**Amazon EC2 Metrics and Dimensions**

Amazon Elastic Compute Cloud (Amazon EC2) sends metrics to CloudWatch for your EC2 instances. Basic (five-minute) monitoring is enabled by default. You can enable detailed (one-minute) monitoring. For information about additional metrics for Amazon EC2 instances that are in an Auto Scaling group, see [Auto Scaling Metrics and Dimensions](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/as-metricscollected.html).

For more information about how to monitor Amazon EC2, see [Monitoring Your Instances with CloudWatch](http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-cloudwatch.html)in the *Amazon EC2 User Guide for Linux Instances*.

**Metric Collection and Calculation on C5 and M5 Instances**

Because C5 and M5 instances use the Nitro hypervisor, they publish CloudWatch metrics differently than other instances, which use a Xen-based hypervisor. For more information, see [Nitro Hypervisor](https://aws.amazon.com/ec2/faqs/#ec2-hypervisor).

In basic monitoring for EC2 instances, seven pre-selected metrics are available at the five-minute frequency. When you use basic monitoring on any type of EC2 instance (using either hypervisor), the hypervisor measures five separate samples per metric during each five-minute interval. The way that these data points are published to CloudWatch depends on the type of hypervisor used by the instance.

* On Xen instances, these five samples are aggregated and reported to CloudWatch as a single data point at the end of the five-minute interval, with a time stamp corresponding to the beginning of the five-minute interval. The statistic set for this data point includes a SampleCount of 5.
* On Nitro instances, the data point is published to CloudWatch immediately after the first sample is taken during the interval. When each subsequent sample within the interval occurs, the statistic set for that data point is updated to reflect all the samples taken so far, but the time stamp remains the same. If you are graphing or monitoring the data as it is reported, the statistics for the current data point can appear to change as each sample is taken during the five-minute interval. Each of the five samples taken during the interval are included in the SampleCount statistic.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Time** | 1:01PM | 1:02PM | 1:03PM | 1:04PM | 1:05PM |
| **Sample Value** | 10 | 15 | 10 | 5 | 10 |
| **Published on Xen instances (with basic monitoring enabled)** | Nothing yet | Nothing yet | Nothing yet | Nothing yet | Average=10  Sum=50  Minimum=5  Maximum=15  SampleCount=5  Timestamp=1:00PM |
| **Published on Nitro instances (with basic monitoring enabled)** | Average=10  Sum=10  Minimum=10  Maximum=10  SampleCount=1  Timestamp=1:00PM | Average=12.5  Sum=25  Minimum=10  Maximum=15  SampleCount=2  Timestamp=1:00PM | Average=11.666  Sum=35  Minimum=10  Maximum=15  SampleCount=3  Timestamp=1:00PM | Average=10  Sum=40  Minimum=5  Maximum=15  SampleCount=4  Timestamp=1:00PM | Average=10  Sum=50  Minimum=5  Maximum=15  SampleCount=5  Timestamp=1:00PM |

In the example shown in the preceding table, basic monitoring is being used. A sample is taken each minute over a five-minute interval. On a Xen-based instance, nothing is published until the end of the five-minute interval. At this time (1:05PM), the five samples are aggregated and a single data point is written with a 1:00PM time stamp. This data point has a SampleCount of 5. This one data point contributes 50 to the Sumstatistic for the metric.

On a Nitro instance, the data point is first written to CloudWatch at 1:01PM with a time stamp of 1:00PM and a value of 10. At 1:02PM, the new sample of 15 causes the Average of the 1:00PM data point to change to 12.5, the average of the two samples. This also changes the Maximum to 15. The time stamp of this data point remains 1:00PM. At 1:03PM, the data point Average changes to 11.6. Finally, after the last sample at 1:05PM, the Average returns to 10. The final Average statistic of the data point is 10, but it has contributed 50 to the Sum as this five-minute period includes a SampleCount of 5 and each of the five samples is included in the Sum. The Minimum and Maximum values reflect the highest and lowest values of the samples taken during the interval.

Because of the way that Nitro instances collect and calculate the data, we recommend that you only set alarms with at least two evaluation periods or require "M out of N" breaching periods, where M is at least 2. This is because on Nitro instances, a single sample during one five-minute period can cause the data point to temporarily breach the threshold, even if subsequent samples in that period bring the final statistic set of this data point to within the threshold. This "mutable" data point in the current sample can cause the alarm to activate during the middle of the interval. For more information, see [Evaluating an Alarm](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/AlarmThatSendsEmail.html#alarm-evaluation).

**Amazon EC2 Metrics**

The following metrics are available by default from each EC2 instance. You can enable additional metrics by installing the CloudWatch agent on the instance. For more information about installing the CloudWatch agent on an instance, see [Collect Metrics and Logs from Amazon EC2 Instances and On-Premises Servers with the CloudWatch Agent](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/Install-CloudWatch-Agent.html). For more information about the additional metrics that can be collected, see[Metrics Collected by the CloudWatch Agent](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/metrics-collected-by-CloudWatch-agent.html).

