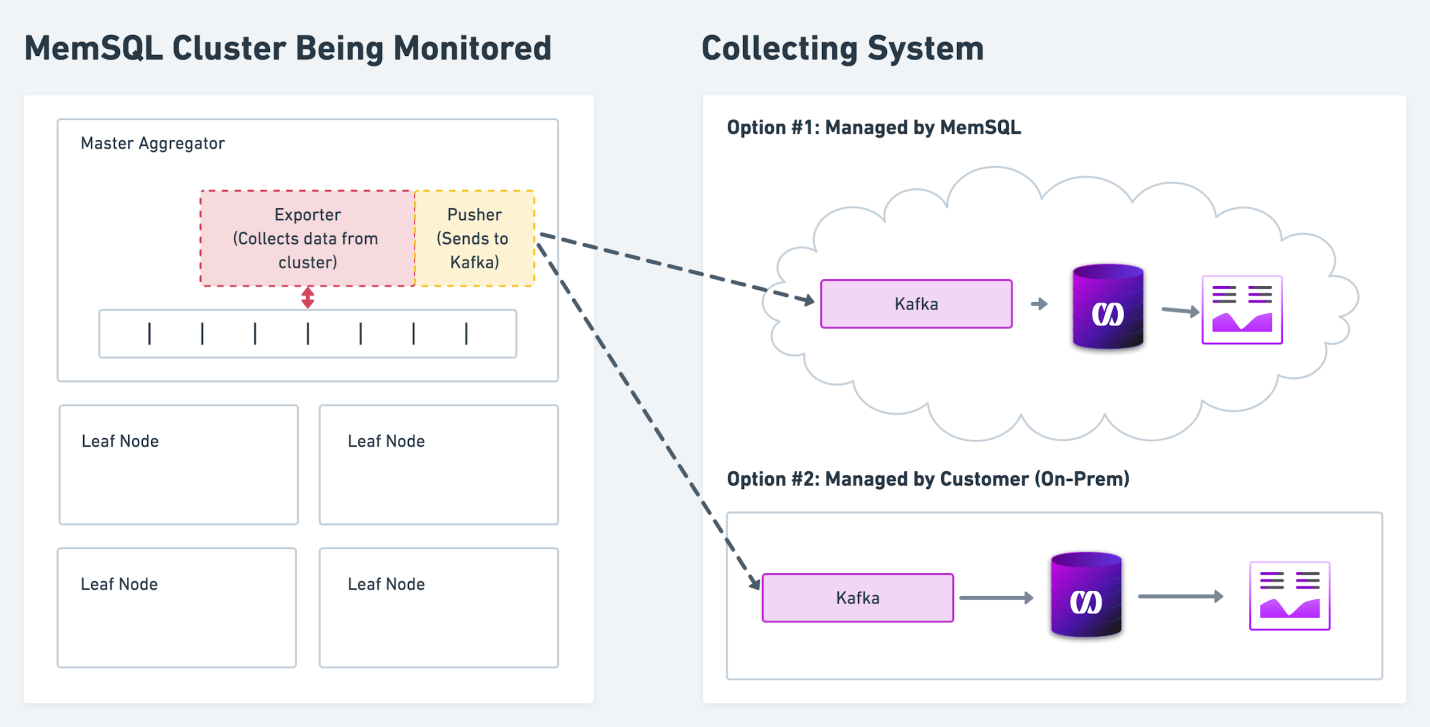
The MemSQL *monitored* cluster pushes event data to the MemSQL *collecting* cluster where it collected and stored. When these event data are then analyzed through the associated dashboards, trends can be spotted and, if necessary, remediated.



