**Monitoring Memory and Disk Metrics for Amazon EC2 Linux Instances**

**New CloudWatch Agent Available**

A new multi-platform CloudWatch agent is available. You can use a single agent to collect both system metrics and log files from Amazon EC2 instances and on-premises servers. The new agent supports both Windows Server and Linux and enables you to select the metrics to be collected, including sub-resource metrics such as per-CPU core. We recommend you use the new agent instead of the older monitoring scripts to collect metrics and logs. For more information about the CloudWatch agent, see [Collect Metrics from Amazon EC2 Instances and On-Premises Servers with the CloudWatch Agent](http://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/Install-CloudWatch-Agent.html) in the *Amazon CloudWatch User Guide*.

The rest of this section is informational for customers who are still using the older Perl scripts for monitoring. You can download these [Amazon CloudWatch Monitoring Scripts for Linux](https://aws.amazon.com/code/8720044071969977) from the AWS sample code library.

**CloudWatch Monitoring Scripts**

The Amazon CloudWatch Monitoring Scripts for Amazon Elastic Compute Cloud (Amazon EC2) Linux-based instances demonstrate how to produce and consume Amazon CloudWatch custom metrics. These sample Perl scripts comprise a fully functional example that reports memory, swap, and disk space utilization metrics for a Linux instance.

Standard Amazon CloudWatch usage charges for custom metrics apply to your use of these scripts. For more information, see the [Amazon CloudWatch](https://aws.amazon.com/cloudwatch/pricing) pricing page.

**Contents**

* [Supported Systems](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/mon-scripts.html#mon-scripts-systems)
* [Package Contents](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/mon-scripts.html#mon-scripts-contents)
* [Prerequisites](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/mon-scripts.html#mon-scripts-perl_prereq)
* [Getting Started](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/mon-scripts.html#mon-scripts-getstarted)
* [mon-put-instance-data.pl](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/mon-scripts.html#using_put_script)
* [mon-get-instance-stats.pl](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/mon-scripts.html#using_get_script_powershell)
* [Viewing Your Custom Metrics in the Console](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/mon-scripts.html#script_viewmetrics)
* [Troubleshooting](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/mon-scripts.html#mon-script-troubleshooting)

**Supported Systems**

These monitoring scripts are intended for use with Amazon EC2 instances running Linux. The scripts have been tested on instances using the following Amazon Machine Images (AMIs), both 32-bit and 64-bit versions:

* Amazon Linux 2014.09.2
* Red Hat Enterprise Linux 7.4 and 6.9
* SUSE Linux Enterprise Server 12
* Ubuntu Server 16.04 and 14.04

**Note**

On servers running SUSE Linux Enterprise Server 12, you may need to first download the perl-Switch package. You can download and install this package with the following commands:

wget http://download.opensuse.org/repositories/devel:/languages:/perl/SLE\_12\_SP3/noarch/perl-Switch-2.17-32.1.noarch.rpm

sudo rpm -i perl-Switch-2.17-32.1.noarch.rpm

You can also monitor memory and disk metrics on Amazon EC2 instances running Windows by sending this data to CloudWatch Logs. For more information, see [Sending Logs, Events, and Performance Counters to Amazon CloudWatch](http://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/send_logs_to_cwl.html) in the *Amazon EC2 User Guide for Windows Instances*.

**Package Contents**

The package for the monitoring scripts contains the following files:

* **CloudWatchClient.pm** – Shared Perl module that simplifies calling Amazon CloudWatch from other scripts.
* **mon-put-instance-data.pl** – Collects system metrics on an Amazon EC2 instance (memory, swap, disk space utilization) and sends them to Amazon CloudWatch.
* **mon-get-instance-stats.pl** – Queries Amazon CloudWatch and displays the most recent utilization statistics for the EC2 instance on which this script is executed.
* **awscreds.template** – File template for AWS credentials that stores your access key ID and secret access key.
* **LICENSE.txt** – Text file containing the Apache 2.0 license.
* **NOTICE.txt** – Copyright notice.

**Prerequisites**

With some versions of Linux, you must install additional modules before the monitoring scripts will work.

***Amazon Linux AMI***

If you are running Amazon Linux AMI version 2014.03 or later, you must install additional Perl modules.

**To install the required packages**

1. Log on to your instance. For more information, see [Connect to Your Linux Instance](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstances.html).
2. At a command prompt, install packages as follows:

**sudo yum install perl-Switch perl-DateTime perl-Sys-Syslog perl-LWP-Protocol-https -y**

**Note**

If you are running Amazon Linux 2, you may also need to install perl-Digest-SHA.x86\_64.

