a table.
Provisioned Read Capacity Units	The number of provisioned read capacity units for a table or a global secondary index.
Provisioned Write Capacity Units	The number of provisioned write capacity units for a table or a global secondary index
Read Throttle Events	Requests to DynamoDB that exceed the provisioned read capacity units for a table or a global secondary index.
Returned Bytes	The number of bytes returned by GetRecords operations (Amazon DynamoDB Streams) during the specified time period.

Returned Records Count	The number of stream records returned by <code>GetRecords</code> operations (Amazon DynamoDB Streams) during the specified time period.
Successful Request Latency	The number of successful table operations requests.
Throttled Requests	Requests to DynamoDB that exceed the provisioned throughput limits on a table.
Time To Live Deleted Item Count	The number of items deleted by Time To Live (TTL) during the specified time period.
Write Throttle Events	Requests to DynamoDB that exceed the provisioned write capacity units for a table or a global secondary index.

AWS EBS Service

AWS: EBS Instance Configuration	
Object Name	Object Description
Volume ID	The unique identifier of this EBS volume.
IO Capacity	If provisioned, the peak IOPS of the volume.
Storage Capacity	The storage capacity of the EBS volume.
Type	Whether or not this EBS volume has been provisioned (type <code>consistent-iops</code> or <code>standard</code>).
Volume Status	The volume state.
EC2 Instance ID	The ID of the EC2 instance that is attached to the EBS Volume if any.
Attached State	Shows whether the EBS volume is attached to an EC2 instance.
Capacity	The size of the snapshot.
Description	The description of the snapshot.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).

Key	The tag key. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for your Amazon EC2 instances on the account that helps you track each owner of the instance and stack level. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.
Last State Count	Quantity of the last state collected only for creating and deleting states.
Owner	The ID of the snapshot.
Progress	The progress towards completion of the snapshot.
Snapshot ID	The AWS ID of the snapshot.
Snapshots	The volume of the EBS snapshot.
Start Time	The time the snapshot was started.
Status	The status of the snapshot.
Value	The tag value. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for your Amazon EC2 instances on the account that helps you track each owner of the instance and stack level. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.

AWS: EBS Instance Performance	
Object Name	Object Description
Burst Balance	The percent of General Purpose SSD (gp2) burst-bucket I/O credits available.
EBS Storage Capacity	The storage capacity of the EBS volume.
Volume Consumed Read Write Ops	The consumed total IOPS on the volume.

Volume Idle Time	The average time that the volume has been idle.
Volume Queue Length	The average queue length for the volume.
Volume Read Bytes	The average bytes read from the volume.
Volume Read Ops	The average number of read operations for the volume.
Volume Throughput Percent	The percentage of available throughput currently consumed by the volume.
Volume Total Read Time	The total read times for the volume.
Volume Total Write Time	The total write times for the volume.
Volume Write Bytes	The average bytes written by the volume.
Volume Write Ops	The average write operations for the volume.

AWS EC2 Service

AWS: EC2 Instance Configuration	
Object Name	Object Description
AMI Launch Index	The Amazon Machine Image (AMI) index within the EC2's launch reservation list.
Architecture	The architecture of the image.
Architecture	The instance architecture. Type: String. Valid values: i386 x86_64
Availability Zone	The Availability Zone under which the EC2 instance will be launched.
AWS EC2 Identifier Namespace	The namespace used to link the CCC application.
AWS EC2/EBS Volume	The volume ID of the Amazon EBS volume.
Block Device Attach Time	The attach time for an Amazon EBS volume mapped to the instance (for example, 2010-09-15T17:15:20.000Z)
Block Device Delete On Termination	Indicates whether the Amazon EBS volume is deleted on instance termination.
Block Device Name	The device name exposed to the instance (for example, /dev/sdh, or xvdh).

Block Device Status	The status for the Amazon EBS volume. Valid values: attaching attached detaching detached
CCC Application in AWS EC2 Instance	CCC Applications in EC2 Instances.
Client Token	The idempotency token you provided when you launched the instance. Type: String
Delete On Termination	Indicates whether the EBS volume is deleted on instance termination.
Description	The description of the AMI was provided during image creation.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
EBS Optimized	Indicates whether the instance is optimized for EBS I/O. Type: xsd:boolean
EC2 Unique ID	The unique identifier of the EC2 Instance that matches the identifier given by the CCC application.
Elastic IP	One or more elastic IP addresses assigned to the EC2 instance.
ENA Support	Specifies whether enhanced networking with ENA is enabled.
Encrypted	Indicates whether the encryption state of an EBS volume is changed while being restored from a backing snapshot.
ENI Description	The ENI description.
ENI ID	The ID of the ENI.
ENI MAC Address	The MAC address of the interface.
ENI Owner ID	The ID of the owner of the ENI.
ENI Private IP Address	The IP address of the interface within the subnet.
ENI Source/Dest Check	Flag to indicate whether to validate network traffic to or from this network interface.
ENI Status	The interface's status (available in-use).
ENI Subnet ID	The ID of the VPC subnet.
ENI VPC ID	The ID of the VPC.
Group	The name of the initial security group to which this EC2 instance is assigned.

Groups	A list of all current groups to which this EC2 instance belongs. Within a VPC, this can change post-launch.
Hypervisor	The hypervisor type of the image.
Hypervisor	The type of root device that the instance uses. Type: String. Valid values: ebs instance-store
Image Creation Date	The date and time the image was created.
Image ID	The Amazon Machine Image (AMI) id which this instance is booted from.
Image Location	The location of the AMI.
Image Name	The name of the AMI was provided during image creation.
Image Type	The type of image.
Instance ID	The unique identifier of this instance.
Instance Index	The "device name" (if any) of this instance along with its zone and ID. The "device name" is the value of the 'Name' key in the instance's launch-time tag set.
Instance Profile	The instance profile id and arn associated with this instance.
Instance State	The string representation of the instance's current state.
Instance Type	The cpu, core, and memory capacity type-name of this instance. For example: t1.micro, c1.medium, m1.xlarge.
Kernel ID	The kernel's unique identifier for this Amazon Machine Image (AMI). This is most useful when running a User Provided Kernel (UPK).
Key	The tag key. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for your account's Amazon EC2 instances that helps you track each instance's owner and stack level. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.

Key	The tag key. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for your account's Amazon EC2 instances that helps you track each instance's owner and stack level. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.
Key Pair Name	The name of the SSH key associated with the instance.
Launch Time	The UTC time when this instance was launched.
Lifecycle	Whether or not this instance is 'normal'; or 'spot'. A 'spot' lifecycle means this instance's compute-time was purchased in the 'spot market' and has limitations on how, where, and when it can run.
Monitoring Level	Whether or not AWS CloudWatch metrics are reported on a 'basic' (5 min.) or 'detailed' (1 min.) intervals.
Monitoring State	Indicates whether monitoring is enabled for the instance. Type: String. Valid values: disabled enabled
Persistent Lifecycle	Whether or not Termination Protection is enabled.
Placement Group	The Security Group where the instance is placed upon association with a Virtual Private Cloud (VPC).
Platform	A high-level name of the operating system running on the instance. If such instance is Linux/Unix based the Collection Object won't be visible, only if it is Windows
Platform Details	The platform details associated with the billing code of the AMI.
Private DNS Name	The private AWS Cloud DNS name associated with the instance. Each instance will always have at least one Private DNS name.
Private IP Address	The AWS cloud-local IP address.
Public	Indicates whether the image has public launch permissions.
Public DNS Name	The Public DNS Name (if any) associated with the instance. This DNS name is generally Internet routable and associated with one or more Elastic IPs.
Public IP Address	The public IP address of the instance.

Ramdisk ID	The unique identifier of the ramdisk (if any) associated with the instance.
Region	The AWS Region (datacenter) where the instance is located.
Reservation	The unique identifier associated with the Reservation which launched this instance.
Root Device	The root block device (filesystem) of the instance.
Root Device Name	The device name of the root device volume.
Root Device Type	The type of root device used by the AMI.
Root Device Type	The type of root device that the instance uses. Type: String. Valid values: ebs instance-store
Security Group Id	The ID of the security group.
Security Group Name	The name of the security group.
Snapshot ID	The ID of the snapshot.
Sriov Net Support	Specifies whether enhanced networking with the Intel 82599 Virtual Function interface is enabled.
State	The current state of the AMI.
State Reason	The reason for the most recent state transition, typically "User initiated", that the state of the instance has changed.
Subnet Component Unique Identifier	The SL1 component unique identifier for the associated subnet, if any.
Subnet ID	If associated with a VPC, this is the AWS identifier of the subnet within the VPC where the instance resides.
Tag	The key of the tag.
Tenancy	Multi IAM placement status
Value	The value of the tag.

Value	The tag value. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for your account's Amazon EC2 instances that helps you track each instance's owner and stack level. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.
Value	The tag value. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for your account's Amazon EC2 instances that helps you track each instance's owner and stack level. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.
Virtualization Type	The type of virtualization of the AMI.
Virtualization Type	This is always listed as 'paravirtual' or 'hvm' (xen or ovm).
Volume Size	The size of the volume, in GiBs.
Volume Type	The volume type.
VPC ID	If associated with a Virtual Private Cloud (VPC), this is the VPC AWS identifier.

AWS: EC2 Instance Performance

Object Name	Object Description
CPU Utilization	The percentage of allocated EC2 compute units that are currently in use on the instance. This metric identifies the processing power required to run an application upon a selected instance.

Disk Read Bytes	Bytes read from all ephemeral disks available to the instance (if your instance uses Amazon EBS, see Amazon EBS Metrics.) This metric is used to determine the volume of the data the application reads from the hard disk of the instance. This can be used to determine the speed of the application.
Disk Read Ops	Completed read operations from all ephemeral disks available to the instance (if your instance uses Amazon EBS, see Amazon EBS Metrics.) This metric identifies the rate at which an application reads a disk. This can be used to determine the speed in which an application reads data from a hard disk.
Disk Write Bytes	Bytes written to all ephemeral disks available to the instance (if your instance uses Amazon EBS, see Amazon EBS Metrics.) This metric is used to determine the volume of the data the application writes onto the hard disk of the instance. This can be used to determine the speed of the application.
Disk Write Ops	Completed write operations to all ephemeral disks available to the instance (if your instance uses Amazon EBS, see Amazon EBS Metrics.) This metric identifies the rate at which an application writes to a hard disk. This can be used to determine the speed in which an application saves data to a hard disk.
Network In	The number of bytes received on all network interfaces by the instance. This metric identifies the volume of incoming network traffic to an application on a single instance.
Network Out	The number of bytes sent out on all network interfaces by the instance. This metric identifies the volume of outgoing network traffic to an application on a single instance.
Network Packets In	The number of packets received on all network interfaces by the instance. This metric identifies the volume of incoming traffic in terms of the number of packets on a single instance. This metric is available for basic monitoring only.
Network Packets Out	The number of packets sent out on all network interfaces by the instance. This metric identifies the volume of outgoing traffic in terms of the number of packets on a single instance. This metric is available for basic monitoring only.

Status Check Failed	A combination of StatusCheckFailed_Instance and StatusCheckFailed_System that reports if either of the status checks has failed. Values for this metric are either 0 (zero) or 1 (one.) A zero indicates that the status check passed. A one indicates a status check failure. Note Status check metrics are available at 1 minute frequency. For a newly launched instance, status check metric data will only be available after the instance has completed the initialization state. Status check metrics will become available within a few minutes of being in the running state.
Status Check Failed for Instance	Reports whether the instance has passed the EC2 instance status check in the last minute. Values for this metric are either 0 (zero) or 1 (one.) A zero indicates that the status check passed. A one indicates a status check failure. Note Status check metrics are available at 1 minute frequency. For a newly launched instance, status check metric data will only be available after the instance has completed the initialization state. Status check metrics will become available within a few minutes of being in the running state.
Status Check Failed for System	Reports whether the instance has passed the EC2 system status check in the last minute. Values for this metric are either 0 (zero) or 1 (one.) A zero indicates that the status check passed. A one indicates a status check failure. Note Status check metrics are available at 1 minute frequency. For a newly launched instance, status check metric data will only be available after the instance has completed the initialization state. Status check metrics will become available within a few minutes of being in the running state.

AWS: EC2 Service Health

Object Name	Object Description
Date	The timestamp of this health status update string.
Health	A text description of this AWS Service's health status.

AWS ECS Service

AWS: ECS Cluster Instance Configuration

Object Name	Object Description
Active Services	The number of services that are running on the cluster in an ACTIVE state.
Agent Connected	This parameter returns true if the agent is connected to Amazon ECS. Registered instances with an agent that may be unhealthy or stopped return false.
AWS ECS Instance / EC2	The EC2 instance ID of the container instance.
Cluster ARN	The Amazon Resource Name (ARN) that identifies the cluster.
Cluster Name	The name of the cluster.
Cluster Status	The status of the cluster. The valid values are ACTIVE or INACTIVE .
Container Instance ARN	The Amazon Resource Name (ARN) of the container instance.
Container Instance ARN	The Amazon Resource Name (ARN) of the container instance.
Key	The key of the tag, that belongs to the ECS Cluster instance.
Pending Tasks	The number of tasks in the cluster that are in the PENDING state.
Pending Tasks	The number of tasks on the container instance that are in the PENDING status.
Registered At	The Unix time stamp for when the container instance was registered.
Registered Container Instances	The number of container instances registered into the cluster.
Registered CPU	This value represents the total amount reserved of the resource CPU that can be allocated on the container instance to tasks.
Registered Memory	This value represents the total amount reserved of the resource Memory that can be allocated on the container instance to tasks.
Registered Ports	This value represents the total amount reserved of the resource Ports that can be allocated on the container instance to tasks.
Registered Ports UDP	This value represents the total amount reserved of the resource Ports UDP that can be allocated on the container instance to tasks.

Remaining CPU	This value represents the remaining of resource CPU that has not already been allocated to tasks and is therefore available for new tasks.
Remaining Memory	This value represents the remaining of resource Memory that has not already been allocated to tasks and is therefore available for new tasks.
Remaining Ports	This value represents the remaining of resource Ports that has not already been allocated to tasks and is therefore available for new tasks.
Remaining Ports UDP	This value represents the remaining of resource Ports UDP that has not already been allocated to tasks and is therefore available for new tasks.
Running Tasks	The number of tasks in the cluster that are in the RUNNING state.
Running Tasks	The number of tasks on the container instance that are in the RUNNING status.
Statistic Name	The name of the key value pair. For environment variables, this is the name of the environment variable. (EC2 Launch Type)
Statistic Name	The name of the key value pair. For environment variables, this is the name of the environment variable. (Fargate Launch Type)
Statistic Value	The value of the key value pair. For environment variables, this is the value of the environment variable. (EC2 Launch Type)
Statistic Value	The value of the key value pair. For environment variables, this is the value of the environment variable. (Fargate Launch Type)
Status	The status of the container instance. The valid values are ACTIVE , INACTIVE , or DRAINING.
Value	The value of the tag, that belongs to the ECS Cluster instance.