Provided dashboards include:

* A detailed cluster view, akin to a “birds-eye view” of a single cluster
* Granular breakdown of memory usage for a host
* Active session history, including aggregated resource consumption by activity and activity type
* Activity history, including historical resource consumption by a specific activity
* MemSQL status and variables view, with collected status variables from each host in the cluster
* Node breakout with system metrics from each host in the cluster

## Prerequisites

1. A MemSQL cluster to monitor (the *monitored* cluster)
2. A MemSQL to store monitoring data (the *collecting* cluster)

**Note**: This can be the same cluster as the monitored cluster

1. A Kafka cluster with:
   1. The ability to create two new [topics](https://kafka.apache.org/documentation/#basic_ops_add_topic)
   2. Brokers configured so that they are accessible by both the monitored and collecting clusters
2. Grafana installed and running, and the collecting cluster is accessible to it.
3. The [provided Grafana dashboard JSON layout files](https://drive.google.com/drive/folders/1sBZfoqKT4nhA_kXuKOx-RIAMHp0PFra7?usp=sharing).

## Port Configuration

|  |  |
| --- | --- |
| **Default Port** | **Used by** |
| 3000 | Grafana |
| 3306 | MemSQL |
| 8080 | MemSQL Studio |
| 9092 | Kafka |
| 9104 | memsql\_exporter |

# Configure Components

## MemSQL Exporter

MemSQL Exporter (memsql\_exporter) is a satellite process that collects data about a running cluster and funnels it to either a MemSQL Pusher (memsql\_pusher) or a Prometheus process which pulls from the Exporter and sends it to a Kafka queue.

## MemSQL Pusher

MemSQL Pusher (memsql\_pusher) is a satellite process that takes input from one or more exporters (e.g. memsql\_exporter, or other Prometheus ecosystem tools) processes and sends the collected data to the <org>\_metrics and <org>\_blobs Kafka topics.

## Kafka

1. Create two new Kafka topics. Note the <org>\_ prefix on each one.
   1. <org>\_metrics
   2. <org>\_blobs
2. **Optional**: Enable security for both of these topics.
   1. SASL for username and password
   2. SSL for wire encryption
3. Confirm that the Kafka broker be accessed from the collecting Master Aggregator.

**Note**: Potential sources of issues can include:

* 1. Improperly defined AWS security groups
  2. Improperly defined iptables

1. Confirm that all Kafka brokers are accessible from all hosts in the monitoredMemSQL cluster.

telnet <broker ip> <kafka broker port>

# Monitor the MemSQL Cluster

A MemSQL database is required to capture the metric data exposed by memsql\_exporter. The collecting MemSQL cluster can be the same as, or separate from, the monitoredMemSQL cluster.

## Configure the Collecting Database

On the *collecting* **Master Aggregator**:

1. Run monitoring.sql to create a monitoring database and associated tables
   1. [monitoring.sql](https://drive.google.com/open?id=1p1TUo3QjLS8iSISjTk5JYplAc55pxAI7)
2. Create a metrics and blobs pipeline using the correct Kafka broker address and security credentials.

\*NOTE\* the topic names includes the organization name. This will need to match the organization name

## Configure Grafana

Create a monitoring datasource. This requires both admin privileges and SELECT and EXECUTE grants to the monitoring database as per: <https://docs.memsql.com/sql-reference/v6.8/grant/>

1. Review the [guide on how to use the Grafana UI](https://grafana.com/docs/guides/getting_started/#how-to-add-a-data-source).
2. Configure the provisioning/datasources/ folder using the following sample datasource YaML.

# View the Dashboards

When all components are configured properly, the following dashboards can be used to monitor MemSQL cluster health, including:

* [**Detailed Cluster View**](http://example.com/d/detailed-cluster-view/detailed-cluster-view): Akin to a “birds-eye” view of a single cluster
* [**Memory Usage**](http://example.com/d/memory-usage/memory-usage)**:** A granular breakdown of memory usage for a host
* [**Active Session History**](http://example.com/d/active-session-history/active-session-history)**:** Aggregate resource consumption by activity and activity type
* [**Activity History**](http://example.com/d/activity-history/activity-history): Historical resource consumption by a specific activity
* [**MemSQL Status and Variables View**](http://example.com/d/memsql-view-status-variables/memsql-view-status-and-variables): Collected status variables from each host in the cluster
* [**Node Breakout**](http://example.com/d/node-metrics-breakout/node-metrics-breakout)**:** System metrics from each host