# Azure services in Grafana

## Uses Of Grafana

- **Visualize** → Fast and flexible visualizations with a multitude of options allow you to visualize your data any way you want.
- **Dynamic Dashboards** → Create dynamic & reusable dashboards with template variables that appear as dropdowns at the top of the dashboard.
- **Explore Metrics** → Explore your data through ad-hoc queries and dynamic drill-down. Split view and compare different time ranges, queries, and data sources side by side.
- **Explore Logs** → Experience the magic of switching from metrics to logs with preserved label filters. Quickly search through all your logs or streaming them live.
- **Alerting** → Visually define alert rules for your most important metrics. Grafana will continuously evaluate and send notifications to systems like Slack, PagerDuty, VictorOps, OpsGenie.
- **Mixed Data Sources** → Mix different data sources in the same graph! You can specify a data source on a per-query basis. This works for even custom data sources.
- **Annotations** → Annotate graphs with rich events from different data sources. Hover over events shows you the full event metadata and tags.
- **Ad-hoc Filters** → Ad-hoc filters allow you to create new key/value filters on the fly, which are automatically applied to all queries that use that data source.

## Features

- Support for all the Azure Monitor metrics
- includes support for the latest API version that allows multi-dimensional filtering for the Storage and SQL metrics.
- Automatic time grain mode which will group the metrics by the most appropriate time grain value depending on whether you have zoomed in to look at fine-grained metrics or zoomed out to look at an overview.
- Application Insights metrics
- Write raw log analytics queries, and select x-axis, y-axis, and grouped values manually.
- Automatic time grain support
- Support for Log Analytics (both for Azure Monitor and Application Insights)
- You can combine metrics from both services in the same graph.

## Observe all your telemetry data in one place

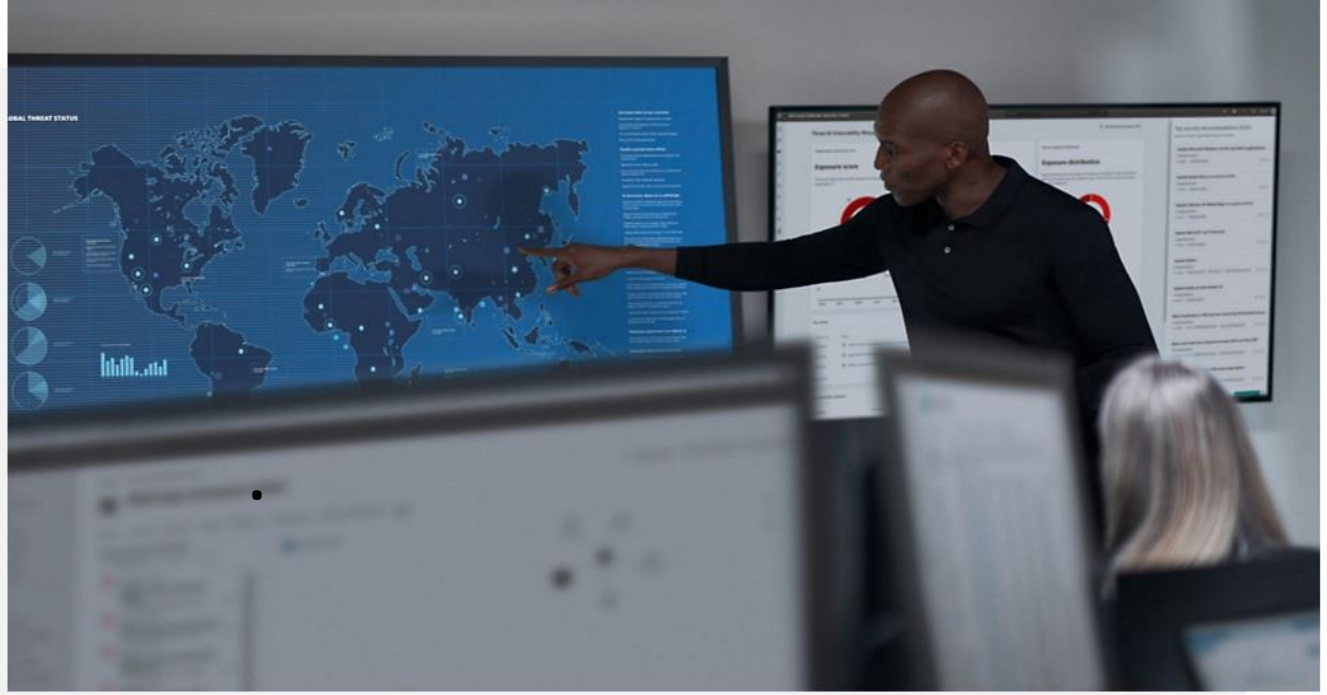
Access a wide variety of data sources supported by Grafana Enterprise and connect to your data stores in Azure and elsewhere. Combine charts, logs, and alerts to create one holistic view of your application and infrastructure. Correlate information across multiple datasets.

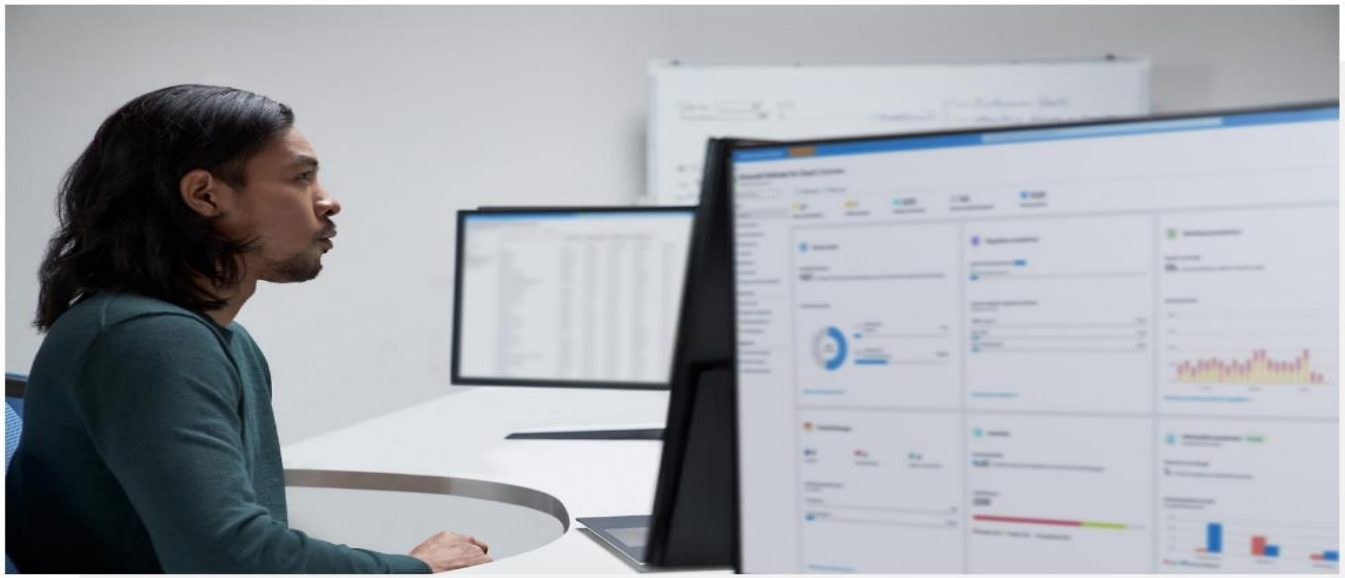

## Collaborate easily with your team

Share Grafana dashboards with people inside and outside of your organization. Allow others to contribute to solution monitoring and troubleshooting.