AWS: ECS Cluster Instance Performance

Object Name	Object Description
-------------	--------------------

CPU Reservation	"The percentage of CPU units that are reserved by running tasks in the cluster. Cluster CPU reservation is measured as the total CPU units that are reserved by Amazon ECS tasks on the cluster, divided by the total CPU units that were registered for all of the container instances in the cluster. This metric is only used for tasks using the EC2 launch type."
CPU Utilization	"The percentage of CPU units that are used in the cluster or service. Cluster CPU utilization is measured as the total CPU units in use by Amazon ECS tasks on the cluster, divided by the total CPU units that were registered for all of the container instances in the cluster. Cluster CPU utilization metrics are only used for tasks using the EC2 launch type."
Memory Reservation	"The percentage of memory that is reserved by running tasks in the cluster. Cluster memory reservation is measured as the total memory that is reserved by Amazon ECS tasks on the cluster, divided by the total amount of memory that was registered for all of the container instances in the cluster. This metric is only used for tasks using the EC2 launch type."
Memory Utilization	"The percentage of memory that is used in the cluster or service. Cluster memory utilization is measured as the total memory in use by Amazon ECS tasks on the cluster, divided by the total amount of memory that was registered for all of the container instances in the cluster. Cluster memory utilization metrics are only used for tasks using the EC2 launch type."

AWS: ECS Cluster Services Configuration

Object Name	Object Description
AWS ECS Service/Classic Load Balancer	The load balancer used for the service if it has created with a classic load balancer type.
AWS ECS Service/Security Groups	The security groups associated with this ECS service.
AWS ECS Service/Subnets	The subnets associated with this ECS service.
AWS ECS Service/Target Group	The target group that is associated with this service if it was created with an application or network load balancer.
AWS ECS Service/VPC	The VPC associated with the service.

Container Name	The container being used with this service on the associated application ELB.
Container Port	The port of the container being used with this service on the associated application ELB.
Created At	The service creation time.
Desired Tasks	The desired number of instantiations of the task definition to keep running on the service.
Key	The key of the tag, that belongs to the ECS Cluster services.
Launch Type	Type of ECS launch (EC2 or Fargate).
Pending Tasks	The number of tasks in the cluster that are in the PENDING state.
Platform Version	The platform version on which your task is running.
Running Tasks	The number of tasks in the cluster that are in the RUNNING state.
Service ARN	The Amazon ID (ARN) of the ECS service.
Service Name	Name of the ECS Service.
Service Status	The status of the service. The valid values are ACTIVE , DRAINING , or INACTIVE.
Service Type	The scheduling strategy to use for the service (Daemon or Replica).
Task Definition	The ARN of the ECS task definition.
Value	The value of the tag, that belongs to the ECS Cluster services.

AWS: ECS Cluster Services Performance

Object Name	Object Description
CPU Utilization	The percentage of CPU units that are used in the cluster or service.
Desired Tasks Count	The desired number of instantiations of the task definition to keep running on the service.
Memory Utilization	The percentage of memory that is used in the cluster or service.
Running Tasks Count	The number of tasks in the services that are in the RUNNING state.

Total Pending Deployments	The total pending deployments of the service.
Total Running Deployments	The total running deployments of the service.

AWS: ECS Cluster Tasks Configuration	
Object Name	Object Description
Container Id	The Globally Unique Identifier (GUID) of the container from the Amazon Resource Name (ARN).
Name	The name of the container.
Container Id	The Globally Unique Identifier (GUID) of the container from the Amazon Resource Name (ARN).
Image	The image used to start a container. This string is passed directly to the Docker daemon.
Last Status	The last known status of the container.
Memory	The hard limit (in MiB) of memory to present to the container. If your container attempts to exceed the memory specified here, the container is killed.
Bind IP	The IP address that the container is bound to on the container instance.
Connectivity	The connectivity status of a task.
Container Port	The port number on the container that is used with the network binding.
CPU	The number of CPU units used by the task as expressed in a task definition. It can be expressed as an integer using CPU units. If you are using the EC2 launch type, this field is optional. If you are using the Fargate launch type, this field is required.
CPU	The number of cpu units reserved for the container.
Created At	The Unix timestamp for when the task was created (the task entered the PENDING state).
Desired Status	The desired status of the task.
Essential	If the essential parameter of a container is marked as true , and that container fails or stops for any reason, all other containers that are part of the task are stopped. If the essential parameter of a container is marked as false, then its failure does not affect the rest of the containers in a task.

Group	The name of the task group associated with the task.
Health	The health status of the container. If health checks are not configured for this container in its task definition, then it reports the health status as UNKNOWN.
Health Status	The health status for the task, which is determined by the health of the essential containers in the task. If all essential containers in the task are reporting as HEALTHY , then the task status also reports as HEALTHY . If any essential containers in the task are reporting as UNHEALTHY or UNKNOWN , then the task status also reports as UNHEALTHY or UNKNOWN , accordingly.
Host Port	"The port number on the container instance to reserve for your container. If you are using containers in a task with the awsvpc or host network mode, the hostPort can either be left blank or set to the same value as the containerPort. If you are using containers in a task with the bridge network mode, you can specify a non-reserved host port for your container port mapping, or you can omit the hostPort (or set it to 0) while specifying a containerPort and your container automatically receives a port in the ephemeral port range for your container instance operating system and Docker version."
Key	The key of the tag, that belongs to the ECS Cluster tasks.
Last Status	The last known status of the task.
Launch Type	The launch type on which your task is running. Options are: EC2 or Fargate.
MAC Address	The MAC Address of the network interface.
Memory	The amount of memory (in MiB) used by the task as expressed in a task definition. If you are using the EC2 launch type, this field is optional. If you are using the Fargate launch type, this field is required.
Network Interface Id	The Network Interface Id of the elastic network interface attached if any.
Network Mode	The Network mode of the task. Options are: awsvpc, bridge, host or none.
Platform Version	The platform version on which your task is running. A platform version is only specified for tasks using the Fargate launch type. If one is not specified, the LATEST platform version is used by default.

Private IP	Private IP address of the elastic network interface.
Protocol	The protocol used for the network binding. Options are tpc or udp.
Started At	The Unix timestamp for when the task started (the task transitioned from the PENDING state to the RUNNING state).
Started By	The tag specified when a task is started. If the task is started by an Amazon ECS service, then the startedBy parameter contains the deployment ID of the service that starts it.
Subnet Id	The subnet id of the elastic network interface attachment.
Task ARN	The Amazon Resource Name (ARN) of the task.
Task Definition	The ARN of the task definition that creates the task.
Value	The value of the tag, that belongs to the ECS Cluster tasks.
Version	The version counter for the task. Every time a task experiences a change that triggers a CloudWatch event, the version counter is incremented. If you are replicating your Amazon ECS task state with CloudWatch Events, you can compare the version of a task reported by the Amazon ECS API actions with the version reported in CloudWatch Events for the task (inside the detail object) to verify that the version in your event stream is current.

AWS: ECS Service Health

Object Name	Object Description
Date	The date of the RSS feed entry.
Health	The message of the RSS feed entry.

AWS EFS Service

AWS: EFS File System Configuration

Object Name	Object Description
-------------	--------------------

Creation Time	The time that the file system was created.
Creation Token	The opaque string specified in the request.
Encrypted	A Boolean value that, if true, indicates that the file system is encrypted.
File System Id	The ID of the file system, assigned by Amazon EFS.
Key	The tag key (String).
KMS Key Id	The ID of an AWS Key Management Service (AWS KMS) customer master key (CMK) that was used to protect the encrypted file system.
Last Size Update	The latest known time at which the size of data was determined.
Life Cycle State	The lifecycle phase of the file system.
Name	You can add tags to a file system, including a Name tag. If the file system has a Name tag, Amazon EFS returns the value in this field.
Number of Mount Targets	The current number of mount targets that the file system has.
Owner Id	The AWS account that created the file system. If the file system was created by an IAM user, the parent account to which the user belongs is the owner.
Performance Mode	The performance mode of the file system.
Provisioned Throughput	The throughput, measured in MiB/s, that you want to provision for a file system.
Throughput Mode	The throughput mode for a file system. There are two throughput modes to choose from for your file system: bursting and provisioned.
Value	The value of the tag key.

AWS: EFS File System Performance

Object Name	Object Description
Burst Credit Balance	The number of burst credits that a file system has.
Client Connections	The number of client connections to a file system. When using a standard client, there is one connection per mounted Amazon EC2 instance.

Data Read IO Bytes	The number of bytes for each file system read operation.
Data Write IO Bytes	The number of bytes for each file write operation.
Metadata IO Bytes	The number of bytes for each metadata operation.
Percent IO Limit	Shows how close a file system is to reaching the I/O limit of the General Purpose performance mode. If this metric is at 100% more often than not, consider moving your application to a file system using the Max I/O performance mode.
Permitted Throughput	The maximum amount of throughput a file system is allowed. For file systems in the Provisioned Throughput mode, if the amount of storage allows your file system to drive a higher amount of throughput than you provisioned, this metric will reflect the higher throughput instead of the provisioned amount. For file systems in the Bursting Throughput mode, this value is a function of the file system size and BurstCreditBalance.
Total IO Bytes	The number of bytes for each file system operation, including data read, data write, and metadata operations.

AWS: EFS File System Usage Performance

Object Name	Object Description
Size of Infrequent Access Data	The latest known metered size (in bytes) of data stored in the Infrequent Access storage class.
Size of Standard Access Data	The latest known metered size (in bytes) of data stored in the Standard storage class.
Total File System Size	The latest known metered size (in bytes) of data stored in the file system.

AWS EKS Service

AWS: EKS Cluster Instance Configuration

Object Name	Object Description
ARN	The ARN that AWS provides for this cluster within EKS.

Certificate Authority	The Certificate Authority used by this cluster for signing requests.
Client Request Token	Unique, case-sensitive identifier that you provide to ensure the idempotency of the request.
Created At	The time at which this cluster was created.
Endpoint	The REST endpoint used to make API calls into the Kubernetes cluster.
Keys	Tags keys assigned to this cluster.
Logging Enabled	An object representing the enabled or disabled Kubernetes control plane logs for your cluster.
Name	The name of this cluster within EKS.
Platform Version	The platform version of your Amazon EKS cluster.
Role ARN	The ARN of the AWS role used to manage this cluster.
Security Group IDs	The security groups associated with your cluster.
Status	The active status of this cluster.
Subnet IDs	The subnets associated with your cluster.
Tags	Tags assigned to this cluster.
Types	The available cluster control plane log types.
Values	Tags values assigned to this cluster.
Version	The version of Kubernetes that this cluster is running.
VPC	The VPCs associated with your cluster.
VPC Endpoint Private Access	This parameter indicates whether the Amazon EKS private API server endpoint is enabled. If the Amazon EKS private API server endpoint is enabled, Kubernetes API requests that originate from within your cluster's VPC use the private VPC endpoint instead of traversing the internet.
VPC Endpoint Public Access	This parameter indicates whether the Amazon EKS public API server endpoint is enabled. If the Amazon EKS public API server endpoint is disabled, your clusters Kubernetes API server can receive only requests that originate from within the cluster VPC.

AWS: EKS Token Manager

Object Name	Object Description
-------------	--------------------

Token Created	This token is used to authenticate with the EKS Cluster, True if the token was created otherwise False.
Token Expiration Time	The token expiration time.

AWS Elastic Beanstalk Service

AWS: Elastic Beanstalk Application Instance Health Configuration

Object Name	Object Description
Instance Id	The ID of the Amazon EC2 instance.
Availability Zone	The availability zone in which the instance runs.
Color	Represents the color indicator that gives you information about the health of the EC2 instance.
Health Status	Returns the health status of the instance.
Launched At	The time at which the EC2 instance was launched.

AWS: Elastic Beanstalk Application Performance

Object Name	Object Description
Environment Name Cache	Provides cache of Environment Names for consumption by other collection objects.
P10 Latency	The average latency for the slowest 90 percent of requests over the last 10 seconds.
P50 Latency	The average latency for the slowest 50 percent of requests over the last 10 seconds.
P75 Latency	The average latency for the slowest 25 percent of requests over the last 10 seconds.
P85 Latency	The average latency for the slowest 15 percent of requests over the last 10 seconds.
P90 Latency	The average latency for the slowest 10 percent of requests over the last 10 seconds.
P95 Latency	The average latency for the slowest 5 percent of requests over the last 10 seconds.
P99 Latency	The average latency for the slowest 1 percent of requests over the last 10 seconds.

P999 Latency	The average latency for the slowest 0.1 percent of requests over the last 10 seconds.
Request Count	Average number of requests handled by the web server per second over the last 10 seconds.
Status 2xx	The average number of requests over the last 10 seconds that resulted in a 2xx (200, 201, etc.) status code.
Status 3xx	The average number of requests over the last 10 seconds that resulted in a 3xx (300, 301, etc.) status code.
Status 4xx	The average number of requests over the last 10 seconds that resulted in a 4xx (400, 401, etc.) status code.
Status 5xx	The average number of requests over the last 10 seconds that resulted in a 5xx (500, 501, etc.) status code.

AWS: Elastic Beanstalk Data Tier Configuration

Object Name	Object Description
Engine	The name of the database engine to use for this instance.
Instance Class	The database instance type.
Multi Availability Zone	Specifies whether a database instance Multi-AZ deployment needs to be created.
Storage (GB)	The allocated database storage size, specified in gigabytes.
When Deleted	Decides whether to delete or snapshot the DB instance on environment termination.

AWS: Elastic Beanstalk Network Tier Configuration

Object Name	Object Description
-------------	--------------------

Associate Public IP Address	Specifies whether to launch instances with public IP addresses in your Amazon VPC. Instances with public IP addresses do not require a NAT device to communicate with the Internet. You must set the value to true if you want to include your load balancer and instances in a single public subnet.
Instance Subnets	The IDs of the Auto Scaling group subnet or subnets. If you have multiple subnets, specify the value as a single comma-delimited string of subnet IDs (for example, "subnet-11111111,subnet-22222222").
RDS Subnets	Contains the IDs of the database subnets. This is only used if you want to add an Amazon RDS DB Instance as part of your application. If you have multiple subnets, specify the value as a single comma-delimited string of subnet IDs (for example, "subnet-11111111,subnet-22222222").
VPC ID	The ID for your Amazon VPC.

AWS: Elastic Beanstalk Service Health

Object Name	Object Description
Date	A timestamp when the health status of this service was originally written.
Health	A text description of the health status of this AWS service.

AWS: Elastic Beanstalk Web Tier Configuration

Object Name	Object Description
Add Instance When (>)	If the measurement is higher than this number for the breach duration, a trigger is fired. 0 to 20000000.
Allow URL fopen:	Specifies if PHP file functions are allowed to retrieve data from remote locations, such as websites or FTP servers.
Availability Zones	The availability zones configuration for the beanstalk application.
AWS Elastic Beanstalk/EC2	The ec2 associated with the beanstalk application.