The AWS/EC2 namespace includes the following CPU credit metrics for your T2 instances.

|  |  |
| --- | --- |
| **Metric** | **Description** |
| CPUCreditUsage | [T2 instances] The number of CPU credits spent by the instance for CPU utilization. One CPU credit equals one vCPU running at 100% utilization for one minute or an equivalent combination of vCPUs, utilization, and time (for example, one vCPU running at 50% utilization for two minutes or two vCPUs running at 25% utilization for two minutes).  CPU credit metrics are available at a five-minute frequency only. If you specify a period greater than five minutes, use the Sum statistic instead of the Average statistic.  Units: Credits (vCPU-minutes) |
| CPUCreditBalance | [T2 instances] The number of earned CPU credits that an instance has accrued since it was launched or started. For T2 Standard, theCPUCreditBalance also includes the number of launch credits that have been accrued.  Credits are accrued in the credit balance after they are earned, and removed from the credit balance when they are spent. The credit balance has a maximum limit, determined by the instance size. Once the limit is reached, any new credits that are earned are discarded. For T2 Standard, launch credits do not count towards the limit.  The credits in the CPUCreditBalance are available for the instance to spend to burst beyond its baseline CPU utilization.  When an instance is running, credits in the CPUCreditBalance do not expire. When the instance stops, the CPUCreditBalance does not persist, and all accrued credits are lost.  CPU credit metrics are available at a five-minute frequency only.  Units: Credits (vCPU-minutes) |
| CPUSurplusCreditBalance | [T2 Unlimited instances] The number of surplus credits that have been spent by a T2 Unlimited instance when its CPUCreditBalance is zero.  The CPUSurplusCreditBalance is paid down by earned CPU credits. If the number of surplus credits exceeds the maximum number of credits the instance can earn in a 24-hour period, the spent surplus credits above the maximum incur an additional charge.  Units: Credits (vCPU-minutes) |
| CPUSurplusCreditsCharged | [T2 Unlimited instances] The number of spent surplus credits that are not paid down by earned CPU credits, and thus incur an additional charge.  Spent surplus credits are charged when any of the following occurs:   * The spent surplus credits exceed the maximum number of credits the instance can earn in a 24-hour period. Spent surplus credits above the maximum are charged at the end of the hour. * The instance is stopped or terminated. * The instance is switched from Unlimited to Standard.   Units: Credits (vCPU-minutes) |

The AWS/EC2 namespace includes the following instance metrics.

|  |  |
| --- | --- |
| **Metric** | **Description** |
| CPUUtilization | 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.  To use the percentiles statistic, you must enable detailed monitoring.  Depending on the instance type, tools in your operating system can show a lower percentage than CloudWatch when the instance is not allocated a full processor core.  Units: Percent |
| DiskReadOps | Completed read operations from all instance store volumes available to the instance in a specified period of time.  To calculate the average I/O operations per second (IOPS) for the period, divide the total operations in the period by the number of seconds in that period.  Units: Count |
| DiskWriteOps | Completed write operations to all instance store volumes available to the instance in a specified period of time.  To calculate the average I/O operations per second (IOPS) for the period, divide the total operations in the period by the number of seconds in that period.  Units: Count |
| DiskReadBytes | Bytes read from all instance store volumes available to the instance.  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.  The number reported is the number of bytes received during the period. If you are using basic (five-minute) monitoring, you can divide this number by 300 to find Bytes/second. If you have detailed (one-minute) monitoring, divide it by 60.  Units: Bytes |
| DiskWriteBytes | Bytes written to all instance store volumes available to the instance.  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.  The number reported is the number of bytes received during the period. If you are using basic (five-minute) monitoring, you can divide this number by 300 to find Bytes/second. If you have detailed (one-minute) monitoring, divide it by 60.  Units: Bytes |
| NetworkIn | The number of bytes received on all network interfaces by the instance. This metric identifies the volume of incoming network traffic to a single instance.  The number reported is the number of bytes received during the period. If you are using basic (five-minute) monitoring, you can divide this number by 300 to find Bytes/second. If you have detailed (one-minute) monitoring, divide it by 60.  Units: Bytes |
| NetworkOut | The number of bytes sent out on all network interfaces by the instance. This metric identifies the volume of outgoing network traffic from a single instance.  The number reported is the number of bytes sent during the period. If you are using basic (five-minute) monitoring, you can divide this number by 300 to find Bytes/second. If you have detailed (one-minute) monitoring, divide it by 60.  Units: Bytes |
| NetworkPacketsIn | 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.  Units: Count  Statistics: Minimum, Maximum, Average |
| NetworkPacketsOut | 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.  Units: Count  Statistics: Minimum, Maximum, Average |

