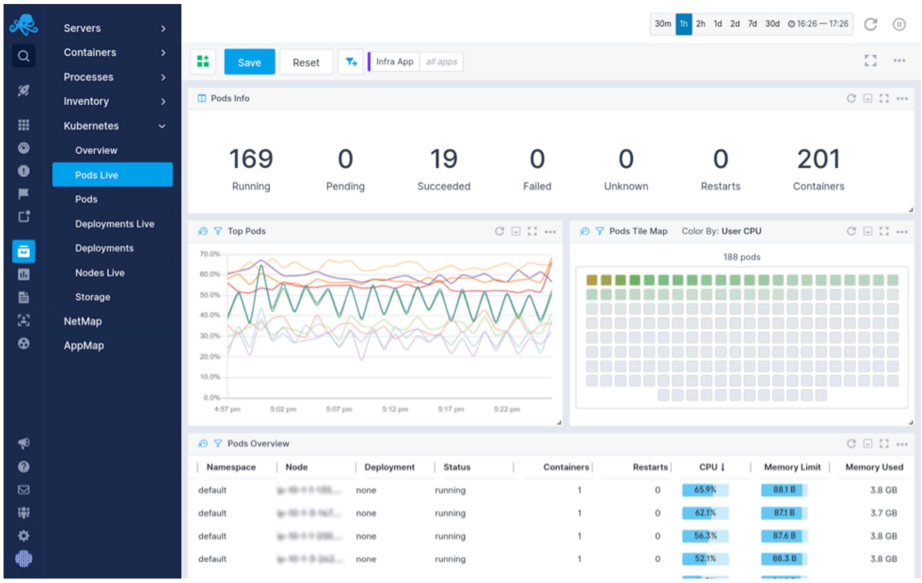
title: Kubernetes Monitoring Integration description: Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation. To start monitoring Kubernetes with Sematext, you only need to install a tiny agent that adds basically no CPU or memory overhead.

Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation. To start monitoring Kubernetes with Sematext, you only need to install a tiny agent that adds basically no CPU or memory overhead.

Monitoring Kubernetes with Sematext

Sematext Monitoring will give you detailed insights into your cluster's health, performance metrics, resource counts amongst other important metrics. Speaking of metrics, check out thispage for a summarized list of the key metrics you can follow with Sematext as well as a short explanation for each one of them.



169	0	19	0	0	0	201
Running	Pending	Succeeded	Failed	Unknown	Restarts	Containers

Namespace	Node	Deployment	Status	Containers	Restarts	CPU ↓	Memory Limit	Memory Used
default	[redacted]	none	running	1	0	65.9%	88.1 B	3.8 GB
default	[redacted]	none	running	1	0	62.1%	87.1 B	3.7 GB
default	[redacted]	none	running	1	0	56.3%	87.6 B	3.8 GB
default	[redacted]	none	running	1	0	52.1%	88.3 B	3.8 GB

Figure 1: alt_text

Helm Chart

To start monitoring Kubernetes with Sematext install the Sematext Agent. The easiest way to do that is with a Helm chart. It's available in the official charts repo and it will install to all nodes in your cluster. To install it run the following command:

```
helm install --name sematext-agent \
  --set containerToken=<YOUR_CONTAINER_TOKEN> \
  --set infraToken=<YOUR_INFRA_TOKEN> \
  --set logsToken=<YOUR_LOGS_TOKEN> \
  --set region=<"US" or "EU"> \
  stable/sematext-agent
```

Check out github for more details.

Sematext Operator

You can also install Sematext Operator using this command:

```
kubectl apply -f https://raw.githubusercontent.com/sematext/sematext-operator/master/bundle.
```

After the installation has finished you can create the SematextAgent resource that deploys the agent to all the nodes in your cluster.

```
apiVersion: sematext.com/v1alpha1
kind: SematextAgent
metadata:
  name: sematext-agent
spec:
  region: <"US" or "EU">
  containerToken: YOUR_CONTAINER_TOKEN
  logsToken: YOUR_LOGS_TOKEN
  infraToken: YOUR_INFRA_TOKEN
```

For those looking for a more hands-on approach, there's a manual installation procedure with `kubectl`.

Shipping Kubernetes logs to Sematext

Due to its nature, Kubernetes can be difficult to debug and without proper tooling this process will take a lot longer than it has too. Sematext helps you shed light on what caused the anomaly that led to the crash.