## Secure access with Microsoft Entra ID

Centralize identity management in Microsoft Entra ID. Control which users can use a Grafana instance and leverage managed identities to access Azure data sources from services such as Azure Monitor.





## Create dashboards with ease

Get started quickly with prebuilt dashboards for Azure services and import existing charts directly from the Azure Portal.

# Monitor your Azure services in Grafana

## Set up Grafana

Follow these steps to set up Grafana.

## Set up Azure Managed Grafana

Azure Managed Grafana is optimized for the Azure environment and works seamlessly with Azure Monitor. You can:

- Manage user authentication and access control by using Microsoft Entra identities.
- Pin charts from the Azure portal directly to Azure Managed Grafana dashboards.

Use this quickstart guide to create an Azure Managed Grafana workspace by using the Azure portal.

## Set up Grafana locally

To set up a local Grafana server, download and install Grafana in your local environment.

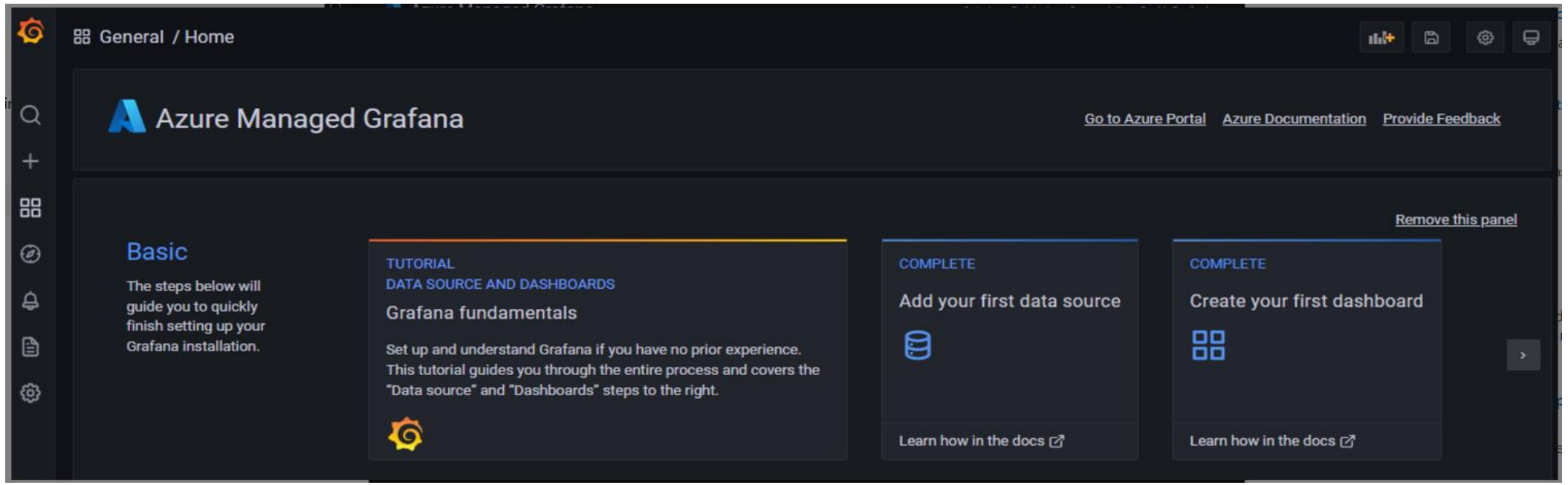
## Sign in to Grafana

> **Important**
>
> Internet Explorer and the older Microsoft Edge browsers aren't compatible with Grafana. You must use a chromium-based browser including Microsoft Edge. For more information, see Supported web browsers for Grafana.

Sign in to Grafana by using the endpoint URL of your Azure Managed Grafana workspace or your server's IP address.

# Configure an Azure Monitor data source plug-in

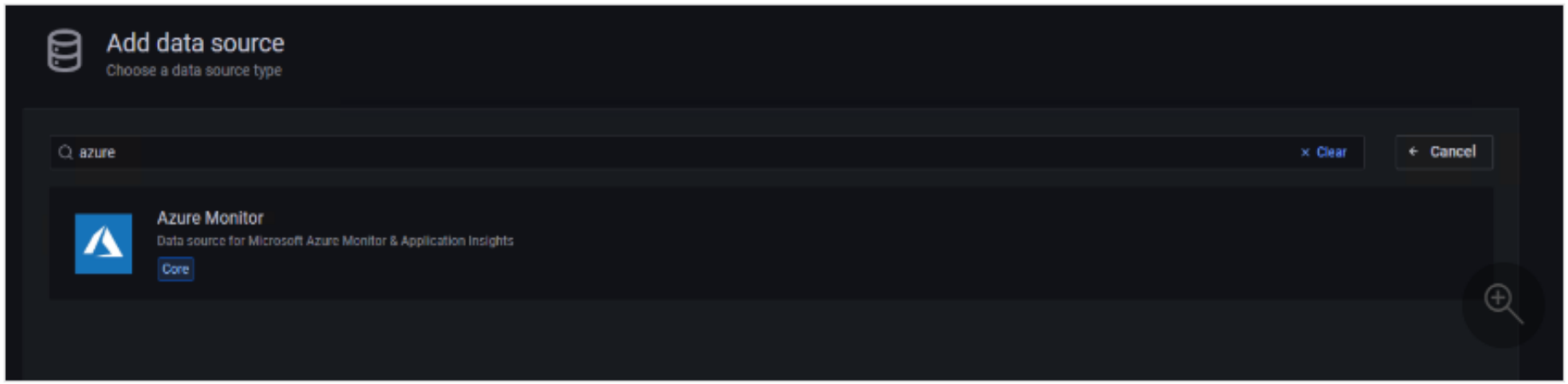
Azure Managed Grafana includes an Azure Monitor data source plug-in. By default, the plug-in is preconfigured with a managed identity that can query and visualize monitoring data from all resources in the subscription in which the Grafana workspace was deployed. Skip ahead to the section "Build a Grafana dashboard."




You can expand the resources that can be viewed by your Azure Managed Grafana workspace by configuring additional permissions to assign the included managed identity the Monitoring Reader role on other subscriptions or resources.

If you're using an instance that isn't Azure Managed Grafana, you have to set up an Azure Monitor data source.