Connection Draining Enabled	If the beanstalk load balancer attempts connections to an Amazon EC2 instance before forcibly closing connections.
Cross Zone Load Balancing Enabled	If the beanstalk load balancer attempts connections to an Amazon EC2 instance in multiple availability zones.
Deployment Batch Size (%)	The size of the set of instances to deploy in each batch.
Display Errors	Information about the errors that are common to all actions.
Environment ID	The beanstalk application environment id.
Environment Name	The beanstalk application environment name.
Environment Type	The beanstalk application environment type.
Instance Port	The beanstalk environment port.
Instance Type	The beanstalk environment type.
Log Publication	Copy the log files for the Amazon EC2 instances of your application into the Amazon S3 bucket associated with your application (valid values are true or false).
Max Execution Time	The maximum number of attempts that Elastic Beanstalk attempts to send the message to the web application that will process it before moving the message to the dead letter queue.
Memory Limit	Amount of memory allocated to the PHP environment.
Notification Protocol	Protocol used to send notifications to your endpoint.
Number Instances	Number of instances.
NumProcesses	The number of daemon processes that should be started for the process group when running WSGI applications.
NumThreads	The number of threads to be created to handle requests in each daemon process within the process group when running WSGI applications.
Remove Instance When (<)	If the measurement falls below this number for the breach duration, a trigger is fired. 0 to 20000000.
Rolling Updates Enabled	Rolling configuration update batches can be processed periodically (time-based), with a delay between each batch, or based on health. The following values are supported for RollingUpdateType: Health Time Immutable.
Send To	Endpoint where you want to be notified of important events affecting your application.

StaticFiles	Information about setting of the virtual path and directory mappings in the Static Files section of the Modify software configuration page.
WSGIPath	The file that contains the WSGI application. This file must have an application callable (default: application.py).
Zlib Output Compression	Specifies whether or not PHP should use compression for output.

AWS ElastiCache Service

AWS: ElastiCache Cluster Configuration	
Object Name	Object Description
Auto Upgrade	Whether or not the cluster engine can be upgrading during maintenance.
Cluster Node Type	The EC2 class of the cluster nodes.
Creation Time	The time the cluster was created.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Endpoint	The DNS entry for the configuration endpoint.
Engine	The cache engine (either Redis or Memecache) of the cluster.
Id	The unique identifier of the cluster.
Maintenance Window	The datetime when the cluster can undergo maintenance.
Name	The name of the cache cluster.
Nodes	The number of nodes in the cluster.
Notification ARN	The Amazon Resource Name (ARN), if any, of the SNS instance for event notifications.
Parameter Groups	The parameter groups of the cluster.
Preferred Zone	The name of the Availability Zone in which the cluster is located or "Multiple" if the cache nodes are located in different Availability Zones.

Security Groups	The security groups of the cluster.
Status	The status of the cluster.
Version	The version of the engine.

AWS: ElastiCache Cluster Performance

Object Name	Object Description
CPU Utilization	The percentage of CPU utilization.
Freeable Memory	The amount of free memory on the cluster.
Network Bytes In	The bytes the host has read from the network.
Network Bytes Out	The number of bytes the host has written to the network.
Swap Usage	The swap used, if any, on the cluster.

AWS: ElastiCache Memcached Node Configuration

Object Name	Object Description
Availability Zone	The Availability Zone where this node was created and now resides, also referred to as Customer Availability Zone.
Creation Time	The creation time of the node.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Endpoint	The DNS entry of the node.
Id	The node's unique identifier.
Name	The name of the cluster node.
Port	The TCP port of the service running on the node.
Status	The node's EC2 status.

AWS: ElastiCache Memcached Node Performance

Object Name	Object Description
-------------	--------------------

Bytes Read into Memcached	The number of bytes that have been read from the network by the cache node.
Bytes Used for Cache Items	The number of bytes used to store cache items.
Bytes Used For Hash	The number of bytes currently used by hash tables.
Bytes Written Out From Memcached	The number of bytes that have been written to the network by the cache node.
Check and Set Bad Values	The number of CAS (check and set) requests the cache has received where the Cas value did not match the Cas value stored.
Check and Set Misses	The number of Cas requests the cache has received where the key requested was not found.
Check and Set Request Hits	The number of Cas requests the cache has received where the requested key was found and the Cas value matched.
Config Get Requests	The cumulative number of "config get" requests.
Config Set Requests	The cumulative number of "config set" requests.
Current Connections	The number of connections connected to the cache. Note that due to the design of Memcached, this will always return a minimum count of 10.
Current Items	The number of items currently stored in the cache. Note that due to the design of Memcached, this will always return a minimum count of 10.
Decrement Hits	The number of decrement requests the cache has received where the requested key was found.
Decrement Misses	The number of decrement requests the cache has received where the requested key was not found.
Delete Hits	The number of delete requests the cache has received where the requested key was found.
Delete Misses	The number of delete requests the cache has received where the requested key was not found.
Evicted Unfetched Items	The number of valid items evicted from the least recently used cache (LRU) which were never touched after being set.
Evictions	The number of non-expired items the cache evicted to allow space for new writes.
Expired Unfetched Items	The number of expired items reclaimed from the LRU which were never touched after being set.

Flush Commands	The number of flush commands the cache has received.
Get Commands	The number of get commands the cache has received.
Get Hits	The number of get requests the cache has received where the key requested was found.
Get Misses	The number of get requests the cache has received where the key requested was not found.
Increment Hits	The number of increment requests the cache has received where the key requested was found.
Increment Misses	The number of increment requests the cache has received where the key requested was not found.
New Connections	The number of new connections the cache has received. This is derived from the memcached total_connections statistic by recording the change in total_connections across a period of time. This will always be at least 1, due to a connection reserved for an ElastiCache.
New Items	The number of new items the cache has received. This is derived from the memcached total_connections statistic by recording the change in total_connections across a period of time. This will always be at least 1, due to a connection reserved for a ElastiCache.
Reclaimed	The number of expired items the cache evicted to allow space for new writes.
Set Commands	The number of set commands the cache has received.
Slabs Moved	The total number of slab pages that have been moved.
Stored Configurations	The number of configurations stored.
Touch Commands	The cumulative number of "touch" requests.
Touch Hits	The number of keys that have been touched and were given a new expiration time.
Touch Misses	The number of items that have been touched, but were not found.
Unused Memory	The amount of unused memory the cache can use to store items.

AWS: ElastiCache Node Performance

Object Name	Object Description
-------------	--------------------

CPU Credit Balance	The number of earned CPU credits that an instance has accrued since it was launched or started. CPU credit metrics are available at a five-minute frequency only.
CPU Credit Usage	The number of CPU credits spent by the instance for CPU utilization. CPU credit metrics are available at a five-minute frequency only.
CPU Utilization	The percentage of CPU utilization for the entire host.
Freeable Memory	The amount of free memory available on the host.
Network Bytes In	The number of bytes the host has read from the network.
Network Bytes Out	The number of bytes sent out on all network interfaces by the instance.
Network Packets In	The number of packets received on all network interfaces by the instance. This metric identifies the volume of incoming traffic in terms of the number of packets on a single instance.
Network Packets Out	The number of packets sent out on all network interfaces by the instance. This metric identifies the volume of outgoing traffic in terms of the number of packets on a single instance.
Swap Usage	The amount of swap used on the host.

AWS: ElastiCache Redis Node Configuration

Object Name	Object Description
Availability Zone	The Availability Zone where this node was created and now resides, also referred to as Customer Availability Zone.
Creation Time	The datetime when this node was created.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Endpoint	The DNS entry of this node.
Id	The unique identifier of the Redis node.
Name	The name of the Redis node.
Port	The TCP port of the service cache on the node.
Status	The EC2 status of the node.

AWS: ElastiCache Redis Node Performance

Object Name	Object Description
Bytes Used For Cache	The total number of bytes allocated by Redis.
Cache Hits	The number of successful key lookups.
Cache Misses	The number of unsuccessful key lookups.
Current Connections	The number of client connections, excluding connections from read replicas.
Current Items	The number of items in the cache.
Engine CPU Utilization	Provides more precise visibility into the load of the Redis process itself.
Evictions	The number of keys that have been evicted due to the maxmemory limit.
Get Type Commands	The total number of get types of commands.
Hash Based Commands	The total number of commands that are hash-based.
Hyper Log Log Based Cmds	This is derived from the Redis commandstats statistic by summing all of the pf type of commands (pfadd, pfcount, pfmerge).
Key Based Commands	The total number of commands that are key-based.
List Based Commands	The total number of commands that are list-based.
New Connections	The total number of connections that have been accepted by the server during this period.
Reclaimed	The total number of key expiration events.
Replication Bytes	The number of bytes that the primary is sending to all of its replicas.
Replication Lag	This metric is only applicable for a cache node running as a read replica. It represents how far behind, in seconds, the replica is in applying changes from the primary cache cluster.
Save In Progress	This binary metric returns 1 whenever a background save (forked or forkless) is in progress, and 0 otherwise.
Set Based Commands	The total number of commands that are set-based.
Set Type Commands	The total number of set types of commands.
Sorted Set Based Commands	The total number of commands that are sorted set-based.
String Based Commands	The total number of commands that are string-based.

AWS: ElastiCache Service Health

Object Name	Object Description
Date	A timestamp when this service's health status was originally written.
Health	A text description of this AWS Service's health status.

AWS ELB Service

AWS: ELB Instance Configuration

Object Name	Object Description
Availability Zones	The availability zones where the ELB is deployed.
AWS ELB/EC2	The unique identifier set for the EMR-EC2 relationship.
AWS ELB/VPC	The unique identifier set for the VPC relationship.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Key	The key of the tag, that belongs to the elastic load balancer.
Listener Instance Port	The listener port for the ELB.
Load Balanced Instances	The group label for the EMR-EC2 relationship.
Load Balancer Name	The name of the ELB.
Load Balancer Port	The load-balanced port for the ELB.
Load Balancer Protocol	The protocol supported (HTTP, SMTP, etc.) by the ELB.
Value	The value of the tag, that belongs to the application load balancer.

AWS: ELB Instance Performance

Object Name	Object Description
-------------	--------------------

Backend Connection Errors	The count of the number of connections that were not successfully established between the load balancer and the registered instances. Because the load balancer will retry when there are connection errors, this count can exceed the request rate.
Healthy Host Count Average	The average number of healthy hosts in the ELB.
Healthy Host Count Maximum	The maximum number of healthy hosts in the ELB.
HTTPCode_Backend_2xx	The count of the HTTP 200 status codes on the back-end of the ELB.
HTTPCode_Backend_3xx	The count of the HTTP 300 status codes on the back-end of the ELB.
HTTPCode_Backend_4xx	The count of HTTP 400 status codes (errors) on the back-end of the ELB.
HTTPCode_Backend_5xx	The count of the HTTP 500 status codes (errors) on the back-end of the ELB.
Latency Average	The average latency across the ELB.
Latency Maximum	The maximum latency across the ELB.
Latency Minimum	The minimum latency across the ELB.
Request Count	The total requests datum counts for the ELB.
Spillover Count	A count of the total number of requests that were rejected due to the queue being full.
Surge Queue Length	A count of the total number of requests that are pending submission to a registered instance.
Unhealthy Host Count Average	The average number of unhealthy hosts in the ELB.
Unhealthy Host Count Maximum	The maximum number of unhealthy hosts in the ELB.
Unhealthy Host Count Minimum	The minimum number of unhealthy hosts in the ELB.

AWS: ELB Service Health

Object Name	Object Description
Date	The timestamp when this status was written.
Health	A text description of this AWS Service's health status.

AWS: ELB Target Group Instance Configuration

Object Name	Object Description
Application LoadBalancer ID	The dimension resource identifier of the load balancer that forwards traffic to the target group.
AWS Target Group/EC2	The unique identifiers of EC2s for the target groups.
AWS Target Group/VPC	The unique identifiers of VPC for target groups.
HealthCheck Interval	The approximate amount of time, in seconds, between health checks of an individual target.
HealthCheck Path	The destination for the health check request.
HealthCheck Port	The port to use to connect with the target.
HealthCheck Protocol	The protocol to use to connect with the target.
HealthCheck Timeout	The amount of time, in seconds, during which no response means a failed health check.
Healthy Threshold Count	The number of consecutive health checks successes required before considering an unhealthy target healthy.
Key	The keys of the tag, that belongs to the target group.
Load Balancer ARNs	The Amazon Resource Name (ARN) of the application or network load balancers.
Target Group ARN	The Amazon Resource Name (ARN) of the target group.
Target Group Id	The unique identifier for the target group component.
Target Group Name	The name of the target group component.
Target Group Port	The port on which the targets are listening.
Target Group Protocol	The protocol to use for routing traffic to the targets.
Target Group VPC Id	The ID of the VPC for the targets.
Target Type	The type of target that you must specify when registering targets with this target group. The possible values are instance (targets are specified by instance ID) or ip (targets are specified by IP address).
Unhealthy Threshold Count	The number of consecutive health check failures required before considering the target unhealthy.
Value	The value of the tag, that belongs to the target group.

AWS: ELB Target Group Instance Performance

Object Name	Object Description
Healthy Host Count (Average)	The number of targets that are considered healthy.
Healthy Host Count (Maximum)	The number of targets that are considered healthy.
Healthy Host Count (Minimum)	The number of targets that are considered healthy.
HTTPCode Target 2XX Count	The number of HTTP response codes generated by the targets. This does not include any response codes generated by the load balancer. For now we are not collecting data from amazon api, because it is not supported yet for Network ELB associated with Target Group.
HTTPCode Target 3XX Count	The number of HTTP response codes generated by the targets. This does not include any response codes generated by the load balancer. For now we are not collecting data from amazon api, because it is not supported yet for Network ELB associated with Target Group.
HTTPCode Target 4XX Count	The number of HTTP response codes generated by the targets. This does not include any response codes generated by the load balancer. For now we are not collecting data from amazon api, because it is not supported yet for Network ELB associated with Target Group.
HTTPCode Target 5XX Count	The number of HTTP response codes generated by the targets. This does not include any response codes generated by the load balancer. For now we are not collecting data from amazon api, because it is not supported yet for Network ELB associated with Target Group.
Request Count Per Target	The average number of requests received by each target in a target group. You must specify the target group using the TargetGroup dimension. For now we are not collecting data from amazon api, because it is not supported yet for Network ELB associated with Target Group.
Target Connection Error Count	The number of connections that were not successfully established between the load balancer and target. This metric does not apply if the target is a Lambda function.

Target Response Time	The time elapsed, in seconds, after the request leaves the load balancer until a response from the target is received. This is equivalent to the target_processing_time field in the access logs. For now we are not collecting data from amazon api, because it is not supported yet for Network ELB associated with Target Group.
TLS Negotiation Error Count per Target Group	The number of TLS connections initiated by the load balancer that did not establish a session with the target. Possible causes include a mismatch of ciphers or protocols. For now we are not collecting data from amazon api, because it is not supported yet for Network ELB associated with Target Group.
Unhealthy Host Count (Average)	The number of targets that are considered unhealthy.
Unhealthy Host Count (Maximum)	The number of targets that are considered unhealthy.
Unhealthy Host Count (Minimum)	The number of targets that are considered unhealthy.

AWS EMR Service

AWS: EMR Instance Configuration	
Object Name	Object Description
AWS EMR/EC2	The unique identifier set for the EMR-EC2 relationship.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
EMR Nodes as EC2 Instances	The group label for the EMR-EC2 relationship.
Instance ID	The unique identifier of EMR master instance.
Name	The name of the cluster.
State	The current state of the cluster.

AWS: EMR Instance Performance	
Object Name	Object Description
Apps Completed	The number of applications submitted to YARN that have completed.