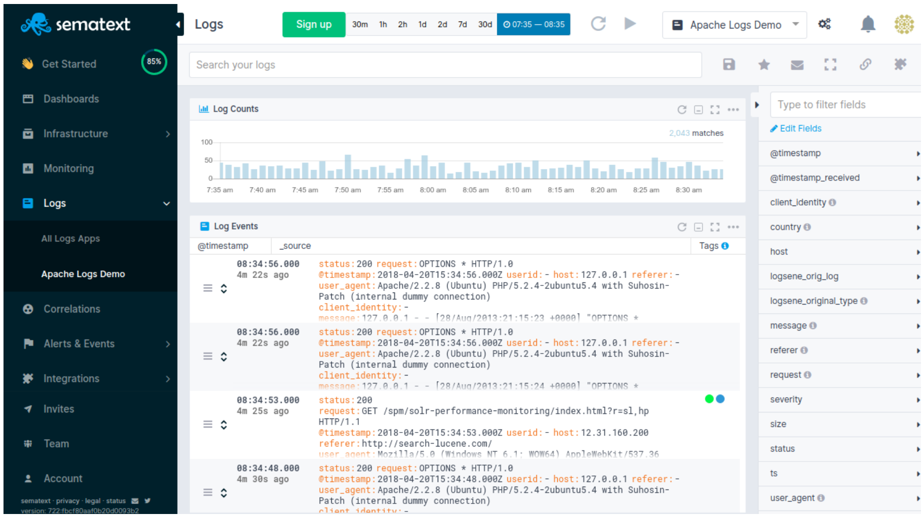
To configure Kubernetes log shipping we're going to use Helm.

Helm

To install Logagent with Helm you'll need to run the following command:

```
helm install st-logagent \
  --set logsToken=<YOUR_LOGS_TOKEN> \
  --set region=<US or EU> \
  stable/sematext-agent
```

Deleting Logagent can be done with:



```
08:34:56.000   status:200 request:OPTIONS * HTTP/1.0
4m 22s ago     @timestamp:2018-04-20T15:34:56.000Z userid:- host:127.0.0.1 referer:-
               user_agent:Apache/2.2.8 (Ubuntu) PHP/5.2.4-2ubuntu5.4 with Suhosin-
               Patch (internal dummy connection)
               client_identity:-
               message:127.0.0.1 - - [28/Aug/2013:21:15:23 +0000] "OPTIONS *

08:34:56.000   status:200 request:OPTIONS * HTTP/1.0
4m 22s ago     @timestamp:2018-04-20T15:34:56.000Z userid:- host:127.0.0.1 referer:-
               user_agent:Apache/2.2.8 (Ubuntu) PHP/5.2.4-2ubuntu5.4 with Suhosin-
               Patch (internal dummy connection)
               client_identity:-
               message:127.0.0.1 - - [28/Aug/2013:21:15:24 +0000] "OPTIONS *

08:34:53.000   status:200
4m 25s ago     request:GET /spm/solr-performance-monitoring/index.html?r=sl,hp
               HTTP/1.1
               @timestamp:2018-04-20T15:34:53.000Z userid:- host:12.31.160.200
               referer:http://search-lucene.com/
               user_agent:Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36

08:34:48.000   status:200 request:OPTIONS * HTTP/1.0
4m 30s ago     @timestamp:2018-04-20T15:34:48.000Z userid:- host:127.0.0.1 referer:-
               user_agent:Apache/2.2.8 (Ubuntu) PHP/5.2.4-2ubuntu5.4 with Suhosin-
               Patch (internal dummy connection)
               client_identity:-
```

Figure 2: alt_text

```
helm delete st-logagent
```

If you are looking to use a different type of integration you can check out this page.