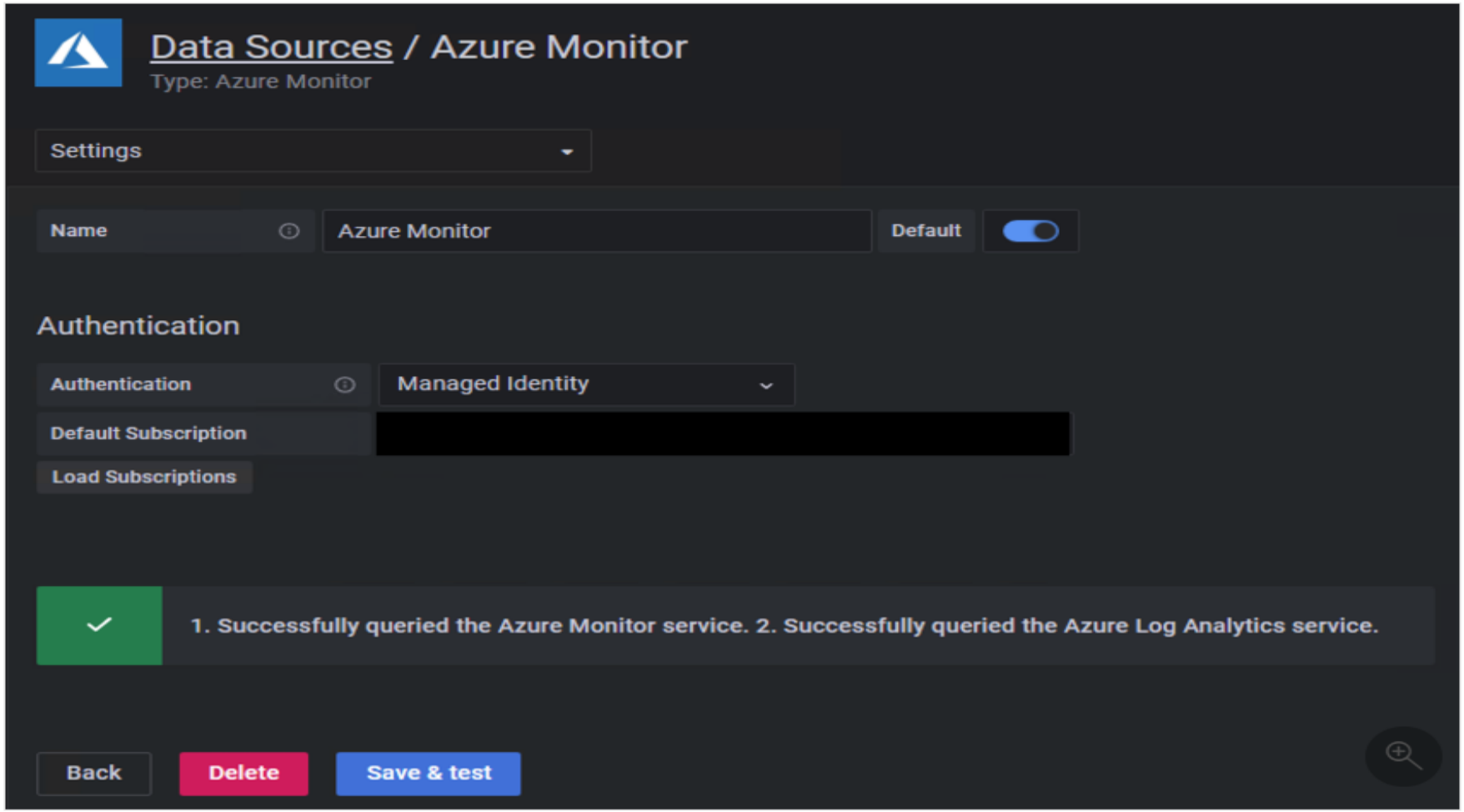
1. Select **Add data source**, filter by the name **Azure**, and select the **Azure Monitor** data source.




2. Pick a name for the data source and choose between managed identity or app registration for authentication.

If you're hosting Grafana on your own Azure Virtual Machines or Azure App Service instance with managed identity enabled, you can use this approach for authentication. However, if your Grafana instance isn't hosted on Azure or doesn't have managed identity enabled, you'll need to use app registration with an Azure service principal to set up authentication.

2. Select **Save & test** and Grafana will test the credentials. You should see a message similar to the following one.



## Use app registration

1. Create a service principal. Grafana uses a Microsoft Entra service principal to connect to Azure Monitor APIs and collect data. You must create, or use an existing service principal, to manage access to your Azure resources:

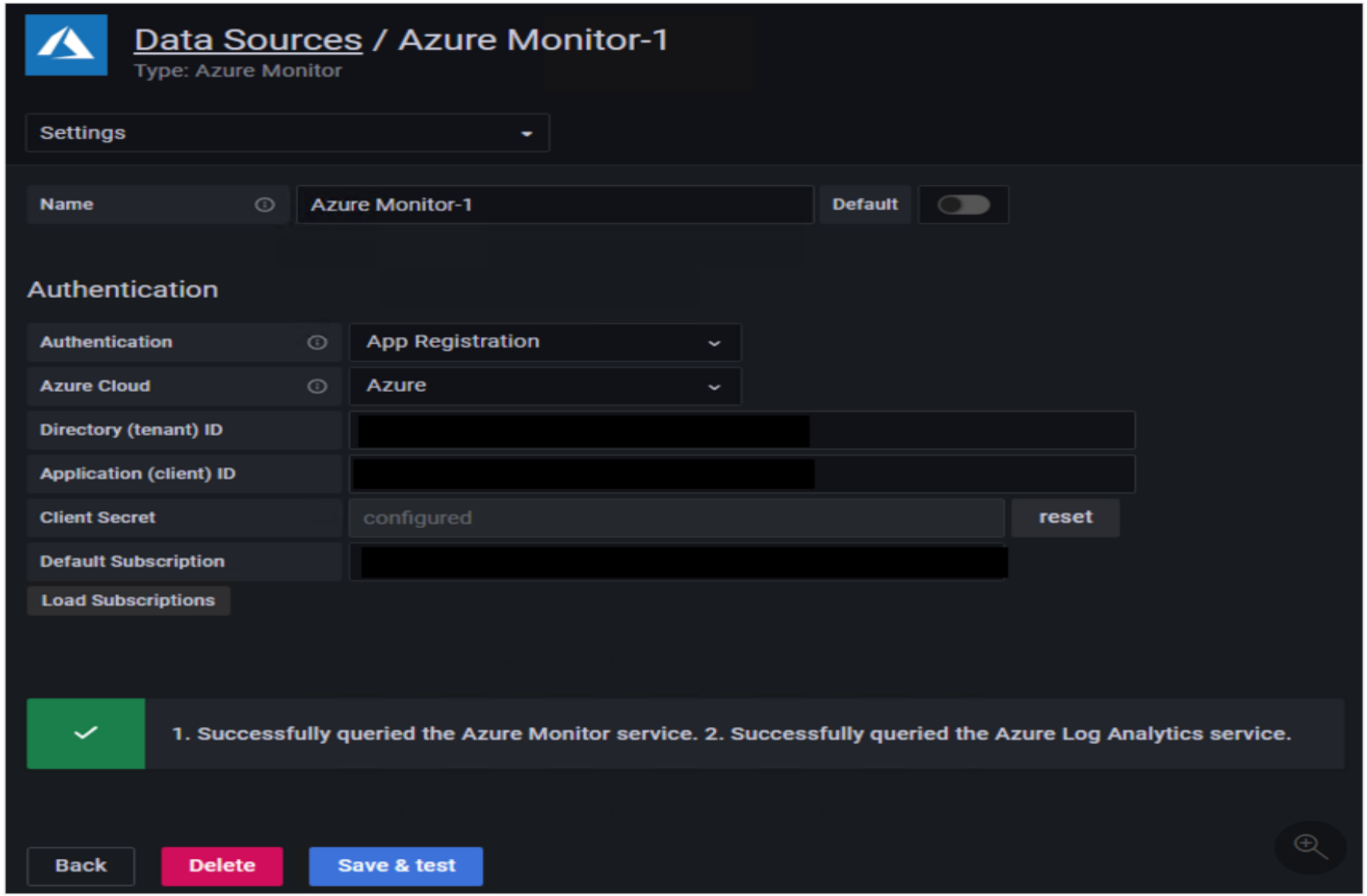
    - See Create a Microsoft Entra app and service principal in the portal to create a service principal. Copy and save your tenant ID (Directory ID), client ID (Application ID), and client secret (Application key value).
    - View Assign application to role to assign the Monitoring Reader role to the Microsoft Entra application on the subscription, resource group, or resource you want to monitor.