Apps Failed	The number of applications submitted to YARN that have failed to complete.
Apps Killed	The number of applications submitted to YARN that have been killed.
Apps Pending	The number of applications submitted to YARN that are in a pending state.
Apps Running	The number of applications submitted to YARN that are in a running state.
Apps Submitted	The number of applications submitted to YARN.
Capacity Remaining (GB)	The amount of remaining HDFS disk capacity.
Container Allocated	The number of resource containers allocated by the ResourceManager.
Container Pending	The number of containers in the queue that have not yet been allocated.
Container Pending Ratio	The ratio of pending containers to containers allocated ($\text{ContainerPendingRatio} = \text{ContainerPending} / \text{ContainerAllocated}$). If $\text{ContainerAllocated} = 0$, then $\text{ContainerPendingRatio} = \text{ContainerPending}$. The value of $\text{ContainerPendingRatio}$ represents a number, not a percentage. This value is useful for scaling cluster resources based on container allocation behavior.
Container Reserved	The number of containers reserved.
Core Nodes Pending	The average number of pending nodes in the run.
Core Nodes Running	The average number of core nodes running for this run.
Corrupt Blocks	The number of blocks that HDFS reports as corrupted.
Dfs Pending Replication Blocks	The status of block replication: blocks being replicated, age of replication requests, and unsuccessful replication requests.
HBase Backup Failed	Whether the last backup failed. This is set to 0 by default and updated to 1 if the previous backup attempt failed. This metric is only reported for HBase clusters. Use case: Monitor HBase backups Units: Count

HBase Most Recent Backup Duration	The amount of time it took the previous backup to complete. This metric is set regardless of whether the last completed backup succeeded or failed. While the backup is ongoing, this metric returns the number of minutes after the backup started. This metric is only reported for HBase clusters.
HBase Time Since Last Successful Backup	The number of elapsed minutes after the last successful HBase backup started on your cluster. This metric is only reported for HBase clusters.
HDFS Bytes Read	The average number of bytes read from all Hadoop filesystems.
HDFS Bytes Written	The average number of bytes written to the Hadoop filesystems.
HDFS Utilization	The percentage utilization of the available Hadoop filesystems.
Is Idle	Whether or not this instance is idle at the moment.
Jobs Failed	The number of jobs failed (if any) during the current run.
Jobs Running	The number of current actively running jobs in the current run.
Live Data Nodes	The number of active data nodes with IO for the current run.
Live Task Trackers	The Percentage of active task tracking nodes for the current run.
Map Reduce Lost Nodes	The number of nodes allocated to MapReduce that have been marked in a LOST state.
Map Slots Open	The number of map slots currently open.
MapReduce Active Nodes	The number of nodes presently running MapReduce tasks or jobs. Equivalent to YARN metric
MapReduce Decommissioned Nodes	The number of nodes allocated to MapReduce applications that have been marked in a DECOMMISSIONED state.
MapReduce Rebooted Nodes	The number of nodes available to MapReduce that have been rebooted and marked in a REBOOTED state.
MapReduce Total Nodes	The number of nodes presently available to MapReduce jobs.
MapReduce Unhealthy Nodes	The number of nodes available to MapReduce jobs marked in an UNHEALTHY state.

Memory Allocated (MB)	The amount of memory allocated to the cluster.
Memory Available (MB)	The amount of memory available to be allocated.
Memory Reserved (MB)	The amount of memory reserved.
Memory Total (MB)	The total amount of memory in the cluster.
Missing Blocks	The current number of missing blocks.
Pending Deletion Blocks	The number of blocks marked for deletion.
Reduce Slots Open	The number of open reduction slots available.
Remaining Map Tasks	The number of remaining map tasks for this run.
Remaining Map Tasks Per Slot	The average number of remaining map tasks per available slot.
Remaining Reduce Tasks	The number of remaining reduce tasks for this run.
Running Map Tasks	The number of currently running map tasks for this run.
Running Reduce Tasks	The number of currently running reduce tasks for this run.
S3 Bytes Read	The number of Simple Storage Service (S3) bytes read during this run.
S3 Bytes Written	The number of Simple Storage Service (S3) bytes written during this run.
Task Nodes Pending	The number of pending (inactive) task nodes at the moment.
Task Nodes Running	The number of task nodes running at the moment.
Total Load	The total percentage load across all nodes at the moment.
Under Replicated Blocks	The number of blocks that need to be replicated one or more times.
YARN Memory Available Percentage	The percentage of remaining memory available to YARN ($\text{YARNMemoryAvailablePercentage} = \text{MemoryAvailableMB} / \text{MemoryTotalMB}$). This value is useful for scaling cluster resources based on YARN memory usage.

AWS: EMR Service Health

Object Name	Object Description
Date	The timestamp when this health status was written.

Health	A text description of the health status of this AWS service.
--------	--

AWS Glacier Service

AWS: Glacier Instance Configuration	
Object Name	Object Description
# of Archives	The number of archives in this vault.
ARN	The unique identifier (ARN) of the vault.
AWS Glacier/SNS	DCMR Relation
AWS: SNS Relationship	Label for the DCMR relationship
Creation Time	The datetime when the vault was created.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Job Type Notifications	A list of job types which trigger notifications.
Key	The tag key. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment.
Last Updated	The most recent, if any, date the inventory of the vault was modified.
Name	The name of the vault.
Notifications	The SNS Topics, if any, which publish events related to this vault.
Region	The region of the vault.
Size	The size of the vault.
Tags	Label for glacier tags.
Value	The tag value. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment.

AWS: Glacier Service Health	
-----------------------------	--

Object Name	Object Description
Date	A timestamp when the health status of this service was originally written.
Health	A text description of the health status of this AWS service.

AWS IoT Service

AWS: IoT Service Performance

Object Name	Object Description
Connect.Success	The number of successful connections to the message broker.
GetThingShadow.Accepted	The number of GetThingShadow requests processed successfully.
Ping.Success	The number of ping messages received by the message broker.
Subscribe.Success	The number of subscribe requests that were successfully processed by the message broker.
UpdateThingShadow.Accepted	The number of UpdateThingShadow requests processed successfully.

AWS: IoT Thing Instance Configuration

Object Name	Object Description
Attribute Keys	The key of the thing attributes assigned to the resource.
Attribute Values	The value of the thing attributes assigned to the resource.
Billing Group	The name of the billing group the thing belongs to.
Key	The key of the tags assigned to the resource.
Thing Name	The name of the thing.
Thing ARN	The ARN of the thing to describe.
Thing Id	The ID of the thing to describe.
Thing Type Name	The thing type name.

Value	The value of the tags assigned to the resource.
Version	The current version of the thing record in the registry.

AWS KMS Service

AWS: KMS Configuration	
Object Name	Object Description
Key Id	The unique identifier of every Key from Key Management Service.
Target Key Id	String that contains the key identifier referred to by the alias.
Target Key Id	The key identifier which referred to by the tags.
Alias	String that contains the alias. This value begins with alias/.
ARN	The Amazon Resource Name (ARN) of every Key from Key Management Service.
Creation Date	The date and time when the Key was created.
Key Manager	The manager of the Key. In AWS are either Customer Managed or AWS managed.
Origin	The source of the Key. They could be: AWS_KMS, EXTERNAL, AWS_CLOUDHSM.
Status	The state of the Key.
Tag Key	The key of the tag.
Tag Value	The value of the tag.

AWS: KMS Performance	
Object Name	Object Description
Seconds Until Key Material Expiration	This metric tracks the number of seconds remaining until imported key material expires. This metric is valid only for CMKs whose origin is EXTERNAL and whose key material is or was set to expire.

AWS Lambda Service

AWS: Lambda Function Configuration	
Object Name	Object Description
Name	The alias name.
State	The state of the event source mapping.
State Transition Reason	The reason the event source mapping is in its current state.
Description	The alias description.
Last Processing Result	The result of the last AWS Lambda invocation of your Lambda function.
Version	List of versions of the Lambda function.
Action	The action that the trigger makes to Lambda Function.
DLQ Resource	The Dead-Letter Queue service. Possible values are "SNS" or "SQS".
Event Source ARN	The Amazon Resource Name (ARN) of the Amazon Kinesis stream that is the source of events.
Function Version	The function version to which the alias points.
Service	The Amazon Service that triggers the Lambda Function.
Version Description	Descriptions of each version of the Lambda function.
ARN	Lambda function ARN that is qualified using the alias name as the suffix.
AWS Lambda Function/Security Group	A list of security group IDs associated with the Lambda function.
AWS Lambda Function/SNS-SQS	The Id of the SQS or SNS service that the lambda is using for the error handling storage.
AWS Lambda Function/Subnet	A list of subnet IDs associated with the Lambda function.
AWS Lambda Function/VPC	The custom VPC that the lambda belongs to.
Batch Size	The largest number of records that AWS Lambda will retrieve from your event source at the time of invoking your function.

Code Size (B)	The size, in bytes, of the function .zip file you uploaded.
Description	The user-provided description for the lambda function.
Event Source ARN	The Amazon Resource Name (ARN) of the Amazon Service that is the source of triggers.
Execution Role	The Amazon Resource Name (ARN) of the IAM role that Lambda assumes when it executes your function to access any other Amazon Web Services (AWS) resources.
Function ARN	Lambda Function Amazon Resource Name identifier.
Function Name	The name of the function.
Handler	The function Lambda calls to begin executing your function.
Key	The key of the lambda function tag.
Last Modified	The time stamp of the last time you updated the function.
Last Modified	The UTC time string indicating the last time the event mapping was updated.
Memory Size (MB)	The memory size, in MB, you configured for the function. Must be a multiple of 64 MB.
Runtime	The runtime environment for the Lambda function.
Timeout (s)	The function execution time in seconds at which Lambda should terminate the function. Because the execution time has cost implications, we recommend you set this value based on your expected execution time. The default is 3 seconds.
Value	The value of the lambda function tag.
Version	The version of the Lambda function.
Version ARN	List of ARNs with the detail of versions of the Lambda function.

AWS: Lambda Function Performance

Object Name	Object Description
Dead Letter Errors	Incremented when Lambda is unable to write the failed event payload to your configured Dead Letter Queues. The major event expires after 90 minutes.

Duration	Elapsed wall clock time from when the function code starts executing as a result of an invocation to when it stops executing.
Errors	Number of invocations that failed due to errors in the function (response code 4XX).
Invocations	Number of times a function is invoked in response to an event or invocation API call.
Iterator Age	Measures the age of the last record for each batch of records processed. Age is the difference between the time Lambda received the batch, and the time the last record in the batch was written to the stream.
Throttles	Number of Lambda function invocation attempts that were throttled due to invocation rates exceeding the customer's concurrent limits (error code 429).

AWS: Lambda Function Qualified Configuration

Object Name	Object Description
Alias ARN	Lambda function ARN that is qualified using the alias name as the suffix.
Alias Description	Alias description.
Alias Function Version	Function version to which the alias points.
Alias Name	Alias name.
Alias Revision ID	Represents the latest updated revision of the function or alias.
AWS Lambda Function Qualified/Security Groups	A list of security group IDs associated with the Lambda function.
AWS Lambda Function Qualified/Subnets	A list of subnet IDs associated with the Lambda function.
AWS Lambda Function Qualified/VPC	The custom VPC that the lambda belongs to.
Version Code Sha256	It is the SHA256 hash of your function deployment package.
Version Code Size (B)	The size, in bytes, of the function .zip file you uploaded.
Version Description	The user-provided description for lambda function version.
Version Function ARN	The Amazon Resource Name (ARN) assigned to a regular lambda function.

Version Function ARN Version	The Amazon Resource Name (ARN) assigned to the lambda function version.
Version Function Name	The name of the function. Note that the length constraint applies only to the ARN. If you specify only the function name, it is limited to 64 characters in length.
Version Handler	The function Lambda calls to begin executing your function.
Version Last Modified	The time stamp of the last time you updated the function.
Version Memory Size (MB)	The memory size, in MB, you configured for the function. Must be a multiple of 64 MB.
Version Revision ID	Represents the latest updated revision of the function or alias.
Version Role	The Amazon Resource Name (ARN) of the IAM role that Lambda assumes when it executes your function to access any other Amazon Web Services (AWS) resources.
Version Runtime	The runtime environment for the Lambda function.
Version Timeout (s)	The function execution time at which Lambda should terminate the function. Because the execution time has cost implications, we recommend you set this value based on your expected execution time. The default is 3 seconds.
Version Version	The version of the Lambda function.
VPC Config Security Group ID	A list of security group IDs associated with the Lambda function.
VPC Config Subnet ID	A list of subnet IDs associated with the Lambda function.
VPC Config VPC ID	The VPC ID associated with you Lambda function.

AWS: Lambda Function Qualified Performance

Object Name	Object Description
Dead Letter Errors	Incremented when Lambda is unable to write the failed event payload to your configured Dead Letter Queues. The major event expires after 90 minutes.

Duration	Elapsed wall clock time from when the function code starts executing as a result of an invocation to when it stops executing.
Errors	Number of invocations that failed due to errors in the function (response code 4XX).
Invocations	Number of times a function is invoked in response to an event or invocation API call.
Iterator Age	Measures the age of the last record for each batch of records processed. Age is the difference between the time Lambda received the batch, and the time the last record in the batch was written to the stream.
Throttles	Number of Lambda function invocation attempts that were throttled due to invocation rates exceeding the customer's concurrent limits (error code 429).

AWS: Lambda Function Replica Configuration

Object Name	Object Description
Code Size (B)	The size, in bytes, of the function .zip file you uploaded.
Description	The user-provided description for the lambda function.
Execution Role	The Amazon Resource Name (ARN) of the IAM role that Lambda assumes when it executes your function to access any other Amazon Web Services (AWS) resources.
Function ARN	Lambda Function Amazon Resource Name identifier.
Function Name	The name of the function.
Handler	The function Lambda calls to begin executing your function.
Lambda Replica Master ARN	The ARN (Amazon Resource Name) of the master function from which the function was replicated.
Last Modified	The time stamp of the last time you updated the function.
Memory Size (MB)	The memory size, in MB, you configured for the function. Must be a multiple of 64 MB.
Parent (Master) Lambda Function/Replica Lambda Function	The master function from which the function was replicated.
Runtime	The runtime environment for the Lambda function.

Timeout (s)	The function execution time in seconds at which Lambda should terminate the function. Because the execution time has cost implications, we recommend you set this value based on your expected execution time. The default is 3 seconds.
Version	The version of the Lambda function.

AWS: Lambda Service Configuration

Object Name	Object Description
Code Storage (B)	Total size, in bytes, of the deployment packages of your account per region.
Memory Allocated (B)	Memory allocated for all lambda functions.
Total Code Size (B)	Maximum size, in bytes, of a code package you can upload per region. The default size is 75 GB.
Code Size Unzipped (B)	Size, in bytes, of code/dependencies that you can zip into a deployment package (uncompressed zip/jar size) for uploading. The default limit is 250 MB.
Code Size Zipped (B)	Size, in bytes, of a single zipped code/dependencies package you can upload for your Lambda function (.zip/.jar file). Try using Amazon S3 for uploading larger files. Default limit is 50 MB.
Code Storage Percentage	Total size, in bytes, of the deployment packages of your account per region, it is measured in percentage.
Full Account Concurrency	Number of simultaneous executions of your function per region. The default limit is 1000.
Lambda Functions	The number of existing functions of your account per region.
Unreserved Account Concurrency	The number of concurrent executions available to functions that do not have concurrency limits set.