Kubernetes Metrics

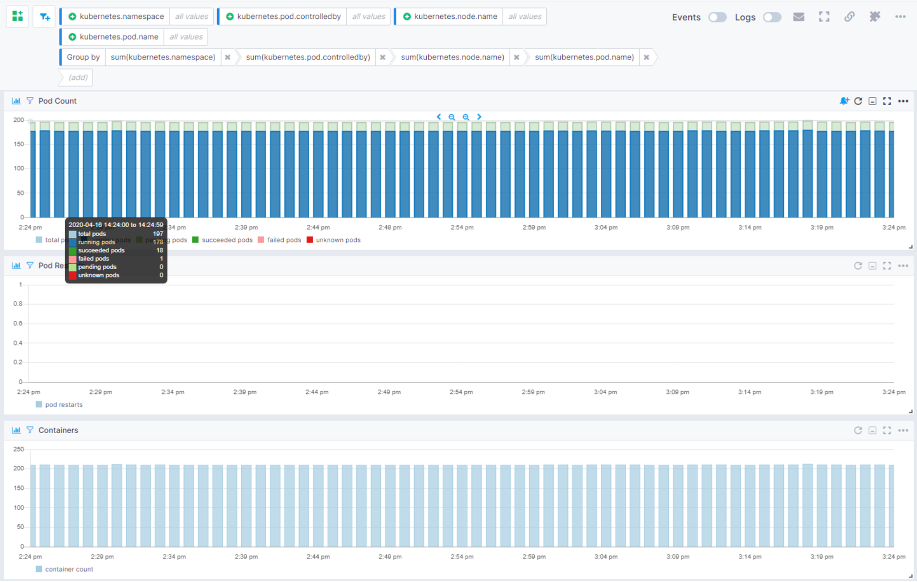
Container and Kubernetes metrics are collected along with labels and tags, which are exposed in the UI to allow slicing and dicing and building of custom dashboards.

Pod Metrics

- Pod count - The total nodes in the cluster
- Pod restarts - The total number of pods scheduled across nodes
- Containers count - The total number ofcontainers
- Succeeded pods - The number of pods that are successfully scheduled
- Failed pods - The number of failed pods
- Unknown pods - The number of pods that are in unknown state
- Pending pods - The number of pods in pending state
- Running pods - Reflects the current number of running pods

Deployment

- Current replicas - The number of active deployment replicas



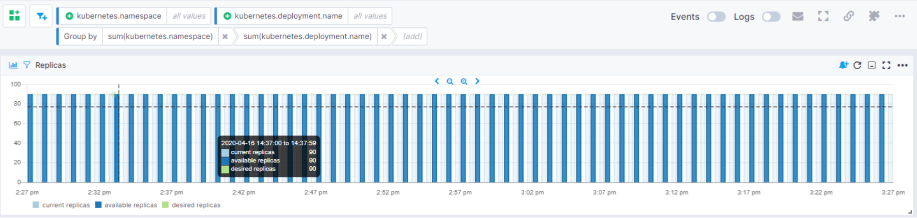
kubernetes.namespace all values | kubernetes.pod.controlledby all values | kubernetes.node.name all values | kubernetes.pod.name all values

Group by sum(kubernetes.namespace) ✖ sum(kubernetes.pod.controlledby) ✖ sum(kubernetes.node.name) ✖ sum(kubernetes.pod.name) ✖ (add)

2020-04-16 14:24:00 to 14:24:59
total pods 197
running pods 178
succeeded pods 18
failed pods 1
pending pods 0
unknown pods 0

Figure 3: alt_text

- Available replicas - The number of pod instances targeted by the deployment
- Desired replicas - The number of non-terminated pods targeted by the deployment that have the desired template specification



kubernetes.namespace all values | kubernetes.deployment.name all values

Group by sum(kubernetes.namespace) ✖ sum(kubernetes.deployment.name) ✖ (add)

2020-04-16 14:37:00 to 14:37:59
current replicas 90
available replicas 90
desired replicas 90

Figure 4: alt_text

Storage

- Read bytes - The number of bytes read from the disk
- Read time - The total amount of time (in nanoseconds) between read request dispatch and request completion
- Read wait time - The total amount of time the read I/O operations for the container spent waiting in the scheduler queues
- Write bytes - The number of bytes written to disk
- Write time - The total amount of time (in nanoseconds) between write

request dispatch and request completion

- Write wait time - Total amount of time the write I/O operations for the container spent waiting in the scheduler queues