2. Provide the connection details you want to use:

    - When you configure the plug-in, you can indicate which Azure Cloud you want the plug-in to monitor: Public, Azure US Government, Azure Germany, or Microsoft Azure operated by 21Vianet.

      > **Note**
      >
      > Some data source fields are named differently than their correlated Azure settings:
      > - Tenant ID is the Azure Directory ID.
      > - Client ID is the Microsoft Entra Application ID.
      > - Client Secret is the Microsoft Entra Application key value.

3. Select **Save & test** and Grafana will test the credentials. You should see a message similar to the following one.

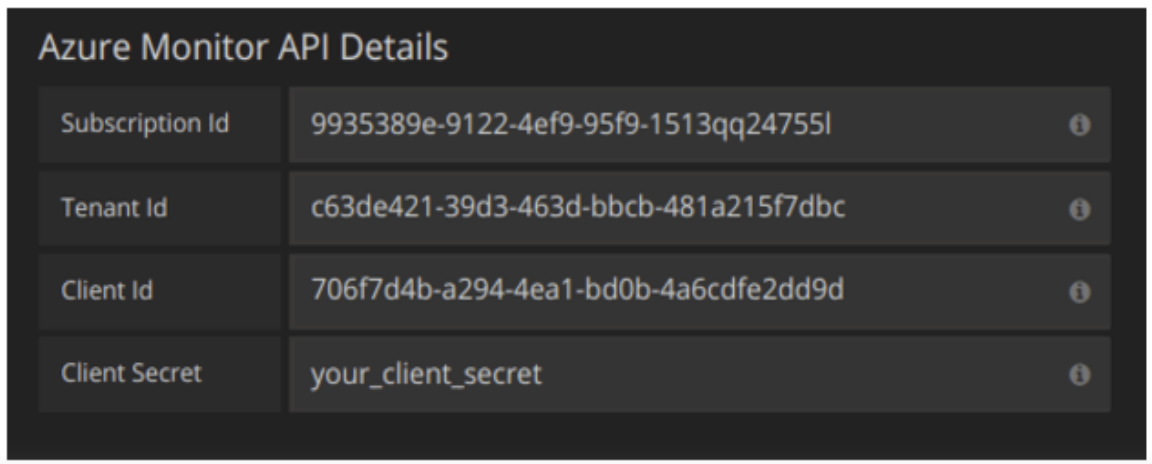


**Step 3:** In the name field, fill in a name for the data source. It can be anything. Some suggestions are Azure Monitor or App Insights.

**Step 4:** If you are using Azure Monitor, then you need 4 pieces of information from the Azure portal (see link above for detailed instructions):

- ***Tenant Id*** *(Azure Active Directory -> Properties -> Directory ID)*
- ***Subscription Id*** *(Subscriptions -> Choose subscription -> Overview -> Subscription ID)*
- ***Client Id*** *(Azure Active Directory -> App Registrations -> Choose your app -> Application ID)*
- ***Client Secret*** *( Azure Active Directory -> App Registrations -> Choose your app -> Keys)*

**Step 5:** Paste these four items into the fields in the Azure Monitor API Details section:



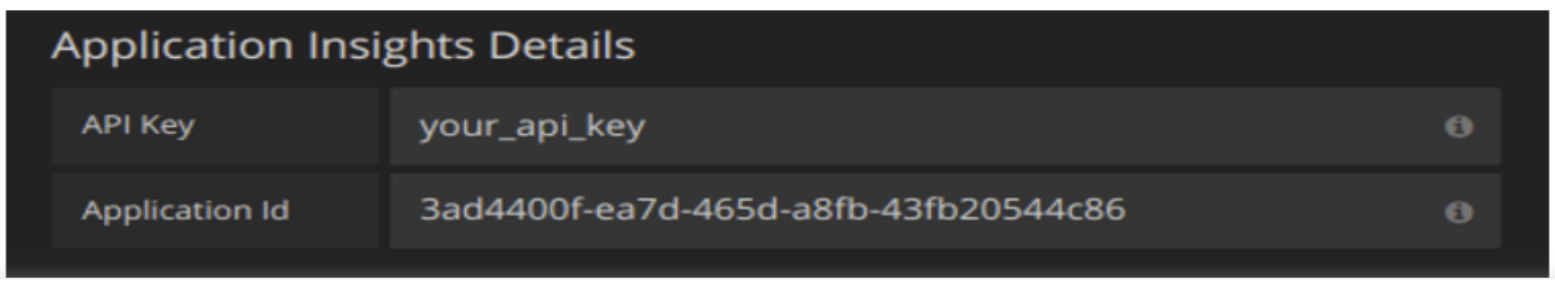
**Step 6:** If you are also using the Azure Log Analytics service, then you need to specify these two config values (or you can reuse the Client Id and Secret from the previous step).