AWS: Lambda Service Health

Object Name	Object Description
Date	The timestamp of this health status update string.
Health	A text description of this AWS Lambda Service health status.

AWS: Lambda Service Performance

Object Name	Object Description
Concurrent Executions	Sum for all functions within this service instance, during this polling interval, of concurrent executions of functions.
Duration	An average of the elapsed wall clock time from when the function code starts executing as a result of an invocation to when it stops executing. This averaged time value is for all functions within this service instance during this polling interval.
Errors	Sum for all functions within this service instance, during this polling interval, of invocations that failed due to errors (response code 4XX).
Invocations	Sum for all functions within this service instance, during this polling interval, of the times functions are invoked in response to an event or invocation API call.
Throttles	Sum for all functions within this service instance, during this polling interval, of function invocation attempts that were throttled due to invocation rates exceeding the customer's concurrent limits (error code 429).
Unreserved Concurrent Executions	Sum for all functions within this service instance, during this polling interval, for functions invoked or running in concurrency for functions that have no custom concurrency limit specified.

AWS LightSail Service

AWS: LightSail Instance Configuration

Object Name	Object Description
Access Direction	The access direction (inbound or outbound).
Access From	The location from which access is allowed (e.g., Anywhere (0.0.0.0/0)).
Access Type	The type of access (Public or Private).
Availability Zone	The Availability Zone.
Blueprint ID	The blueprint ID.

Blueprint Name	The friendly name of the blueprint.
CPU count	The number of vCPUs the instance has.
Created At	The timestamp when the instance was created.
Disk ARN	The Amazon Resource Name (ARN) of the disk.
Disk Attached To	The resources to which the disk is attached.
Disk Attachment State	The attachment state of the disk.
Disk Created At	The date when the disk was created.
Disk Name	The name of the disk.
Disk Path	The disk path.
Disk Resource Type	The resource type of the disk.
Disk Size in GB	The size of the disk in GB.
Distinguished Name	The Amazon Resource Name (ARN) of the instance.
From Port	The first port in the range.
GB per Month Allocation	The amount allocated per month (in GB).
Instance Type	The Amazon EC2 instance type. Note: This collection object may not collect any data if AWS has deprecated the bundle that was used to create the original lightsail instance.
IPv6 Address	The IPv6 address of the instance.
Is Static Ip?	A Boolean value indicating whether this instance has a static IP assigned to it.
Name	The name the user gave the instance.
Price	The price in US dollars. Note: This collection object may not collect any data if AWS has deprecated the bundle that was used to create the original lightsail instance.
Private IP Address	The private IP address of the instance.
Protocol	The protocol being used, it can contain following values tcp -1 udp. The -1 value means all protocols (udp and udp).
Public IP Address	The public IP address of the instance.
RAM size in GB	The amount of RAM in GB on the instance (e.g., 1.0).
Region Name	The AWS Region name.
Resource Type	The type of resource (usually Instance).

State	The status code and the state (e.g., running) for the instance.
Support Code	The support code. Include this code in your email to support when you have questions about an instance or another resource in Lightsail. This code enables our support team to look up your Lightsail information more easily.
System Disk	A Boolean value indicating whether this disk is a system disk (has an operating system loaded on it).
To Port	The last port in the range.

AWS: LightSail Instance Performance

Object Name	Object Description
CPU Utilization	This metric refers to the percentage of CPU capacity that your Amazon Lightsail instance is using at the timestamp. This metric identifies the processing power required to run an application on a selected instance.
Network In	The number of bytes received at the timestamp on all network interfaces by the Amazon Lightsail instance. This metric identifies the volume of incoming network traffic to an application on a single instance.
Network Out	The number of bytes sent out at the timestamp on all network interfaces by the Amazon Lightsail instance. This metric identifies the volume of outgoing network traffic to an application on a single instance.
Status Check Failed	Reports whether the instance has passed both the Amazon lightsail instance status check and the system status check in the last minute. This metric can be either 0 (passed) or 1 (failed).
Status Check Failed Instance	Reports whether the instance has passed the Amazon Lightsail instance status check in the last minute. This metric can be either 0 (passed) or 1 (failed).
Status Check Failed System	Reports whether the instance has passed the system status check in the last minute. This metric can be either 0 (passed) or 1 (failed).

AWS Network ELB Service

AWS: Network ELB Instance Configuration

Object Name	Object Description
AWS Network ELB/Availability Zones	The availability zones for the load balancer.
AWS Network ELB/Target Groups	The unique identifiers of the target groups for the network load balancer.
AWS Network ELB/VPC	The unique identifier of the VPC for the load balancer.
DNS Name	The public DNS name of the network load balancer.
Key	The key of the tag, that belongs to the network load balancer.
Listener ARN	The Amazon Resource Name (ARN) of the listener.
Listener Port	The port on which the load balancer is listening.
Listener Protocol	The protocol for connections from clients to the load balancer.
Load Balancer ARN	The Amazon Resource Name (ARN) of the load balancer.
Load Balancer Name	The name of the load balancer.
Load Balancer State	The state of the load balancer.
Scheme	The DNS name of an Internet-facing load balancer is publicly resolvable to the public IP addresses of the nodes.
Type	The type of load balancer.
Value	The value of the tag, that belongs to the network load balancer.

AWS: Network ELB Instance Performance

Object Name	Object Description
Active Flow Count	The total number of concurrent TCP flows (or connections) from clients to targets.

Active Flow Count TLS	The total number of concurrent TLS flows (or connections) from clients to targets. This metric includes only connections in the ESTABLISHED states.
Client TLS Negotiation Error Count	The total number of TLS handshakes that failed during negotiation between a client and a TLS listener.
Consumed LCUst	The number of load balancer capacity units (LCU) used.
New Flow Count	The total number of new TCP flows (or connections) established from clients to targets in the time period.
New Flow Count TLS	The total number of new TLS flows (or connections) established from clients to targets in the time period.
Processed Bytes	The total number of bytes processed by the load balancer, including TCP/IP headers.
Processed Bytes TLS	The total number of bytes processed by TLS listeners.
Target TLS Negotiation Error Count	The total number of TLS handshakes that failed during negotiation between a TLS listener and a target.
TCP Client Reset Count	The total number of reset (RST) packets sent from a client to a target.
TCP ELB Reset Count	The total number of reset (RST) packets generated by the load balancer.
TCP Target Reset Count	The total number of reset (RST) packets sent from a target to a client.

AWS OpsWorks Service

AWS: OpsWorks Instance Configuration	
Object Name	Object Description
AWS OpsWorks/EC2	The unique identifier for the OpsWorks-EC2 relationship.
Chef Version	The version of Chef used to deploy the stack.
Default Operating System	The default Operating System of the Stack.
Default Root Device Type	The type of the root file system.
Default SSH Key	The SSH keyname associated with the key-pair of the stack.

Default Subnet	The default subnet of the Stack.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Hostname Theme	The default naming scheme, if any, for the instances of the stack.
Id	The unique identifier of the Stack.
Name	The common name of the Stack.
Region	The region where the stack is deployed.
Use Custom Chef Cookbooks	Whether or not the stack was deployed with a custom Chef cookbook.
Use OpsWorks Security Groups	Whether or not to use OpsWorks Security Groups for instances within the stack.
VPC	The virtual private cloud, if any, associated with the Stack.

AWS: OpsWorks Service Health

Object Name	Object Description
Date	A timestamp when this service's health status was originally written.
Health	A text description of the health status of this AWS service.

AWS: OpsWorks Stack Performance

Object Name	Object Description
Active Processes	The number of active processes.
CPU Idle	The percentage of time the CPU is idle.
CPU Nice	The percentage of time that the CPU is handling processes with a positive nice value, which have lower scheduling priority. For further information, see nice (Unix).
CPU Steal	The percentage of time that an instance is waiting for the hypervisor to allocate physical CPU resources.

CPU System	The percentage of time that the CPU is handling system operations.
CPU User	The percentage of time the CPU is handling user operations.
CPU Wait I/O	The percentage of time the CPU is waiting on I/O operations.
Load 1 Minute	The Unix load average over a 1 minute window.
Load 15 Minutes	The Unix load averaged over a 15 minute window.
Load 5 Minutes	The Unix load averaged over a 5 minute window.
Memory Buffers	The amount of buffered memory.
Memory Cached	The amount of cached memory.
Memory Free	The amount of free memory.
Memory Swap	The amount of swap space used.
Memory Total	The total amount of memory.
Memory Used	The total amount of memory used.

AWS RDS Service

AWS: RDS Aurora Cluster Instance Configuration	
Object Name	Object Description
Allocated Storage	The amount of allocated storage available for a database cluster. For AWS Aurora clusters this will always return a value of 1 as Aurora DB Cluster sizes are not fixed.
Availability Zones	Availability zones where database instances in the cluster can be created.
Backup Retention Period	Number of days which automatic DB snapshots are stored.
Capacity	Current capacity active AWS Aurora Serverless DB Clusters. Will return 0 when cluster is paused.
Cluster Creation Time	When the DB cluster was created in UTC
Custom Endpoints	Custom endpoints associated with the cluster.

Database Name	The name of the Database Cluster that was provided at creation time.
DB Cluster ARN	The ARN associated with the DB cluster.
DB Cluster Identifier	DBClusterIdentifier provided for the DB Cluster
DB Cluster Parameter Group	Name of the DB Cluster Parameter Group.
DB Cluster Resource ID	Region unique immutable identifier for the cluster.
DB Subnet Group	Specifies information about the DB Clusters subnet group.
Deletion Protection	Specifies if deletion protection is enabled.
Endpoint	Connection endpoint for the primary instance of the DB Cluster
Engine	Name of the database engine used for the DB cluster.
Engine Mode	Mode of the DB engine for the cluster.
Engine Version	Version of the database engine used for the DB cluster.
Hosted Zone ID	Specifies the ID that Amazon Route 53 assigns when you create a hosted zone.
IAM Database Authentication Enabled	Specifies if IAM database authentication is enabled.
Key	Tag key value.
KMS Key ID	If database is encrypted the KMS Key identifier for the encrypted DB cluster.
Master Username	Master username for the DB cluster.
Multiple Availability Zones	Identifies if the database cluster has instances in multiple availability zones.
Port	The port that the database engine is listening on.
Read Replica Identifier	Contains one or more of the identifiers for the read replicas associated with the DB cluster.
Reader Endpoint	Reader endpoint for the DB Cluster.
Replication Source Identifier	Contains the identifier of the source DB cluster if the DB cluster is a read replica.
Status	The status of the database cluster.
Storage Encrypted	Specifies if the DB Cluster is encrypted.
Value	Tag value.

AWS: RDS Aurora Cluster Instance Performance

Object Name	Object Description
Aborted Clients	The number of client connections that have not been closed properly.
Active Transactions	The average number of current transactions executing on an Aurora database instance per second.
Aurora Bin Log Replica Lag	The amount of time a replica DB cluster running on Aurora with MySQL compatibility lags behind the source DB cluster.
Aurora Global DB Data Transfer Bytes	The amount of redo log data transferred from the master AWS Region to a secondary AWS Region.
Aurora Global DB Replicated Write IO	The number of write I/O operations replicated from the primary AWS Region to the cluster volume in a secondary AWS Region.
Aurora Global DB Replication Lag	The amount of lag when replicating updates from the primary AWS Region.
Aurora Parallel Query Bytes Returned	The number of bytes for the tuple data structures transmitted to the head node during parallel queries.
Aurora Parallel Query Max Concurrent Requests	The maximum number of parallel query sessions that can run concurrently on this Aurora DB instance.
Aurora Parallel Query Pages Pushed Down	The number of data pages (each with a fixed size of 16 KiB) where parallel query avoided a network transmission to the head node.
Aurora Parallel Query Request Attempted	The number of parallel query sessions requested.
Aurora Parallel Query Request Executed	The number of parallel query sessions run successfully.
Aurora Parallel Query Request Failed	The number of parallel query sessions that returned an error to the client.
Aurora Parallel Query Request In Progress	The number of parallel query sessions currently in progress.
Aurora Parallel Query Request Not Chosen	The number of times parallel query was not chosen to satisfy a query.
Aurora Parallel Query Request Not Chosen Below Min Rows	The number of times parallel query was not chosen due to the number of rows in the table.
Aurora Parallel Query Request Not Chosen Column Bit	The number of parallel query requests that use the nonparallel query processing path because of an unsupported data type in the list of projected columns.

Aurora Parallel Query Request Not Chosen Column Geometry	The number of parallel query requests that use the nonparallel query processing path because the table has columns with the GEOMETRY data type.
Aurora Parallel Query Request Not Chosen Column Lob	The number of parallel query requests that use the nonparallel query processing path because the table has columns with a LOB data type, or VARCHAR columns that are stored externally due the declared length.
Aurora Parallel Query Request Not Chosen Column Virtual	The number of parallel query requests that use the nonparallel query processing path because the table contains a virtual column.
Aurora Parallel Query Request Not Chosen Custom Charset	The number of parallel query requests that use the nonparallel query processing path because the table has columns with a custom character set.
Aurora Parallel Query Request Not Chosen Fast DDL	The number of parallel query requests that use the nonparallel query processing path because the table is currently being altered by a fast DDL ALTER statement.
Aurora Parallel Query Request Not Chosen Full Text Index	The number of parallel query requests that use the nonparallel query processing path because the table has full-text indexes.
Aurora Parallel Query Request Not Chosen High Buffer Pool Pct	The number of times parallel query was not chosen because a high percentage of the table data (currently, greater than 95 percent) was already in the buffer pool.
Aurora Parallel Query Request Not Chosen Index Hint	The number of parallel query requests that use the nonparallel query processing path because the query includes an index hint.
Aurora Parallel Query Request Not Chosen InnoDB Table Format	The number of parallel query requests that use the nonparallel query processing path because the table uses an unsupported InnoDB row format.
Aurora Parallel Query Request Not Chosen Long Trx	The number of parallel query requests that used the nonparallel query processing path, due to the query being started inside a long-running transaction.
Aurora Parallel Query Request Not Chosen No Where Clause	The number of parallel query requests that use the nonparallel query processing path because the query does not include any WHERE clause.
Aurora Parallel Query Request Not Chosen Range Scan	The number of parallel query requests that use the nonparallel query processing path because the query uses a range scan on an index.
Aurora Parallel Query Request Not Chosen Row Length Too Long	The number of parallel query requests that use the nonparallel query processing path because the total combined length of all the columns is too long.

Aurora Parallel Query Request Not Chosen Small Table	The number of times parallel query was not chosen due to the overall size of the table, as determined by number of rows and average row length.
Aurora Parallel Query Request Not Chosen Temporary Table	The number of parallel query requests that use the nonparallel query processing path because the query refers to temporary tables that use the unsupported MyISAM or memory table types.
Aurora Parallel Query Request Not Chosen Tx Isolation	The number of parallel query requests that use the nonparallel query processing path because query uses an unsupported transaction isolation level.
Aurora Parallel Query Request Not Chosen Unsupported Access	The number of parallel query requests that use the nonparallel query processing path because the WHERE clause does not meet the criteria for parallel query.
Aurora Parallel Query Request Not Chosen Update Delete Stmt	The number of parallel query requests that use the nonparallel query processing path because the query is part of an UPDATE or DELETE statement.
Aurora Parallel Query Request Throttled	The number of times parallel query was not chosen due to the maximum number of concurrent parallel queries already running on a particular Aurora DB instance.
Aurora PQ Request Not Chosen Few Pages Outside Buffer Pool	The number of times parallel query was not chosen, even though less than 95 percent of the table data was in the buffer pool, because there was not enough unbuffered table data to make parallel query worthwhile.
Aurora Replica Lag	For an Aurora replica, the amount of lag when replicating updates from the primary instance.
Aurora Replica Lag Maximum	The maximum amount of lag between the primary instance and each Aurora DB instance in the DB cluster.
Aurora Replica Lag Minimum	The minimum amount of lag between the primary instance and each Aurora DB instance in the DB cluster.
Aurora Volume Bytes Left Total	The remaining available space for the cluster volume. As the cluster volume grows, this value decreases. If it reaches zero, the cluster reports an out-of-space error.
Backtrack Change Records Creation Rate	The number of backtrack change records created over 5 minutes for your DB cluster.
Backtrack Change Records Stored	The number of backtrack change records used by your DB cluster.