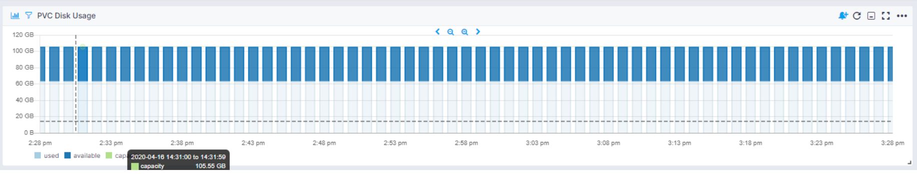


Figure 5: alt_text

Network

- Received bytes - Received amount of bytes on the network interface
- Received packets - Received amount of packets on the network interface
- Received errors - Received amount of errors on the network interface
- Dropped ingress packets - The amount of dropped inbound packets on the network interface
- Transmitted bytes - Transmitted amount of bytes on the network interface
- Transmitted packets - Transmitted amount of packets on the network interface
- Transmitted errors - Transmitted amount of errors on the network interface
- Dropped egress packets - The amount of dropped outbound packets on the network interface

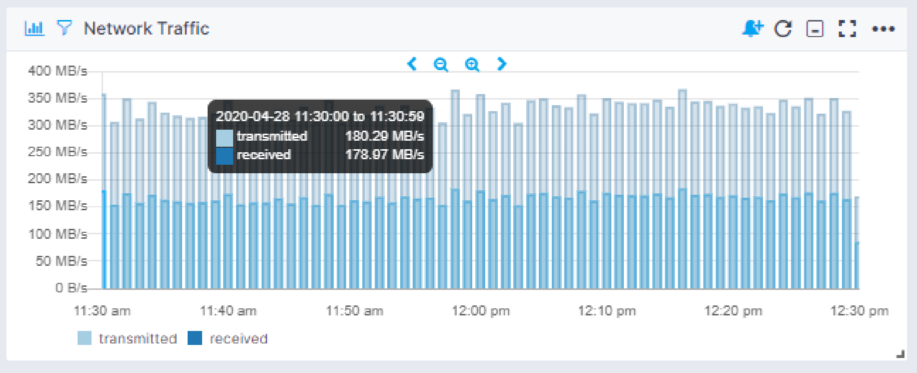


Figure 6: alt_text

Memory

- Memory fail counter - The number of times that memory cgroup limit was exceeded

- Memory limit - Designates the max allowed memory limit for the container cgroup
- Memory pages in - The number of events each time the page is accounted to the container cgroup
- Memory pages out - The number of events each time a page is unaccounted from the container cgroup
- Memory pages fault - Represents the number of page faults accounted the cgroup
- Swap size - The number of bytes of swap usage

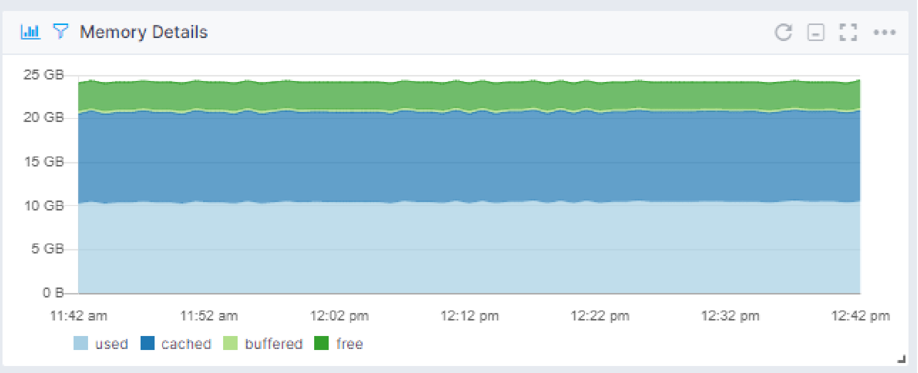


Figure 7: alt_text

CPU

- Cpu usage - The container CPU usage in %
- Throttled time - The total amount of time that processes have been throttled in the container cgroup