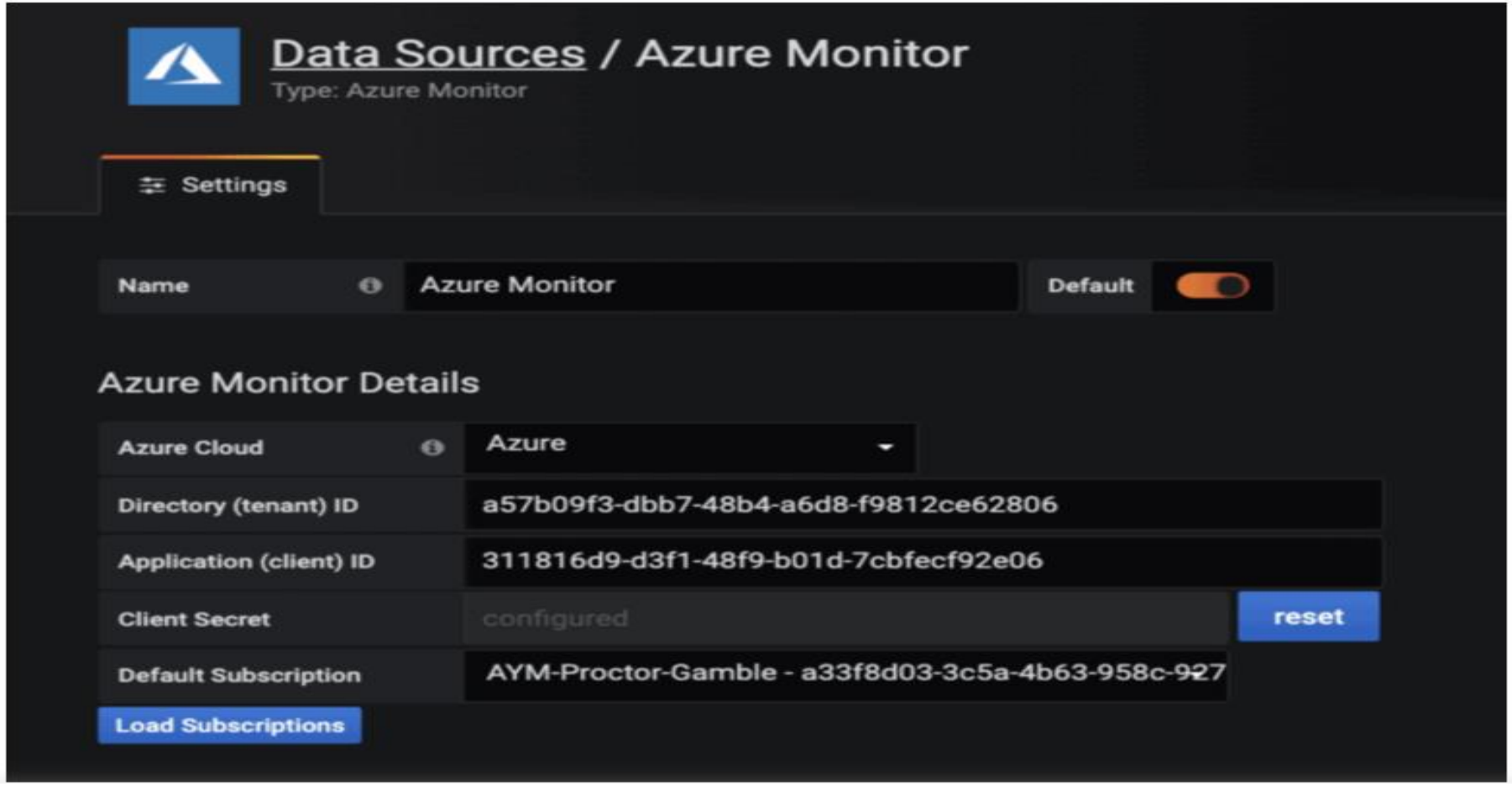
- *Client Id (Azure Active Directory -> App Registrations -> Choose your app -> Application ID)*
- *Client Secret ( Azure Active Directory -> App Registrations -> Choose your app -> Keys -> Create a key -> Use client secret)*

**Step 7:** If you are are using Application Insights, then you need two pieces of information from the Azure Portal (see link above for detailed instructions):