Backtrack Window Actual	The difference between the target backtrack window and the actual backtrack window.
Backtrack Window Alert	The number of times that the actual backtrack window is smaller than the target backtrack window for a given period of time.
Backup Retention Period Storage Used	The total amount of backup storage used to support the point-in-time restore feature within the Aurora DB cluster's backup retention window.
Bin Log Disk Usage	The amount of disk space occupied by binary logs on the primary instance.
Blocked Transactions	The average number of transactions in the database that are blocked per second.
Buffer Cache Hit Ratio	The percentage of requests that are served by the buffer cache.
Commit Latency	The average duration of commit operations.
Commit Throughput	The average number of commit operations per second.
CPU Credit Balance	The number of CPU credits that an instance has accumulated, reported at 5-minute intervals. This metric applies only to db.t2.small and db.t2.medium instances for Aurora MySQL, and to db.t3 instances for Aurora PostgreSQL.
CPU Credit Usage	The number of CPU credits consumed during the specified period, reported at 5-minute intervals. This metric applies only to db.t2.small and db.t2.medium instances for Aurora MySQL, and to db.t3 instances for Aurora PostgreSQL.
CPU Utilization	The percentage of CPU used by an Aurora DB instance.
Database Connections	The current number of connections to an Aurora DB instance.
DDL Latency	The average duration of requests such as example, create, alter, and drop requests.
DDL Throughput	The average number of DDL requests per second.
Deadlocks	The average number of deadlocks in the database per second.
Delete Latency	The average duration of delete operations.
Delete Throughput	The average number of delete queries per second.

Disk Queue Depth	The number of outstanding read/write requests waiting to access the disk.
DML Latency	The average duration of inserts, updates, and deletes.
DML Throughput	The average number of inserts, updates, and deletes per second.
Engine Uptime	The amount of time that the instance has been running.
Free Local Storage	The amount of local storage available.
Freeable Memory	The amount of available random access memory.
Insert Latency	The average duration of insert operations.
Insert Throughput	The average number of insert operations per second.
Login Failures	The average number of failed login attempts per second.
Maximum Used Transaction IDs	The age of the oldest unvacuumed transaction ID, in transactions.
Network Receive Throughput	The amount of network throughput received from clients by each instance in the Aurora MySQL DB cluster.
Network Throughput	The amount of network throughput both received from and transmitted to clients by each instance in the Aurora MySQL DB cluster.
Network Transmit Throughput	The amount of network throughput sent to clients by each instance in the Aurora DB cluster.
Num Binary Log Files	The number of binlog files generated.
Queries	The average number of queries executed per second.
Read IOPS	The average number of disk I/O operations per second.
Read Latency	The average amount of time taken per disk I/O operation.
Read Throughput	The average number of bytes read from disk per second.
Result Set Cache Hit Ratio	The percentage of requests that are served by the Resultset cache.
Rollback Segment History List Length	The undo logs that record committed transactions with delete-marked records.
Row Lock Time	The total time spent acquiring row locks for InnoDB tables.

Select Latency	The average amount of time for select operations.
Select Throughput	The average number of select queries per second.
Serverless Database Capacity	The current capacity of an Aurora Serverless v1 DB cluster.
Snapshot Storage Used	The total amount of backup storage consumed by all Aurora snapshots for an Aurora DB cluster outside its backup retention window.
Sum Binary Log Size	The total size of the binlog files.
Swap Usage	The amount of swap space used. This metric is available for the Aurora PostgreSQL instance classes db.t3.medium, db.r5.large, db.r5.xlarge, db.r4.large, and db.r4.xlarge. For Aurora MySQL, this metric applies only to db.t* instances.
Total Backup Storage Billed	The total amount of backup storage in bytes for which you are billed for a given Aurora DB cluster.
Transaction Logs Disk Usage	The amount of disk space consumed by transaction logs on the Aurora PostgreSQL DB instance.
Update Latency	The average amount of time taken for update operations.
Update Throughput	The average number of updates per second.
Volume Bytes Used	The amount of storage used by your Aurora DB instance.
Volume Read IOPs	The number of billed read I/O operations from a cluster volume within a 5-minute interval.
Volume Write IOPs	The number of write disk I/O operations to the cluster volume, reported at 5-minute intervals.
Write IOPS	The average number of disk I/O operations per second.
Write Latency	The average amount of time taken per disk I/O operation.
Write Throughput	The average number of bytes written to persistent storage every second.

AWS: RDS Instance Configuration

Object Name	Object Description
DB Subnet Group Name	The name of the DB subnet group.

Is Clustered	Boolean value to know if a cluster has members or not.
RDS Instance/RDS Aurora Cluster	The Amazon Resource Name (ARN) for the DB cluster.
VPC ID	For instances associated with a Virtual Private Cloud (VPC), this is the unique identifier of the VPC.
Created Time	The timestamp the instance was created.
Database Class	The size of the database. For example: db.t1.micro, db.m2.2xlarge, db.m5.xlarge, etc.
DB Cluster Parameter Group Status	The status of the DB cluster parameter group for this member of the DB cluster.
DB Default Max Connections	The default maximum number of simultaneous database connections varies by the DB engine type and the memory allocation for the DB instance class.
DB Instance Identifier	The unique identifier for this instance.
DB Parameter Groups	Provides the list of DB parameter groups applied to this DB instance.
DB Security Group Name	Security group associated with RDS instance.
DB Security Group Status	Security group description associated with RDS instance.
DB Subnet Group Description	Provides the description of the DB subnet group.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Engine Name	A high-level name for the database engine such as MySQL, or SQLServer.
Is Cluster Writer	Value that is true if the cluster member is the primary instance for the DB cluster and false otherwise.
Key	The tag key. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for Amazon EC2 instances of your account that helps you track the stack level and owner of each instance. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.

Multi-AZ	A boolean specifying whether or not this instance exists in more than one zone.
Pending Values	A list of pending events such as replication, class change, or group changes.
Primary Zone	The primary availability zone where this instance is addressable.
Promotion Tier	Value that specifies the order in which an Aurora Replica is promoted to the primary instance after a failure of the existing primary instance.
Status	Whether this instance is terminated, starting, stopped, running, or replicating.
Storage Size	The file system size of the local instance used for table spaces.
Value	The tag value. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for Amazon EC2 instances of your account that helps you track the stack level and owner of each instance. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.
VPC Security Group ID	The unique identifier of the VPC security group.
VPC Security Group Name	The name of the VPC security group.
VPC Security Group Status	The status of the VPC security group.

AWS: RDS Instance Performance

Object Name	Object Description
Allocated Storage	Specifies the allocated storage size specified in gibibytes.
Bin Log Disk Usage	The bytes consumed by the binary log of RDS instance.
CPU Credit Balance	The number of CPU credits available for the instance to burst beyond its base CPU utilization.
CPU Credit Usage	The number of CPU credits consumed by the instance.

CPU Utilization	The utilization percentage across the virtualized CPU cores of database.
Database Connections	The number of current database connections.
Disk Queue Depth	The number of outstanding IOs (read/write requests) waiting to access the disk.
Free Storage Space	The available free storage space on the mounted filesystem of database.
Freeable Memory	The available memory which has yet to be freed by the the processes of database.
Max Database Connections	The maximum number of database connections, this value represents the default max_connections defined in AWS documentation. If SL1 administrator wants to customize this max_connections value, just edit the snippet and set the new value for an DB Engine (MAX_MYSQL_CONNECTIONS, MAX_MARIADB_CONNECTIONS, MAX_POSTGRESQL_CONNECTIONS, MAX_ORACLE_CONNECTIONS or MAX_SQLSERVER_CONNECTIONS).
Network Receive Throughput	The incoming (Receive) network traffic on the DB instance, including both customer database traffic and Amazon RDS traffic used for monitoring and replication.
Network Transmit Throughput	The outgoing (Transmit) network traffic on the DB instance, including both customer database traffic and Amazon RDS traffic used for monitoring and replication.
Read IOPS	The read Input/Output Operations per Second for the database instance.
Read Latency	The average read latency on the database instance.
Read Throughput	The average read throughput on the database instance.
Replica Lag	The current lag time (latency) for replication.
Swap Usage	The usage in bytes of the swap space.
Write IOPS	The write Input/Output Operations per Second for the database instance.
Write Latency	The average latency of write operations on the database instance.
Write Throughput	The average write throughput in Bytes per Second for the database instance.

AWS: RDS Service Health

Object Name	Object Description
Date	The timestamp when the health status was written.
Health	A text description of the health status of this AWS service.

AWS Redshift Service

AWS: Redshift Instance Configuration

Object Name	Object Description
Allow Version Upgrade	Whether or not to allow automatic version upgrades to the database.
Audit Logging Enabled	Whether or not audit logs are kept.
Automated Snapshot Retention Period	The number of days a snapshot is maintained.
AWS Redshift Instance / Security Groups	The security groups associated with this cluster.
AWS Redshift Instance / VPC	The VPC associated with this cluster.
Cluster Id	The unique ID of the cluster.
Cluster Name	The name of the cluster.
Cluster Parameter Group	The parameter group of the cluster.
Cluster Public key	The public key of the cluster in RSA base-64 encoded text format.
Cluster Security Groups	The names of the security groups of the cluster.
Cluster Version	The version ID of the Amazon Redshift engine that is running on the cluster.
Created Time	The creation date of the cluster.
Cross-Region Snapshots Enabled	Whether or not database snapshots are automatically copied to another region.
Current Node Type	The EC2 type used for the database nodes.
Database Name	The name of the database.

Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Encrypted	Whether or not (yes/no) the database is encrypted.
Endpoint	The DNS name of the cluster.
In Maintenance Mode	A Boolean (yes or no); if maintenance mode is active.
Key	Redshift cluster tag key.
Maintenance Track	The name of the maintenance track for the cluster.
Maintenance Window	The timespan chosen for maintenance.
Master Username	The username of the master (i.e. root) database user.
Nodes	The number of nodes in the cluster.
Number of Nodes	The number of nodes in the cluster.
Port	The TCP port of the cluster.
Publicly Accessible	Indicates if the cluster can be accessed from a public network.
Value	Redshift cluster tag value.

AWS: Redshift Instance Performance

Object Name	Object Description
CPU Utilization	The utilization of the cluster's CPU.
Database Connections	The number of current database connections.
Health Status	A Boolean duty-cycle for cluster health.
Maintenance Mode	A Boolean duty-cycle for the maintenance of a cluster.
Network Receive Throughput	The total network throughput received from the cluster.
Network Transmit Throughput	The total transmitted network throughput for the cluster.
Percentage Disk Space Used	The percentage of disk space used by the cluster.
Read IOPS	The total read IOPs for the cluster.
Read Latency	The average read latency for the cluster.
Read Throughput	The total read throughput for the cluster.
Write IOPS	The total write IOPs for the cluster.

Write Latency	The average write latency for the cluster.
Write Throughput	The average write throughput for the cluster.

AWS: Redshift Node Instance Configuration

Object Name	Object Description
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Id	The unique identifier of the node.
Name	The node name.
Private IP Address	The private IP address of the node.
Public IP Address	The public IP address of the node.

AWS: Redshift Node Performance

Object Name	Object Description
Commit Queue Length	Number of transactions in Leader Node where the queuing started.
CPU Utilization	The CPU utilization of the cluster.
Database Connections	The number of current database connections.
Health Status	A Boolean duty-cycle for Leader Node health.
Maintenance Mode	A Boolean duty-cycle for the maintenance of Leader Node.
Network Receive Throughput	The total network throughput received from the cluster.
Network Transmit Throughput	The total transmitted network throughput for the cluster.
Percentage Disk Space Used	The percentage of disk space used by the cluster.
Read IOPS	The total read IOPs for the cluster.
Read Latency	The average read latency for the cluster.
Read Throughput	The total read throughput for the cluster.
Total Table Count	The number of user tables open at a particular point in time.
Write IOPS	The total write IOPS for the cluster.

Write Latency	The average write latency for the cluster.
Write Throughput	The average write throughput for the cluster.

AWS: Redshift Service Health

Object Name	Object Description
Date	A timestamp when the health status of this service was originally written.
Health	A text description of the health status of this AWS service.

AWS Route 53 Service

AWS: Route 53 Health Check Instance Configuration

Object Name	Object Description
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Failure Threshold	The minimum count of Health Check request failures before marking a route as unhealthy.
Host	The name of the health check instance.
Id	The unique identifier of the health check.
IP Address	The IP address of the health check.
Port	The port of the health check.
Protocol	The protocol of the health check.
Request Interval	The period of time between health check network requests.
Search String	The URL path component query string, if any, for the Health Check request.
URL	The resource referenced by the URL for the check.

AWS: Route 53 Health Check Performance

Object Name	Object Description
Health Check Status	This metric represents the duty cycle of the Health Check endpoint divided by 100. For example, 0 means the endpoint was responsive 0% of the time during the monitoring interval, 0.5 = 50% of the time, and 1 = responsive 100% of the time.
Health checkers that report the endpoint healthy	The percentage of Amazon Route 53 health checkers that consider the selected endpoint to be healthy.
Number of healthy child health checks	For a calculated health check, the number of health checks that are healthy among the health checks that Amazon Route 53 is monitoring.
TCP connection time	The average time, in milliseconds, that it took Amazon Route 53 health checkers to establish a TCP connection with the endpoint.
Time to complete SSL handshake	The average time, in milliseconds, that it took Amazon Route 53 health checkers to complete the SSL handshake.
Time to first byte	The average time, in milliseconds, that it took Amazon Route 53 health checkers to receive the first byte of the response to an HTTP or HTTPS request.

AWS: Route 53 Hosted Zone Instance Configuration

Object Name	Object Description
AWS Route 53-Hosted Zone/Application ELB	The unique identifiers of Application ELBs in the Route 53.
AWS Route 53-Hosted Zone/Network ELB	The unique identifiers of Network ELBs in the Route 53.
Comment	Comments, if any, regarding the zone.
Delegation Set	The delegation set of the zone.
Id	The unique identifier of the hosted zone.
Name	The name of the hosted zone.
Record Set Count	The number of zone records.

AWS: Route 53 Service Health

Object Name	Object Description
-------------	--------------------

Date	A timestamp when this service's health status was originally written.
Health	A text description of this AWS Service's health status.

AWS S3 Service

AWS: S3 Cache Configuration

Object Name	Object Description
s3_bucket_cache	Whether or not the S3 cache is collected.