The AWS/EC2 namespace includes the following status checks metrics. By default, status check metrics are available at a 1-minute frequency at no charge. For a newly-launched instance, status check metric data is only available after the instance has completed the initialization state (within a few minutes of the instance entering the running state). For more information about EC2 status checks, see [Status Checks For Your Instances](http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/monitoring-system-instance-status-check.html).

|  |  |
| --- | --- |
| **Metric** | **Description** |
| StatusCheckFailed | Reports whether the instance has passed both the instance status check and the system status check in the last minute.  This metric can be either 0 (passed) or 1 (failed).  By default, this metric is available at a 1-minute frequency at no charge.  Units: Count |
| StatusCheckFailed\_Instance | Reports whether the instance has passed the instance status check in the last minute.  This metric can be either 0 (passed) or 1 (failed).  By default, this metric is available at a 1-minute frequency at no charge.  Units: Count |
| StatusCheckFailed\_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).  By default, this metric is available at a 1-minute frequency at no charge.  Units: Count |

Amazon CloudWatch data for a new EC2 instance typically becomes available within one minute of the end of the first period of time requested (the *aggregation period*) in the query. You can set the period—the length of time over which statistics are aggregated—with the Period parameter. For more information on periods, see [Periods](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/cloudwatch_concepts.html#CloudWatchPeriods).

You can use the currently available dimensions for EC2 instances (for example, ImageId or InstanceType) to refine the metrics returned. For information about the dimensions you can use with EC2, see [Dimensions for Amazon EC2 Metrics](https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/ec2-metricscollected.html#ec2-metric-dimensions).

The AWS/EC2 namespace includes the following Amazon EBS metrics for your C5 and M5 instances.

|  |  |
| --- | --- |
| **Metric** | **Description** |
| EBSReadOps | Completed read operations from all Amazon EBS volumes attached to the instance in a specified period of time.  To calculate the average read I/O operations per second (Read IOPS) for the period, divide the total operations in the period by the number of seconds in that period. If you are using basic (five-minute) monitoring, you can divide this number by 300 to calculate the Read IOPS. If you have detailed (one-minute) monitoring, divide it by 60.  Unit: Count |
| EBSWriteOps | Completed write operations to all EBS volumes attached to the instance in a specified period of time.  To calculate the average write I/O operations per second (Write IOPS) for the period, divide the total operations in the period by the number of seconds in that period. If you are using basic (five-minute) monitoring, you can divide this number by 300 to calculate the Write IOPS. If you have detailed (one-minute) monitoring, divide it by 60.  Unit: Count |
| EBSReadBytes | Bytes read from all EBS volumes attached to the instance in a specified period of time.  The number reported is the number of bytes read during the period. If you are using basic (five-minute) monitoring, you can divide this number by 300 to find Read Bytes/second. If you have detailed (one-minute) monitoring, divide it by 60.  Unit: Bytes |
| EBSWriteBytes | Bytes written to all EBS volumes attached to the instance in a specified period of time.  The number reported is the number of bytes written during the period. If you are using basic (five-minute) monitoring, you can divide this number by 300 to find Write Bytes/second. If you have detailed (one-minute) monitoring, divide it by 60.  Unit: Bytes |
| EBSIOBalance% | Available only for the smaller C5 and M5 instance sizes. Provides information about the percentage of I/O credits remaining in the burst bucket. This metric is available for basic monitoring only.  The Sum statistic is not applicable to this metric.  Unit: Percent |
| EBSByteBalance% | Available only for the smaller C5 and M5 instance sizes. Provides information about the percentage of throughput credits remaining in the burst bucket. This metric is available for basic monitoring only.  The Sum statistic is not applicable to this metric.  Unit: Percent |

**Dimensions for Amazon EC2 Metrics**

If you're using Detailed Monitoring, you can filter the EC2 instance data using any of the dimensions in the following table.

|  |  |
| --- | --- |
| **Dimension** | **Description** |
| AutoScalingGroupName | This dimension filters the data you request for all instances in a specified capacity group. An *Auto Scaling group* is a collection of instances you define if you're using Auto Scaling. This dimension is available only for Amazon EC2 metrics when the instances are in such an Auto Scaling group. Available for instances with Detailed or Basic Monitoring enabled. |
| ImageId | This dimension filters the data you request for all instances running this Amazon EC2 Amazon Machine Image (AMI). Available for instances with Detailed Monitoring enabled. |
| InstanceId | This dimension filters the data you request for the identified instance only. This helps you pinpoint an exact instance from which to monitor data. |
| InstanceType | This dimension filters the data you request for all instances running with this specified instance type. This helps you categorize your data by the type of instance running. For example, you might compare data from an m1.small instance and an m1.large instance to determine which has the better business value for your application. Available for instances with Detailed Monitoring enabled. |