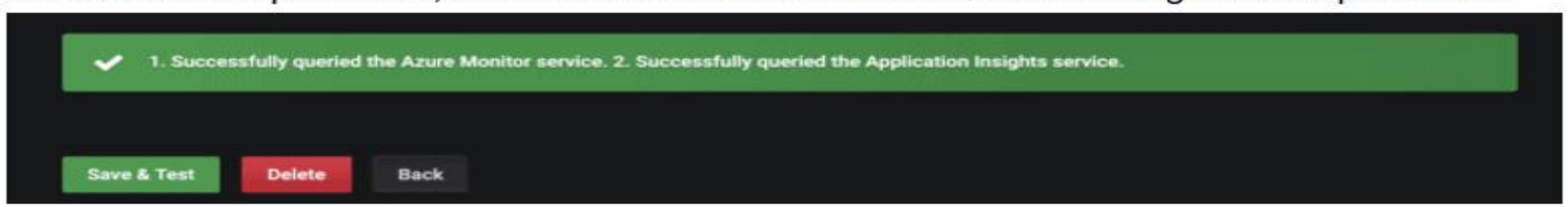
- *Application ID*
- *API Key*

**Step 8:** Paste these two items into the appropriate fields in the Application Insights API Details section:



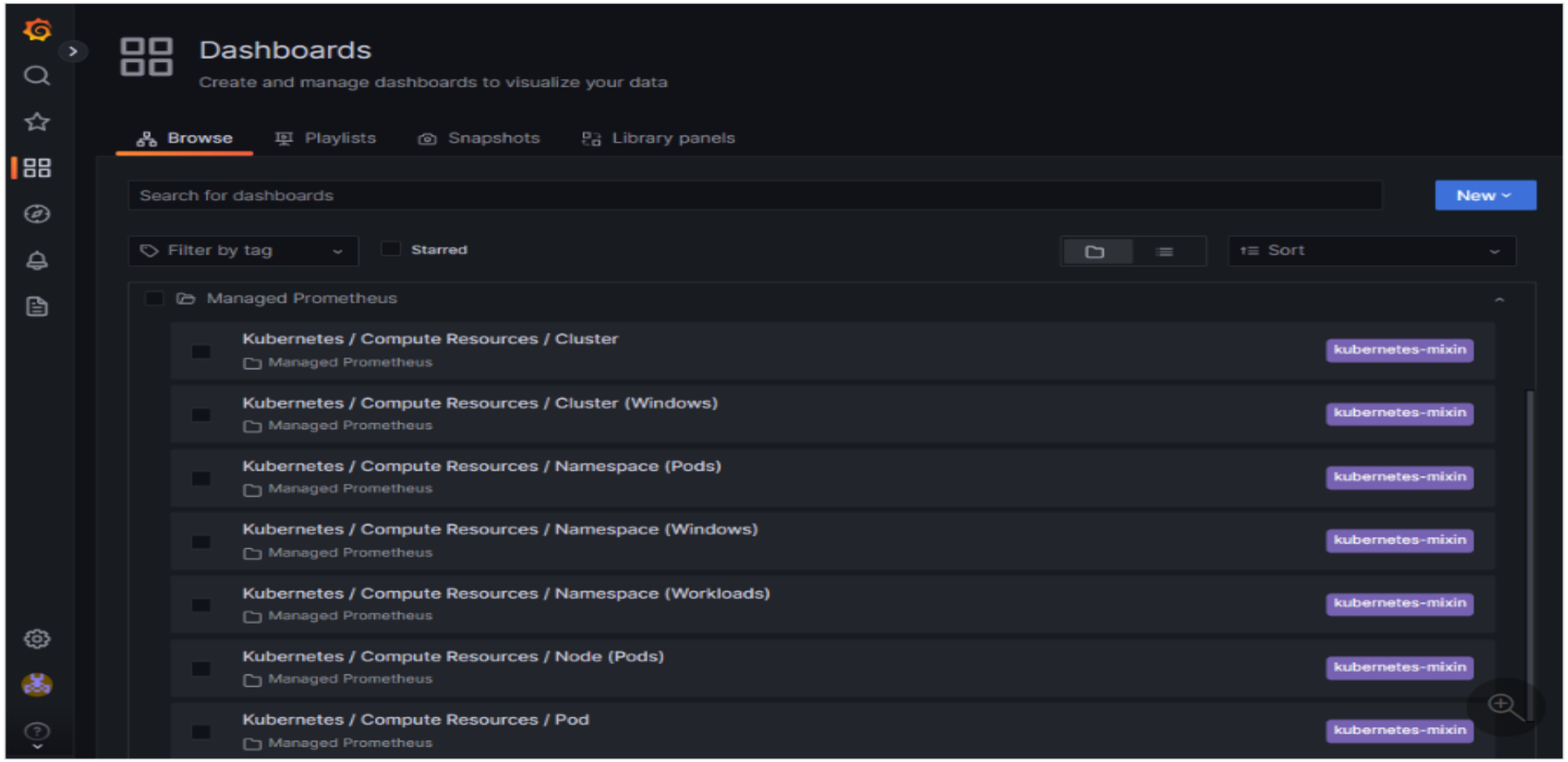



Once after all the details are passed on, we need to click on Save & Test and we must get this output below :



# Use out-of-the-box dashboards

Azure Monitor contains out-of-the-box dashboards to use with Azure Managed Grafana and the Azure Monitor plugin.



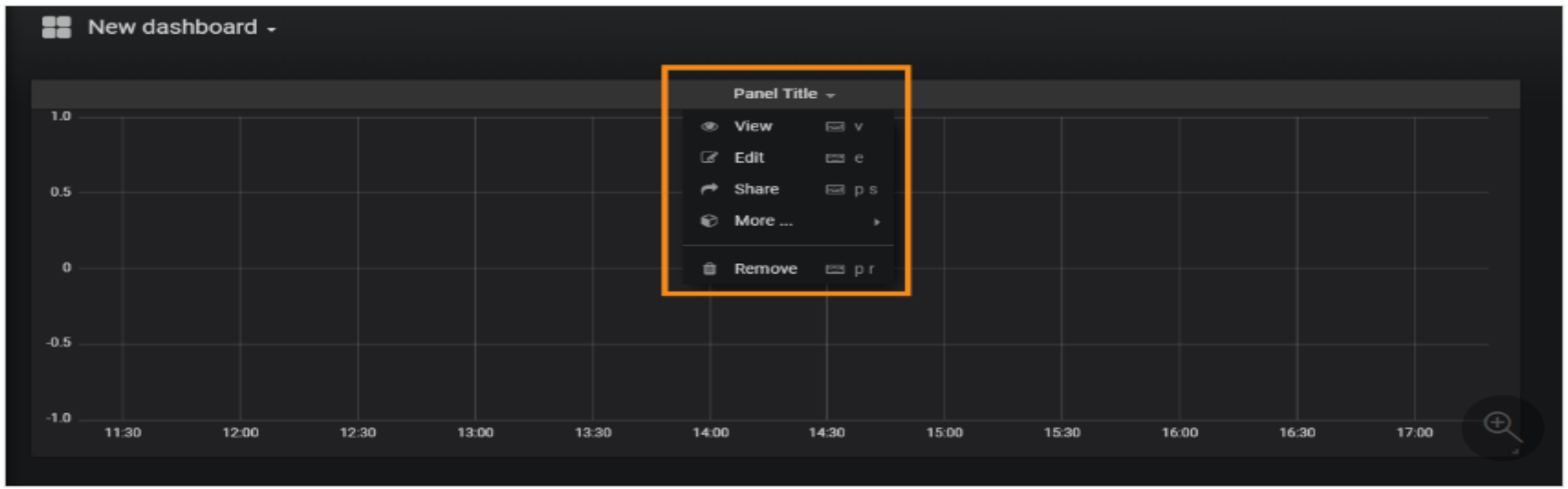

Azure Monitor also supports out-of-the-box dashboards for seamless integration with Azure Monitor managed service for Prometheus. These dashboards are automatically deployed to Azure Managed Grafana when linked to Azure Monitor managed service for Prometheus.

# Build a Grafana dashboard

1. Go to the Grafana home page and select **New Dashboard**.

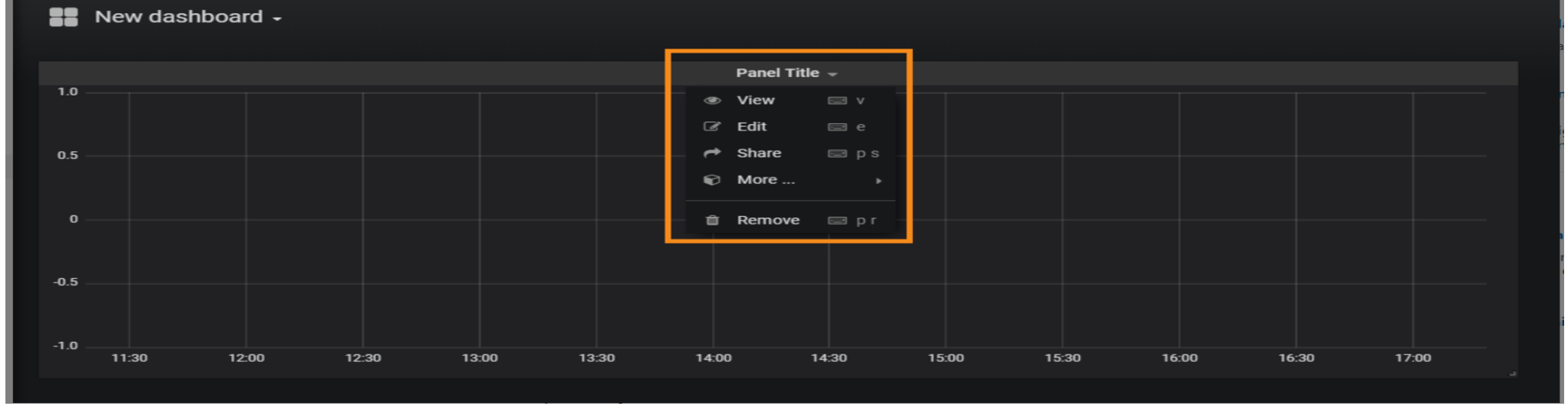
2. In the new dashboard, select **Graph**. You can try other charting options, but this article uses **Graph** as an example.