AWS: S3 Instance Configuration

Object Name	Object Description
Bucket Name	The name of the bucket.
Creation Time	The datetime the bucket was created.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Id	The unique identifier of the bucket.
Key	Name of the tag.
Logging Bucket	If logging is enabled, the name of the S3 bucket where the logs are stored.
Logging Prefix	If logging is enabled, the value, if any, of the S3 bucket's object prefix.
Logging Status	Whether or not this bucket's activities are logged.
Owner	The owner of the bucket.
Region	The AWS region where this bucket is deployed.
Value	Value of the tag.
Web Hosting	The website index and error document configurations, if any, for this bucket.

AWS: S3 Request Performance

Object Name	Object Description
4xx Errors	The number of HTTP 4xx client error status code requests made to an Amazon S3 bucket with a value of either 0 or 1.
5xx Errors	The number of HTTP 5xx server error status code requests made to an Amazon S3 bucket with a value of either 0 or 1.
All Requests	The total number of HTTP requests made to an Amazon S3 bucket, regardless of type.
Bytes Downloaded	The number bytes downloaded for requests made to an Amazon S3 bucket, where the response includes a body.
Bytes Uploaded	The number bytes uploaded that contain a request body, made to an Amazon S3 bucket.
Delete Requests	The number of HTTP DELETE requests made for objects in an Amazon S3 bucket.
First Byte Latency	The per-request time from the complete request being received by an Amazon S3 bucket to when the response starts to be returned.
Get Requests	The number of HTTP GET requests made for objects in an Amazon S3 bucket.
Head Requests	The number of HTTP HEAD requests made to an Amazon S3 bucket.
List Requests	The number of HTTP requests that list the contents of a bucket.
Post Requests	The number of HTTP POST requests made to an Amazon S3 bucket.
Put Requests	The number of HTTP PUT requests made for objects in an Amazon S3 bucket.
Select Bytes Returned	The number of bytes of data returned with Amazon S3 SELECT Object Content requests in an Amazon S3 bucket.
Select Bytes Scanned	The number of bytes of data scanned with Amazon S3 SELECT Object Content requests in an Amazon S3 bucket.
Select Requests	The number of Amazon S3 SELECT Object Content requests made for objects in an Amazon S3 bucket.
Total Request Latency	The elapsed per-request time from the first byte received to the last byte sent to an Amazon S3 bucket.

AWS: S3 Service Health

Object Name	Object Description
Date	A timestamp when this service's health status was originally written.
Health	A text description of this AWS Service's health status.

AWS: S3 Service Performance

Object Name	Object Description
Total Bucket Size Bytes	The total collected metrics for BucketSizeBytes metric and its Storage Types: Standard, StandardIA, ReducedRedundancy, OneZonalStorage, IntelligentTieringFASStorage, GlacierStorage.
Total Number Of Objects	The total collected metrics for NumberOfObjects metric and its Storage Types: AllStorageTypes.

AWS: S3 Storage Performance

Object Name	Object Description
Bucket Size Bytes	The amount of data in bytes stored in a bucket in the Standard storage class, Standard - Infrequent Access (Standard_IA) storage class, or the Reduced Redundancy Storage (RRS) storage class.
Number Of Objects	The total number of objects stored in a bucket for all storage classes.
Ol Average Object Size	The average object size in a bucket. If bucket object versioning is enabled then the calculations do include all the object versions.

AWS SES Service

AWS: SES Service Performance

Object Name	Object Description
-------------	--------------------

Bounces	The number of emails that were rejected by email servers of the recipient. This count also includes soft bounces, meaning AWS attempted to deliver the email but was unable to over a period of time.
Clicks	The number of recipients who clicked one or more links in emails.
Complaints	The number of sent emails marked as spam by recipients.
Deliveries	The number of emails Amazon SES successfully delivered to recipients mail servers.
Opens	The number of emails opened by recipients.
Rejects	Calls to Amazon SES that were not successful and Amazon SES will not attempt to deliver the email.
Rendering Failure	The Email was not sent because there was a template rendering error. This can only occur with Template and Bulk Template emails.
Reputation.BounceRate	The rate at which sent emails are bounced.
Reputation.ComplaintRate	The Rate at which sent emails are receiving complaints.
Sends	The number of successful calls to Amazon SES meaning Amazon SES will attempt to deliver the email.

AWS Shield Standard Service

AWS: Shield Advanced Service Performance	
Object Name	Object Description
DDoS Attack Bits Per Second	The number of bytes observed during a DDoS event for a particular Amazon Resource Name (ARN). Reporting criteria: Non-zero value during an attack. Zero when there is no attack.
DDoS Attack Packets Per Second	The number of packets observed during a DDoS event for a particular Amazon Resource Name (ARN). Reporting criteria: Non-zero value during an attack. Zero when there is no attack.
DDoS Attack Requests Per Second	The number of requests observed during a DDoS event for a particular Amazon Resource Name (ARN). Reporting criteria: Non-zero value during an attack. Zero when there is no attack.

DDoS Detected	Indicates a DDoS event for a particular Amazon Resource Name (ARN). Reporting criteria: Non-zero value indicates a DDoS event. Zero when there is no DDoS event detected.
---------------	---

AWS: Shield Service Configuration

Object Name	Object Description
Attack Id	The unique identifier (ID) of the attack.
Emergency Contact Emails	Email address that the DRT can use to contact you during a suspected attack.
End Time	The end time of the attack.
Name	The name of the protection.
Protection ID	The unique identifier (ID) of the protection.
Resource ARN	The ARN (Amazon Resource Name) of the resource that was attacked.
Resource ARN	The ARN (Amazon Resource Name) of the AWS resource that is protected.
Start Time	The start time of the attack.
Subscription Status	The status of the subscription.

AWS SNS Service

AWS: SNS Instance Configuration

Object Name	Object Description
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Subscription ARN	The ARN of the subscription.

Subscription Endpoint	The location of the endpoint for the subscriber. * For email, this would be a valid email address * For email-json, this would be a valid email address * For http, this would be a URL beginning with http * For https, this would be a URL beginning with https * For sqs, this would be the ARN of an SQS Queue * For sms, this would be a phone number of an SMS-enabled device. For application, the endpoint is the EndpointArn of a mobile app and device.
Subscription Owner	The endpoint owner.
Subscription Protocol	The protocol used to communicate with the subscriber. Current choices are: email email-json http https sqs sms application.
Topic Name	The topic name of this instance.

AWS: SNS Instance Performance	
Object Name	Object Description
Avg Published Size	The average published size of messages for this topic.
Max Published Size	The maximum published size of a message for this topic.
Min Published Size	The minimum published size of a message for this topic.
Number of Messages Published	A count of the number of published messages for this topic.
Number of Notifications Delivered	A count of the number of delivered notifications for this instance.
Number of Notifications Failed	The number of notifications which failed delivery.
Number Of Notifications Filtered Out	The number of messages that were rejected by subscription filter policies. A filter policy rejects a message when the message attributes do not match the policy attributes.
Number Of Notifications Filtered Out Invalid Attributes	The number of messages that were rejected by subscription filter policies because the messages attributes are invalid. For example, because the attribute JSON is incorrectly formatted.
Number Of Notifications Filtered Out No Message Attributes	The number of messages that were rejected by subscription filter policies because the messages have no attributes.

Published Size Count	The datum count of published size messages for this topic.
----------------------	--

AWS: SNS Service Health

Object Name	Object Description
Date	A timestamp when this service's health status was originally written.
Health	A text description of this AWS Service's health status.

AWS: SNS Service Performance

Object Name	Object Description
SMS Month To Date Spent USD	Month to Date Spent on SNS in USD.
SMS Success Rate	The success rate of messages delivered to subscribers per country.
SMS Success Rate Label	Country Label for SMS Success Rate Metrics.

AWS SQS Service

AWS: SQS Instance Configuration

Object Name	Object Description
ARN	The Amazon Resource Name (ARN) (a unique identifier) of this queue.
Created	The timestamp when this queue was created.
Delivery Delay	The built-in delivery delay for messages from the initial time of queueing.
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Last Updated	The timestamp when this queue was last modified.
Max Message Size	The maximum message size, in bytes, that this queue has contained.

Message Delayed	An estimate for the number of delayed messages (incorporating the built-in delay) in the queue.
Messages Available	An estimate for the number of messages still in the queue.
Messages In Flight	An estimate for the number of messages that have been dequeued, but not yet delivered.
Name	The name of the queue.
Retention Period	The maximum time (a kind of TTL) that a message may remain in the queue.
URL	The Uniform Resource Locator (URL) of the queue.
Visibility Timeout	The time period when AWS prevents queue clients from processing messages which has been delivered, but not explicitly deleted by prior clients.

AWS: SQS Instance Performance

Object Name	Object Description
Approximate Age of Oldest Message	The approximate age of the oldest non-deleted message in the queue.
Approximate Number of Messages Delayed	The number of messages in the queue that are delayed and not available for reading immediately. This can happen when the queue is configured as a delay queue or when a message has been sent with a delay parameter.
Approximate Number of Messages Not Visible	The approximate number of messages not visible in this queue.
Approximate Number of Messages Visible	The approximate number of visible messages in this queue.
Avg Sent Message Size	The average size of messages sent through this queue.
Max Sent Message Size	The maximum size of all messages sent through this queue.
Min Sent Message Size	The minimum size of all messages sent through this queue.
Number of Empty Receives	The number of empty messages received by this queue.
Number of Messages Deleted	The number of messages deleted from this queue.
Number of Messages Received	The number of messages received by this queue.

Number of Messages Sent	The number of messages send through this queue.
Sent Message Size Count	The datum counts of all message sizes sent through this queue.

AWS: SQS Service Health

Object Name	Object Description
Date	The timestamp when this health status was written.
Health	A text description of the health status of the AWS Service.

AWS Storage Gateway Service

AWS: Storage Gateway Instance Configuration

Object Name	Object Description
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Gateway ID	The unique identifier of this gateway containing Region information.
Gateway Name	The name of this gateway.

AWS: Storage Gateway Instance Performance

Object Name	Object Description
Cloud Bytes Downloaded	The average number of bytes downloaded from the cloud to this gateway.
Cloud Bytes Uploaded	The average number of bytes uploaded to the cloud from this gateway.
Cloud Download Latency	The average latency when downloading from the cloud to this gateway.
Queued Writes	The average number of queued write operations on this gateway.
Read Bytes	The average number of bytes read by this gateway.

Read Time	The average read times for this gateway.
Working Storage Free	The average free/available working storage on this gateway.
Working Storage Percent Used	The percentage of the local Working Storage buffer currently being used.
Working Storage Used	The average working storage used for this gateway.
Write Bytes	The average number of bytes written by this gateway.
Write Time	The average write times for this storage gateway.

AWS: Storage Gateway Service Health

Object Name	Object Description
Date	The timestamp when this health status update was written.
Health	A text description of the health status of the AWS Service.

AWS: Storage Gateway Volume Configuration

Object Name	Object Description
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Volume ID	The unique identifier for this volume.
Volume Name	The name of this filesystem volume.

AWS: Storage Gateway Volume Performance

Object Name	Object Description
Queued Writes	The average number of queued (pending) write operations on this gateway volume.
Read Bytes	The average bytes read from this gateway volume.
Read Time	The average time for read operations for this gateway volume.
Write Bytes	The average bytes written to this gateway volume.

Write Time	The average time for write operations for this gateway volume.
------------	--

AWS STS Service

AWS: STS Session Manager	
Object Name	Object Description
Account ID	The unique identifier (ID) of the account.
Organization Namespace	Namespace defined to create relationship between organization and its accounts.
Response Status	Response Status as value starting with Success or Error. Once you run the collect method of the sts_session_manager class it will change its value.

AWS Transit Gateway Service

AWS: Transit Gateway Instance Configuration	
Object Name	Object Description
Attachment Id	The ID of the attachment.
Attachment Id	The ID of the attachment.
Attachment Name	The name of the attachment.
Attachment Name	The name of the attachment.
ID	The ID of transit gateway.
Name	The name of transit gateway.
Transit Gateway Route Table Id	The ID of the transit gateway route table.
Transit Gateway/VPC	The ID of the resource.
ARN	The ARN of transit gateway.
Default Association Route Table	Indicates whether this is the default association route table for the transit gateway (true false).
Default Propagation Route Table	Indicates whether this is the default propagation route table for the transit gateway (true false).

Owner Id	The ID of the AWS account that owns the transit gateway.
Resource Id	The ID of the resource.
Resource Type	The resource type. Valid values are vpc vpn direct-connect-gateway peering connect.
Resource Type	The resource type. Valid values are vpc vpn direct-connect-gateway peering connect.
State	The state of transit gateway (available deleted deleting modifying pending).
State	The state of the attachment. Valid values are available deleted deleting failed failing initiatingRequest modifying pendingAcceptance pending rollingBack rejected rejecting.
State	The state of the route table (available deleting deleted pending).
State	The state of the attachment. Valid values are available deleted deleting failed failing initiatingRequest modifying pendingAcceptance pending rollingBack rejected rejecting.
Creation Time	The creation time of transit gateway.
Creation Time	The creation time of the attachment.
Creation Time	The creation time of the route table.
Creation Time	The creation time of the attachment.
Key	The tag key (String).
Value	The value of the tag key.

AWS: Transit Gateway Instance Performance

Object Name	Object Description
Bytes Drop Count Blackhole	The number of bytes dropped because they matched a blackhole route.
Bytes Drop Count No Route	The number of bytes dropped because they did not match a route.
Bytes In	The number of bytes received by the transit gateway.
Bytes Out	The number of bytes sent from the transit gateway.

Packet Drop Count Blackhole	The number of packets dropped because they matched a blackhole route.
Packet Drop Count No Route	The number of packets dropped because they did not match a route.
Packets In	The number of packets received by the transit gateway.
Packets Out	The number of packets sent by the transit gateway.

AWS VPC Service

AWS: VPC Instance Configuration	
Object Name	Object Description
VPC Peer ID	The ID of the VPC peering connection.
VPC Peer Name	The name of the VPC peering connection.
Accepter VPC ID	The ID of the accepter VPC.
Status	The status of the VPC peering connection.
AWS VPC/EC2	The unique identifier for the VPC-EC2 relationship.
AWS VPC/Security Group	The unique identifier for the VPC-Security Group relationship.
AWS VPC/VPC Inter-Account	The relationship between Inter-Account VPCs. Contains the relationship namespace value. A relationship between accounts is drawn when the relationship namespace has the same value as the identity namespace and their respective group indices also have the same value.
AWS VPC/VPC Intra-Account	The relationship between Intra-Account VPCs.
DHCP Options Set	The resource ID of the DHCP option set for the VPC. DHCP options are associated with your AWS account so that you can use them across all of your virtual private clouds (VPC).
Distinguished Name	The internal SL1 distinguished name of the AWS component. This follows closely the format of the AWS Amazon Resource Name (ARN).
Id	The unique identifier of the VPC.
Inter-Account Requester VPC ID	The Inter-Account Requester VPC ID. Value populated only when the current VPC is the Requester VPC.