***Red Hat Enterprise Linux***

You must install additional Perl modules.

**To install the required packages on Red Hat Enterprise Linux 6.9**

1. Log on to your instance. For more information, see [Connect to Your Linux Instance](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstances.html).
2. At a command prompt, install packages as follows:
3. **sudo yum install perl-DateTime perl-CPAN perl-Net-SSLeay perl-IO-Socket-SSL perl-Digest-SHA gcc -y**

**sudo yum install zip unzip**

1. Run CPAN as an elevated user:

sudo cpan

Press ENTER through the prompts until you see the following prompt:

cpan[1]>

1. At the CPAN prompt, run each of the below commands: run one command and it installs, and when you return to the CPAN prompt, run the next command. Press ENTER like before when prompted to continue through the process:
2. cpan[1]> install YAML
3. cpan[1]> install LWP::Protocol::https
4. cpan[1]> install Sys::Syslog
5. cpan[1]> install Switch

**To install the required packages on Red Hat Enterprise Linux 7.4**

1. Log on to your instance. For more information, see [Connect to Your Linux Instance](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstances.html).
2. At a command prompt, install packages as follows:
3. **sudo yum install perl-Switch perl-DateTime perl-Sys-Syslog perl-LWP-Protocol-https perl-Digest-SHA --enablerepo="rhui-REGION-rhel-server-optional" -y**

**sudo yum install zip unzip**

***SUSE Linux Enterprise Server***

You must install additional Perl modules.

**To install the required packages on SUSE**

1. Log on to your instance. For more information, see [Connect to Your Linux Instance](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstances.html).
2. At a command prompt, install packages as follows:
3. **sudo zypper install perl-Switch perl-DateTime**

**sudo zypper install –y "perl(LWP::Protocol::https)"**

***Ubuntu Server***

You must configure your server as follows.

**To install the required packages on Ubuntu**

1. Log on to your instance. For more information, see [Connect to Your Linux Instance](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AccessingInstances.html).
2. At a command prompt, install packages as follows:
3. **sudo apt-get update**
4. **sudo apt-get install unzip**

**sudo apt-get install libwww-perl libdatetime-perl**

**Getting Started**

The following steps show you how to download, uncompress, and configure the CloudWatch Monitoring Scripts on an EC2 Linux instance.

**To download, install, and configure the monitoring scripts**

1. At a command prompt, move to a folder where you want to store the monitoring scripts and run the following command to download them:

curl https://aws-cloudwatch.s3.amazonaws.com/downloads/CloudWatchMonitoringScripts-1.2.2.zip -O

1. Run the following commands to install the monitoring scripts you downloaded:
2. unzip CloudWatchMonitoringScripts-1.2.2.zip
3. rm CloudWatchMonitoringScripts-1.2.2.zip

cd aws-scripts-mon

1. Ensure that the scripts have permission to perform CloudWatch operations using one of the following options:
   * If you associated an IAM role (instance profile) with your instance, verify that it grants permissions to perform the following operations:
     + cloudwatch:PutMetricData
     + cloudwatch:GetMetricStatistics
     + cloudwatch:ListMetrics
     + ec2:DescribeTags
   * Specify your AWS credentials in a credentials file. First, copy the awscreds.template file included with the monitoring scripts to awscreds.conf as follows:

cp awscreds.template awscreds.conf

Add the following content to this file:

AWSAccessKeyId=*my-access-key-id*

AWSSecretKey=*my-secret-access-key*

For information about how to view your AWS credentials, see [Understanding and Getting Your Security Credentials](http://docs.aws.amazon.com/general/latest/gr/aws-sec-cred-types.html) in the *Amazon Web Services General Reference*.

**mon-put-instance-data.pl**

This script collects memory, swap, and disk space utilization data on the current system. It then makes a remote call to Amazon CloudWatch to report the collected data as custom metrics.