3. A blank graph shows up on your dashboard. Select the panel title and select **Edit** to enter the details of the data you want to plot in this graph chart.



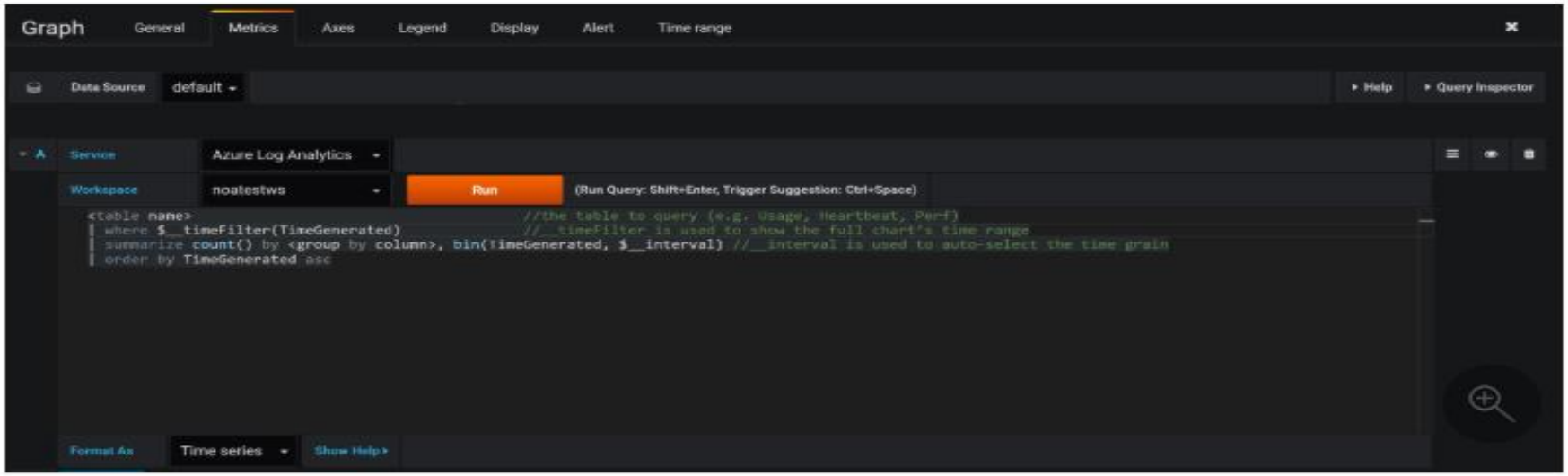

4. Select the Azure Monitor data source you've configured.

   - Visualizing Azure Monitor metrics: Select **Azure Monitor** in the service dropdown list. A list of selectors shows up where you can select the resources and metric to monitor in this chart. To collect metrics from a VM, use the namespace `Microsoft.Compute/VirtualMachines`. After you've selected VMs and metrics, you can start viewing their data in the dashboard.

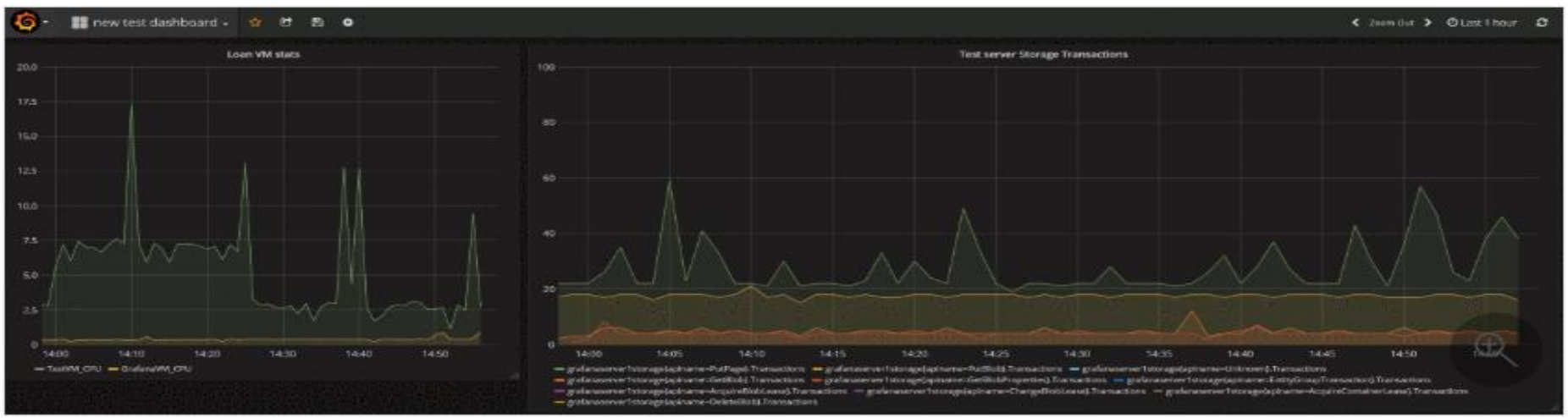



- Visualizing Azure Monitor log data: Select **Azure Log Analytics** in the service dropdown list. Select the workspace you want to query and set the query text. You can copy here any log query you already have or create a new one. As you enter your query, IntelliSense suggests autocomplete options. Select the visualization type, **Time series** > **Table**, and run the query.

> **Note**
>
> The default query provided with the plug-in uses two macros: `$__timeFilter()` and `$__interval`. These macros allow Grafana to dynamically calculate the time range and time grain, when you zoom in on part of a chart. You can remove these macros and use a standard time filter, such as `TimeGenerated > ago(1h)`, but that means the graph wouldn't support the zoom-in feature.