Inter-Account Requester VPC ID	The Inter-Account Requester VPC ID.
Is Default VPC	Is this a default VPC. True False. A default VPC combines the benefits of the advanced features provided by EC2-VPC with the ease of use of EC2-Classic. If you have a default VPC and do not specify a subnet when you launch an instance, the instance is launched into your default VPC. You can launch instances into your default VPC without needing to know anything about Amazon VPC.
Key	The tag key. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for your account's Amazon EC2 instances that helps you track each instance's owner and stack level. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.
Name	The Name property, if any, of the VPC as set in its tags.
Requester VPC ID	The Requester VPC ID.
State	Whether the VPC is available or not.
Tenancy	One of default dedicated. If default, your instance runs on shared hardware. If dedicated, your instance runs on single-tenant hardware.
Value	The tag value. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for your account's Amazon EC2 instances that helps you track each instance's owner and stack level. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.
VPC CIDR	The Classless Inter-Domain Routing(CIDR) block for the VPC.
vpc_identity_namespace	The identity namespace used to identify Inter-Account relationships. Value populated only when the current VPC is the Requester VPC.

AWS: VPC NAT Gateway Instance Configuration

Object Name	Object Description
Allocated ID	The allocation ID of the Elastic IP address that is associated with the NAT gateway.
AWS NAT Gateway/Subnet	The unique identifier for the VPC NAT Gateway-Subnet relationship.
Create Time	The date and time the NAT gateway was created.
Key	The key of a tag assigned to the resource.
NAT Gateway ID	The ID of the NAT gateway.
Network Interface ID	The ID of the network interface associated with the NAT gateway.
Private Ip	The private IP address associated with the Elastic IP address.
Public Ip	The Elastic IP address associated with the NAT gateway.
State	The state of the NAT gateway. Possible values are: pending failed failureMessage available deleting deleted.
Value	The value of a tag assigned to the resource.
VPC ID	The ID of the VPC in which the NAT gateway is located.

AWS: VPC NAT Gateway Instance Performance

Object Name	Object Description
Active Connection Count	The total number of concurrent active TCP connections through the NAT gateway. A value of zero indicates that there are no active connections through the NAT gateway.
Bytes In From Destination	The number of bytes received by the NAT gateway from the destination. If the value for BytesOutToSource is less than the value for BytesInFromDestination, there may be data loss during NAT gateway processing, or traffic being actively blocked by the NAT gateway.
Bytes In From Source	The number of bytes received by the NAT gateway from clients in your VPC. If the value for BytesOutToDestination is less than the value for BytesInFromSource, there may be data loss during NAT gateway processing.

Bytes Out To Destination	The number of bytes sent out through the NAT gateway to the destination. A value greater than zero indicates that there is traffic going to the internet from clients that are behind the NAT gateway. If the value for BytesOutToDestination is less than the value for BytesInFromSource, there may be data loss during NAT gateway processing.
Bytes Out To Source	The number of bytes sent through the NAT gateway to the clients in your VPC. A value greater than zero indicates that there is traffic coming from the internet to clients that are behind the NAT gateway. If the value for BytesOutToSource is less than the value for BytesInFromDestination, there may be data loss during NAT gateway processing, or traffic being actively blocked by the NAT gateway.
Connection Attempt Count	The number of connection attempts made through the NAT gateway. If the value for ConnectionEstablishedCount is less than the value for ConnectionAttemptCount, this indicates that clients behind the NAT gateway attempted to establish new connections for which there was no response.
Connection Established Count	The number of connections established through the NAT gateway. If the value for ConnectionEstablishedCount is less than the value for ConnectionAttemptCount, this indicates that clients behind the NAT gateway attempted to establish new connections for which there was no response.
Error Port Allocation	The number of times the NAT gateway could not allocate a source port. A value greater than zero indicates that too many concurrent connections are open through the NAT gateway.
Idle Timeout Count	The number of connections that transitioned from the active state to the idle state. An active connection transitions to idle if it was not closed gracefully and there was no activity for the last 350 seconds. A value greater than zero indicates that there are connections that have been moved to an idle state. If the value for IdleTimeoutCount increases, it may indicate that clients behind the NAT gateway are re-using stale connections.
Packets Drop Count	The number of packets dropped by the NAT gateway. A value greater than zero may indicate an ongoing transient issue with the NAT gateway.

Packets In From Destination	The number of packets received by the NAT gateway from the destination. If the value for PacketsOutToSource is less than the value for PacketsInFromDestination, there may be data loss during NAT gateway processing, or traffic being actively blocked by the NAT gateway.
Packets In From Source	The number of packets received by the NAT gateway from clients in your VPC. If the value for PacketsOutToDestination is less than the value for PacketsInFromSource, there may be data loss during NAT gateway processing.
Packets Out To Destination	The number of packets sent out through the NAT gateway to the destination. A value greater than zero indicates that there is traffic going to the internet from clients that are behind the NAT gateway. If the value for PacketsOutToDestination is less than the value for PacketsInFromSource, there may be data loss during NAT gateway processing.
Packets Out To Source	The number of packets sent through the NAT gateway to the clients in your VPC. A value greater than zero indicates that there is traffic coming from the internet to clients that are behind the NAT gateway. If the value for PacketsOutToSource is less than the value for PacketsInFromDestination, there may be data loss during NAT gateway processing, or traffic being actively blocked by the NAT gateway.

AWS: VPC Route Table Configuration

Object Name	Object Description
AWS Route Table/Subnet	The unique identifier for the Route Table-Subnet relationship.
AWS Route Table/Virtual Private Gateway	The unique identifier for the Route Table-Virtual Private Gateway relationship.
Destination	The Destination in a route.

Key	The tag key. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for your Amazon EC2 instances account that helps you track each owner of the instance and stack level. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.
Main	Whether the Route Table is the default (main) or otherwise.
Route Table ID	The unique identifier of the VPC Route Table.
Status	Whether a Route in the VPC Route Table is active or not.
Target	The Gateway of a Route defined in a VPC Route Table.
Value	The tag value. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for your Amazon EC2 instances account that helps you track each owner of the instance and stack level. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.
VPC ID	The unique identifier of the VPC.

AWS: VPC Service Health

Object Name	Object Description
Date	The timestamp of this service's health update.
Health	A text description of this AWS Service's health status.

AWS: VPC Subnet Configuration

Object Name	Object Description
Available IP Address Count	The number of IPv4 addresses in the subnet that are available.
AWS VPC Subnet/EC2	The unique identifier for the VPC Subnet-EC2 relationship.
Id	The unique identifier of the subnet.
Key	The tag key. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for your Amazon EC2 instances account that helps you track each owner of the instance and stack level. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.
Name	The Name property, if any, of the subnet as set in its tags.
Subnet CIDR	The IPv4 CIDR block of the subnet.
Value	The tag value. Tags enable you to categorize your AWS resources in different ways, for example, by purpose, owner, or environment. Each tag consists of a key and an optional value, both of which you define. For example, you could define a set of tags for your Amazon EC2 instances account that helps you track each owner of the instance and stack level. We recommend that you devise a set of tag keys that meets your needs for each resource type. Using a consistent set of tag keys makes it easier for you to manage your resources. You can search and filter the resources based on the tags you add.

AWS: VPC Virtual Private Gateway Configuration

Object Name	Object Description
Key	The key of a tag assigned to the resource.
State	The current state of the attachment between the gateway and the VPC (attaching attached detaching detached).

Value	The value of a tag assigned to the resource.
VPC ID	The unique identifier of an attached VPC.
VPN Gateway ID	The unique identifier of the virtual private gateway.
VPN Gateway Name	The name of the virtual private gateway.
VPN Gateway State	The state of the virtual private gateway (pending available deleting deleted).
VPN Gateway Type	The type of virtual private gateway. Currently the only supported type is ipsec.1.

AWS WAF Global Service

AWS: WAF Global WebACL Instance Configuration

Object Name	Object Description
Action	The action that CloudFront or AWS WAF takes when a web request matches the conditions in the Rule.
Default Action	The WebACL instance component default action.
Id	The Id for a Rule.
Name	The WebACL instance component name.
Name	The name for a Rule.
Priority	The order in which the Rules in a WebACL are evaluated.
Type	Type of AWS WAF response to requests that match the settings in a Rule.
Web ACL ARN	The ARN for the WebACL component.
Web ACL Id	The WebACL instance component Unique Identifier.

AWS: WAF Global WebACL Instance Performance

Object Name	Object Description
Allowed (By Rule)	The number of allowed web requests by rule.
Blocked (By Rule Group)	The number of blocked web requests by rule group.
Blocked (By Rule)	The number of blocked web requests by rule.

Counted (By Rule Group)	The number of counted web requests by rule group.
Counted (By Rule)	The number of counted web requests by rule.

AWS: WAF Regional WebACL Instance Configuration

Object Name	Object Description
Action	Specifies the action that API Gateway or Application ELB or AWS WAF takes when a web request matches the conditions in the Rule.
AWS WAF/API Gateway	The API IDs of resources associated with the specified web ACL.
AWS WAF/API Stage	The API Stage IDs of resources associated with the specified web ACL.
AWS WAF/Application ELB	The Application ELB IDs of resources associated with the specified web ACL.
Default Action	The action to perform if none of the Rules contained in the WebACL match.
Metric Name	A friendly name or description for the metrics for this WebACL.
Name	The name for a Rule.
Priority	Specifies the order in which the Rules in a WebACL are evaluated.
Rule Id	The RuleId for a Rule.
Type	Specifies how you want AWS WAF to respond to requests that match the settings in a Rule.
WebACL ARN	The ARN for the WebACL component.
WebACL Id	The WebACL instance component Unique Identifier.
WebACL Name	A friendly name or description of the WebACL.

AWS: WAF Regional WebACL Instance Performance

Object Name	Object Description
Allowed (By Rule)	The number of allowed web requests by rule.
Blocked (By Rule Group)	The number of blocked web requests by rule group.
Blocked (By Rule)	The number of blocked web requests by rule.

Counted (By Rule Group)	The number of counted web requests by rule group.
Counted (By Rule)	The number of counted web requests by rule.

AWS: WAF Service Health

Object Name	Object Description
Date	The date of the RSS feed entry.
Health	The message of the RSS feed entry.

AWS Workspaces Service

AWS: Workspaces Directory Configuration

Object Name	Object Description
Alias	The directory alias.
Change Compute Type	Specifies whether users can change the compute type (bundle) for their WorkSpace.
Customer User Name	The user name for the service account.
Device Type Android	Indicates whether users can use Android and Android-compatible Chrome OS devices to access their WorkSpaces.
Device Type Chrome OS	Indicates whether users can use Chromebooks to access their WorkSpaces.
Device Type IOS	Indicates whether users can use iOS devices to access their WorkSpaces.
Device Type Linux	Indicates whether users can use Linux clients to access their WorkSpaces.
Device Type OSX	Indicates whether users can use OSX clients to access their WorkSpaces.
Device Type Web	Indicates whether users can access their WorkSpaces through a web browser.
Device Type Windows	Indicates whether users can use Windows clients to access their WorkSpaces.
Device Type Zero Client	Indicates whether users can use zero client devices to access their WorkSpaces.

Directory Id	The unique identifier of the directory
Directory Name	The directory name or directory id if name is not present, prefixed by the region where it is located.
Directory Type	The directory type.
DNS IP Addresses	The IP addresses of the DNS servers for the directory.
Enable Internet Access	Specifies whether to automatically assign an Elastic public IP address to WorkSpaces in this directory by default.
Enable Maintenance Mode	Specifies whether maintenance mode is enabled for WorkSpaces.
Enable Work Docs	Specifies whether the directory is enabled for Amazon WorkDocs.
IAM Role Id	The identifier of the IAM role. This is the role that allows Amazon WorkSpaces to make calls to other services, such as Amazon EC2, on your behalf.
Increase Volume Size	Specifies whether users can increase the volume size of the drives on their Workspace.
Rebuild Workspace	Specifies whether users can rebuild the operating system of a Workspace to its original state.
Registration Code	The registration code for the directory. This is the code that users enter in their Amazon WorkSpaces client application to connect to the directory.
Restart Workspace	Specifies whether users can restart their Workspace.
State	The state of the directory registration with Amazon WorkSpaces.
Subnet Ids	The identifiers of the subnets used with the directory.
Switch Running Mode	Specifies whether users can switch the running mode of their Workspace.
Tenancy	Specifies whether the directory is dedicated or shared.
User Enabled As Local Administrator	Specifies whether Workspace users are local administrators on their WorkSpaces.

AWS: Workspaces Directory Performance

Object Name	Object Description
-------------	--------------------

Available	The number of WorkSpaces that returned a healthy status.
Connection Attempt	The number of connection attempts.
Connection Failure	The number of failed connections.
Connection Success	The number of successful connections.
In Session Latency	The round trip time between the WorkSpaces client and the Workspace.
Maintenance	The number of WorkSpaces that are under maintenance.
Session Disconnect	The number of connections that were closed, including user-initiated and failed connections.
Session Launch Time	The amount of time it takes to initiate a WorkSpaces session.
Stopped	The number of WorkSpaces that are stopped.
Trusted Device Validation Attempt	The number of device authentication signature validation attempts.
Trusted Device Validation Failure	The number of failed device authentication signature validations.
Trusted Device Validation Success	The number of successful device authentication signature validations.
Unhealthy	The number of WorkSpaces that returned an unhealthy status.
User Connected	The number of WorkSpaces that have a user connected.

© 2003 - 2022, ScienceLogic, Inc.

All rights reserved.

LIMITATION OF LIABILITY AND GENERAL DISCLAIMER

ALL INFORMATION AVAILABLE IN THIS GUIDE IS PROVIDED "AS IS," WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED. SCIENCELOGIC™ AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT.

Although ScienceLogic™ has attempted to provide accurate information on this Site, information on this Site may contain inadvertent technical inaccuracies or typographical errors, and ScienceLogic™ assumes no responsibility for the accuracy of the information. Information may be changed or updated without notice. ScienceLogic™ may also make improvements and / or changes in the products or services described in this Site at any time without notice.

Copyrights and Trademarks

ScienceLogic, the ScienceLogic logo, and EM7 are trademarks of ScienceLogic, Inc. in the United States, other countries, or both.

Below is a list of trademarks and service marks that should be credited to ScienceLogic, Inc. The ® and ™ symbols reflect the trademark registration status in the U.S. Patent and Trademark Office and may not be appropriate for materials to be distributed outside the United States.

- ScienceLogic™
- EM7™ and em7™
- Simplify IT™
- Dynamic Application™
- Relational Infrastructure Management™

The absence of a product or service name, slogan or logo from this list does not constitute a waiver of ScienceLogic's trademark or other intellectual property rights concerning that name, slogan, or logo.

Please note that laws concerning use of trademarks or product names vary by country. Always consult a local attorney for additional guidance.

Other

If any provision of this agreement shall be unlawful, void, or for any reason unenforceable, then that provision shall be deemed severable from this agreement and shall not affect the validity and enforceability of any remaining provisions. This is the entire agreement between the parties relating to the matters contained herein.

In the U.S. and other jurisdictions, trademark owners have a duty to police the use of their marks. Therefore, if you become aware of any improper use of ScienceLogic Trademarks, including infringement or counterfeiting by third parties, report them to Science Logic's legal department immediately. Report as much detail as possible about the misuse, including the name of the party, contact information, and copies or photographs of the potential misuse to: legal@sciencelogic.com

ScienceLogic

800-SCI-LOGIC (1-800-724-5644)

International: +1-703-354-1010