**Options**

|  |  |
| --- | --- |
| **Name** | **Description** |
| --mem-util | Collects and sends the MemoryUtilization metrics in percentages. This metric counts memory allocated by applications and the operating system as used, and also includes cache and buffer memory as used if you specify the --mem-used-incl-cache-buff option. |
| --mem-used | Collects and sends the MemoryUsed metrics, reported in megabytes. This metric counts memory allocated by applications and the operating system as used, and also includes cache and buffer memory as used if you specify the --mem-used-incl-cache-buff option. |
| --mem-used-incl-cache-buff | If you include this option, memory currently used for cache and buffers is counted as "used" when the metrics are reported for --mem-util, --mem-used, and --mem-avail. |
| --mem-avail | Collects and sends the MemoryAvailable metrics, reported in megabytes. This metric counts memory allocated by applications and the operating system as used, and also includes cache and buffer memory as used if you specify the --mem-used-incl-cache-buff option. |
| --swap-util | Collects and sends SwapUtilization metrics, reported in percentages. |
| --swap-used | Collects and sends SwapUsed metrics, reported in megabytes. |
| --disk-path=PATH | Selects the disk on which to report.  PATH can specify a mount point or any file located on a mount point for the filesystem that needs to be reported. For selecting multiple disks, specify a --disk-path=PATH for each one of them.  To select a disk for the filesystems mounted on / and /home, use the following parameters:  --disk-path=/ --disk-path=/home |
| --disk-space-util | Collects and sends the DiskSpaceUtilization metric for the selected disks. The metric is reported in percentages.  Note that the disk utilization metrics calculated by this script differ from the values calculated by the df -k -l command. If you find the values from df -k -l more useful, you can change the calculations in the script. |
| --disk-space-used | Collects and sends the DiskSpaceUsed metric for the selected disks. The metric is reported by default in gigabytes.  Due to reserved disk space in Linux operating systems, disk space used and disk space available might not accurately add up to the amount of total disk space. |
| --disk-space-avail | Collects and sends the DiskSpaceAvailable metric for the selected disks. The metric is reported in gigabytes.  Due to reserved disk space in the Linux operating systems, disk space used and disk space available might not accurately add up to the amount of total disk space. |
| --memory-units=UNITS | Specifies units in which to report memory usage. If not specified, memory is reported in megabytes. UNITS may be one of the following: bytes, kilobytes, megabytes, gigabytes. |
| --disk-space-units=UNITS | Specifies units in which to report disk space usage. If not specified, disk space is reported in gigabytes. UNITS may be one of the following: bytes, kilobytes, megabytes, gigabytes. |
| --aws-credential- file=PATH | Provides the location of the file containing AWS credentials.  This parameter cannot be used with the --aws-access-key-id and --aws-secret-key parameters. |
| --aws-access-key-id=VALUE | Specifies the AWS access key ID to use to identify the caller. Must be used together with the --aws-secret-key option. Do not use this option with the --aws-credential-file parameter. |
| --aws-secret-key=VALUE | Specifies the AWS secret access key to use to sign the request to CloudWatch. Must be used together with the --aws-access-key-id option. Do not use this option with --aws-credential-file parameter. |
| --aws-iam-role=VALUE | Specifies the IAM role used to provide AWS credentials. The value =VALUE is required. If no credentials are specified, the default IAM role associated with the EC2 instance is applied. Only one IAM role can be used. If no IAM roles are found, or if more than one IAM role is found, the script will return an error.  Do not use this option with the --aws-credential-file, --aws-access-key-id, or --aws-secret-key parameters. |
| --aggregated[=only] | Adds aggregated metrics for instance type, AMI ID, and overall for the region. The value =only is optional; if specified, the script reports only aggregated metrics. |
| --auto-scaling[=only] | Adds aggregated metrics for the Auto Scaling group. The value =only is optional; if specified, the script reports only Auto Scaling metrics. The [IAM policy](http://docs.aws.amazon.com/IAM/latest/UserGuide/ManagingPolicies.html) associated with the IAM account or role using the scripts need to have permissions to call the EC2 action [DescribeTags](http://docs.aws.amazon.com/AWSEC2/latest/APIReference/ApiReference-query-DescribeTags.html). |
| --verify | Performs a test run of the script that collects the metrics, prepares a complete HTTP request, but does not actually call CloudWatch to report the data. This option also checks that credentials are provided. When run in verbose mode, this option outputs the metrics that will be sent to CloudWatch. |
| --from-cron | Use this option when calling the script from cron. When this option is used, all diagnostic output is suppressed, but error messages are sent to the local system log of the user account. |
| --verbose | Displays detailed information about what the script is doing. |
| --help | Displays usage information. |
| --version | Displays the version number of the script. |

***Examples***

The following examples assume that you provided an IAM role or awscreds.conf file. Otherwise, you must provide credentials using the --aws-access-key-id and --aws-secret-key parameters for these commands.