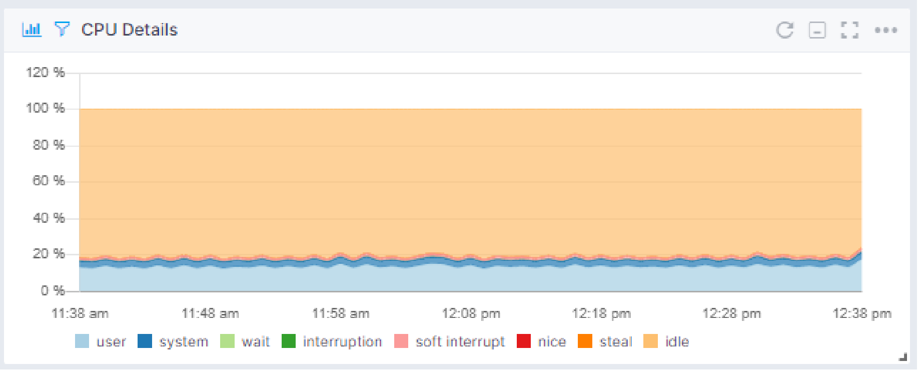


Figure 8: alt_text

Metrics Fields

Name	Type	Unit	Numeric Type	Label	Description
kubernetes.pod.restarts	counter	ns	long	pod restarts	number of pod restarts
kubernetes.pod.container.count	gauge	ns	long	container count	number of containers inside pod
kubernetes.pod.count	gauge	ns	long	pod count	pod count which is always equal to one
kubernetes.pod.count.succeeded	gauge	ns	long	succeeded pod count	equal to one if all containers inside pod have terminated in success
kubernetes.pod.count.failed	gauge	ns	long	failed pod count	equal to one if all containers inside pod have terminated and at least one container has terminated in failure
kubernetes.pod.count.unknown	gauge	ns	long	unknown pod count	equal to one if pod state can’t be obtained
kubernetes.pod.count.pending	gauge	ns	long	pending pod count	equal to one if the pod has been accepted by the scheduler and his containers are waiting to be created
kubernetes.pod.count.running	gauge	ns	long	running pod count	equal to one if the pod has been scheduled on a node and at least one of his containers is running
kubernetes.deployment.count	gauge	ns	long	deployment count	deployment count which is always equal to one

Name	Type	Unit	Numeric Type	Label	Description
kubernetes.deployment.replicas	gauge	ns	long	replica count	number of active replicas
kubernetes.deployment.replicas.avail	gauge	ns	long	available replica count	number of available replicas. Replicas are marked as available if they are passing the health check
kubernetes.deployment.replicas.desired	gauge	ns	long	desired replica count	number of desired replicas as defined in the deployment
kubernetes.pvc.available	gauge	bytes	long	available bytes	number of available bytes in the volume
kubernetes.pvc.used	gauge	bytes	long	used bytes	number of used bytes in the volume
kubernetes.pvc.capacity	gauge	bytes	long	volume capacity	the capacity in bytes of the volume
kubernetes.cluster.pod.count	gauge	ns	long	total pod count	number of pods in the cluster
kubernetes.cluster.deployment.count	gauge	ns	long	total deployment count	number of deployments in the cluster
kubernetes.cluster.node.count	gauge	ns	long	total node count	number of node comprising the cluster

Sematext Agent

The Sematext Agent offers a versatile container engine monitoring and visibility solution that is easy to customize.

Kubernetes Settings

KUBERNETES_ENABLED

Specifies if the Kubernetes monitoring functionality is active. Default value is true. To disable Kubernetes collector set KUBERNETES_ENABLED=false.

KUBERNETES_EVENTS_NAMESPACE

Designates a namespace for Kubernetes event watcher. By default all namespaces are watched for Kubernetes events and forwarded to event/log receivers.

KUBERNETES_NAMESPACES

Defines the comma separated list of namespaces that are queried for Kubernetes resources such as pods or deployments. By default all namespaces are fetched. You can adjust specific namespaces such as KUBERNETES_NAMESPACES=default,kube-system.

KUBERNETES_INTERVAL

Defines the collection interval for Kubernetes resources (default 10s)

KUBERNETES_CLUSTER_ID

Uniquely identifies the cluster where agent is deployed

KUBERNETES_KUBELET_AUTH_TOKEN

Specifies the path for account service token

KUBERNETES_KUBELET_CA_PATH

Determines the file path for the certificate authority utilized during TLS verification

KUBERNETES_KUBELET_CERT_PATH

Determines the file path for the certificate file utilized during TLS verification

KUBERNETES_KUBELET_KEY_PATH

Determines the file path for the private key utilized during TLS verification

KUBERNETES_KUBELET_INSECURE_SKIP_TLS_VERIFY

Indicates whether to skip TLS verification

KUBERNETES_KUBELET_METRICS_PORT

Specifies the port where kubelet Prometheus metrics are exposed (default 10250)

You can find a complete list of **Environment Variables** available at this link.

Containers are discovered from ***cgroupfs*** and the metrics are fetched directly through ***cgroup*** controllers. Check out thispage for a complete list of the metrics shipped by the Sematext Agent.