**To perform a simple test run without posting data to CloudWatch**

./mon-put-instance-data.pl --mem-util --verify --verbose

**To collect all available memory metrics and send them to CloudWatch, counting cache and buffer memory as used**

./mon-put-instance-data.pl --mem-used-incl-cache-buff --mem-util --mem-used --mem-avail

**To set a cron schedule for metrics reported to CloudWatch**

1. Start editing the crontab using the following command:

crontab -e

1. Add the following command to report memory and disk space utilization to CloudWatch every five minutes:

\*/5 \* \* \* \* ~/aws-scripts-mon/mon-put-instance-data.pl --mem-used-incl-cache-buff --mem-util --disk-space-util --disk-path=/ --from-cron

If the script encounters an error, the script will write the error message in the system log.

**To collect aggregated metrics for an Auto Scaling group and send them to Amazon CloudWatch without reporting individual instance metrics**

./mon-put-instance-data.pl --mem-util --mem-used --mem-avail --auto-scaling=only

**To collect aggregated metrics for instance type, AMI ID and region, and send them to Amazon CloudWatch without reporting individual instance metrics**

./mon-put-instance-data.pl --mem-util --mem-used --mem-avail --aggregated=only

**mon-get-instance-stats.pl**

This script queries CloudWatch for statistics on memory, swap, and disk space metrics within the time interval provided using the number of most recent hours. This data is provided for the Amazon EC2 instance on which this script is executed.

**Options**

|  |  |
| --- | --- |
| **Name** | **Description** |
| --recent-hours=N | Specifies the number of recent hours to report on, as represented by N where N is an integer. |
| --aws-credential-file=PATH | Provides the location of the file containing AWS credentials. |
| --aws-access-key-id=VALUE | Specifies the AWS access key ID to use to identify the caller. Must be used together with the --aws-secret-key option. Do not use this option with the --aws-credential-file option. |
| --aws-secret-key=VALUE | Specifies the AWS secret access key to use to sign the request to CloudWatch. Must be used together with the --aws-access-key-id option. Do not use this option with --aws-credential-file option. |
| --aws-iam-role=VALUE | Specifies the IAM role used to provide AWS credentials. The value =VALUE is required. If no credentials are specified, the default IAM role associated with the EC2 instance is applied. Only one IAM role can be used. If no IAM roles are found, or if more than one IAM role is found, the script will return an error.  Do not use this option with the --aws-credential-file, --aws-access-key-id, or --aws-secret-key parameters. |
| --verify | Performs a test run of the script that collects the metrics, prepares a complete HTTP request, but does not actually call CloudWatch to report the data. This option also checks that credentials are provided. When run in verbose mode, this option outputs the metrics that will be sent to CloudWatch. |
| --verbose | Displays detailed information about what the script is doing. |
| --help | Displays usage information. |
| --version | Displays the version number of the script. |

***Example***

To get utilization statistics for the last 12 hours, run the following command:

./mon-get-instance-stats.pl --recent-hours=12

The following is an example response:

Instance metric statistics for the last 12 hours.

CPU Utilization

Average: 1.06%, Minimum: 0.00%, Maximum: 15.22%

Memory Utilization

Average: 6.84%, Minimum: 6.82%, Maximum: 6.89%

Swap Utilization

Average: N/A, Minimum: N/A, Maximum: N/A

Disk Space Utilization on /dev/xvda1 mounted as /

Average: 9.69%, Minimum: 9.69%, Maximum: 9.69%

**Viewing Your Custom Metrics in the Console**

After you successfully run the mon-put-instance-data.pl script, you can view your custom metrics in the Amazon CloudWatch console.

**To view custom metrics**

1. Run mon-put-instance-data.pl as described previously.
2. Open the CloudWatch console at <https://console.aws.amazon.com/cloudwatch/>.
3. Choose **View Metrics**.
4. For **Viewing**, your custom metrics posted by the script are displayed with the prefix System/Linux.

**Troubleshooting**

The **CloudWatchClient.pm** module caches instance metadata locally. If you create an AMI from an instance where you have run the monitoring scripts, any instances launched from the AMI within the cache TTL (default: six hours, 24 hours for Auto Scaling groups) emit metrics using the instance ID of the original instance. After the cache TTL time period passes, the script retrieves fresh data and the monitoring scripts use the instance ID of the current instance. To immediately correct this, remove the cached data using the following command:

**rm /var/tmp/aws-mon/*instance-id***