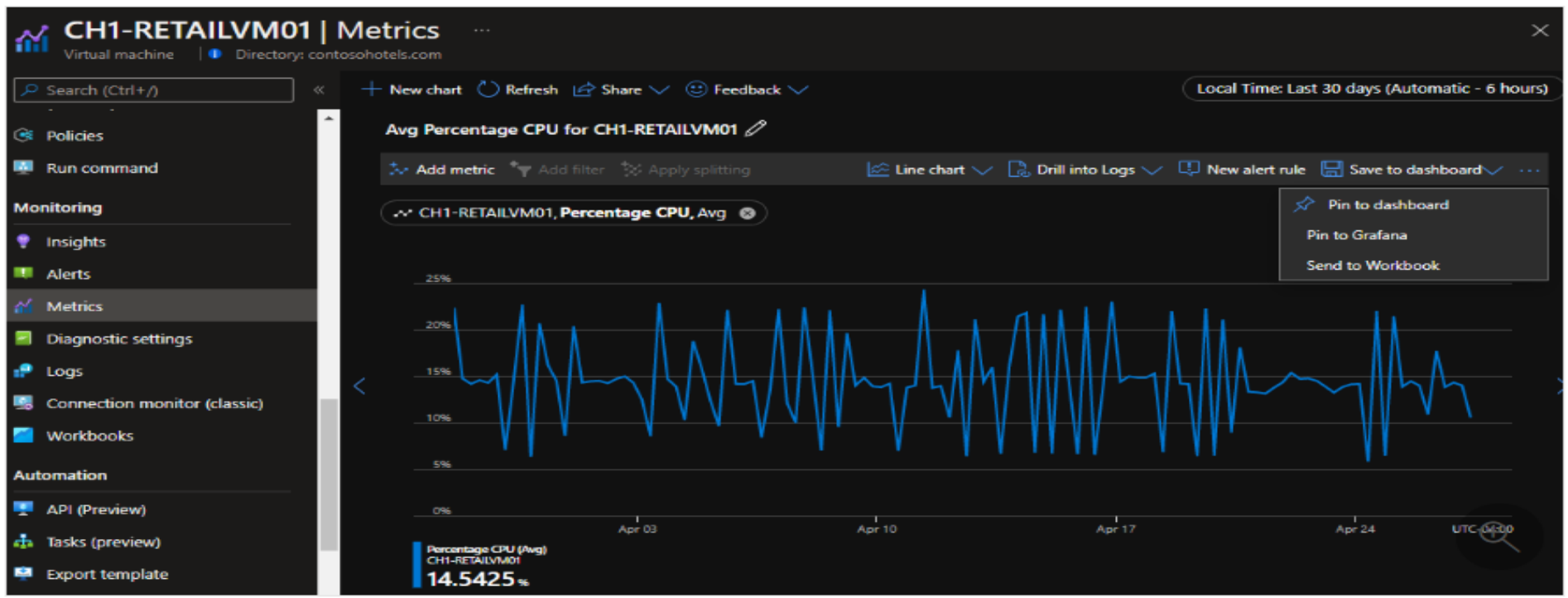



5. The following dashboard has two charts. The one on the left shows the CPU percentage of two VMs. The chart on the right shows the transactions in an Azure Storage account broken down by the Transaction API type.



# Pin charts from the Azure portal to Azure Managed Grafana

In addition to building your panels in Grafana, you can also quickly pin Azure Monitor visualizations from the Azure portal to new or existing Grafana dashboards by adding panels to your Grafana dashboard directly from Azure Monitor. Go to **Metrics** for your resource. Create a chart and select **Save to dashboard**, followed by **Pin to Grafana**. Choose the workspace and dashboard and select **Pin** to complete the operation.



# Advanced Grafana features

Grafana has advanced features.

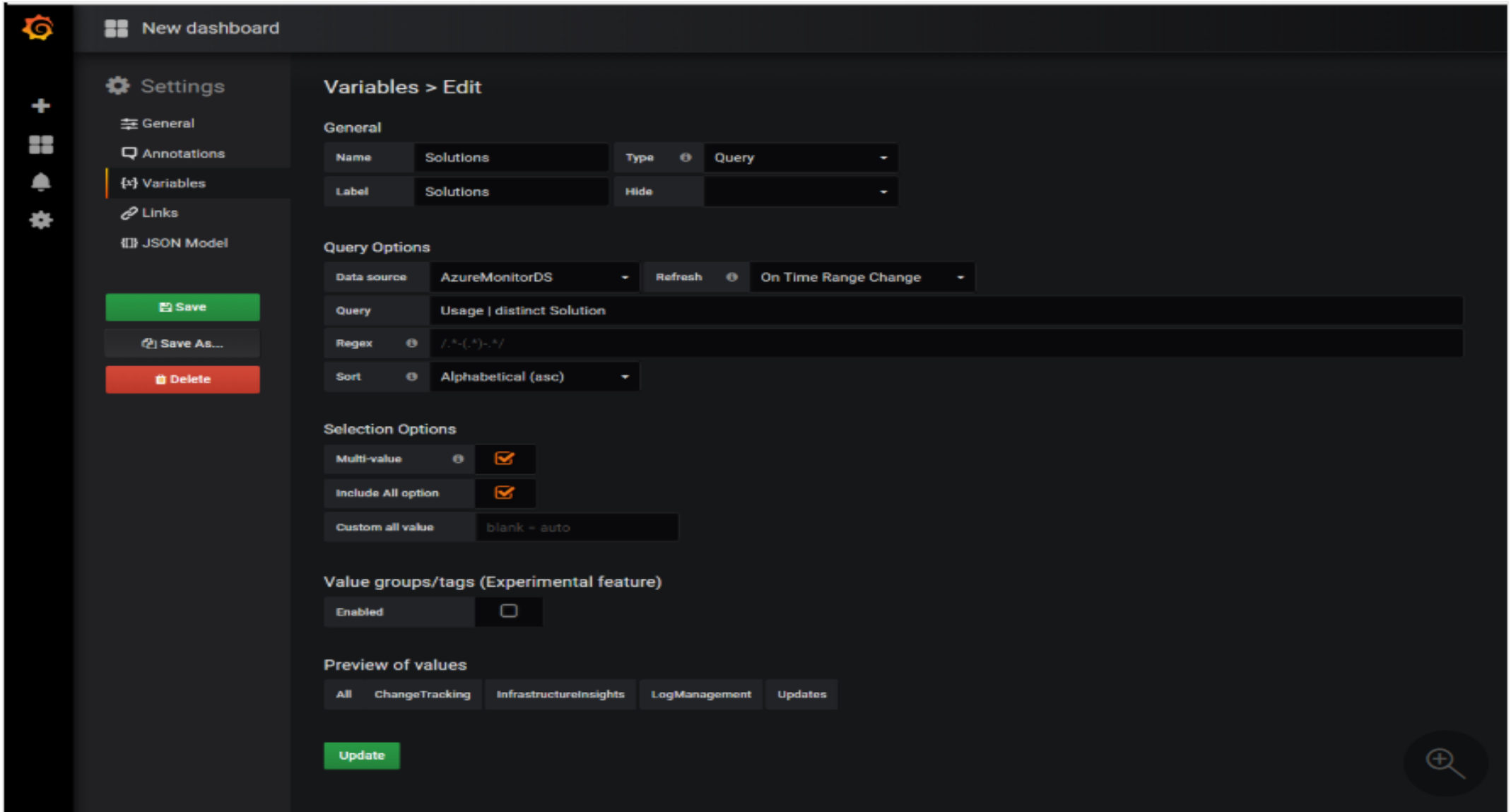
## Variables

Some query values can be selected through UI dropdowns and updated in the query. Consider the following query as an example:

Copy

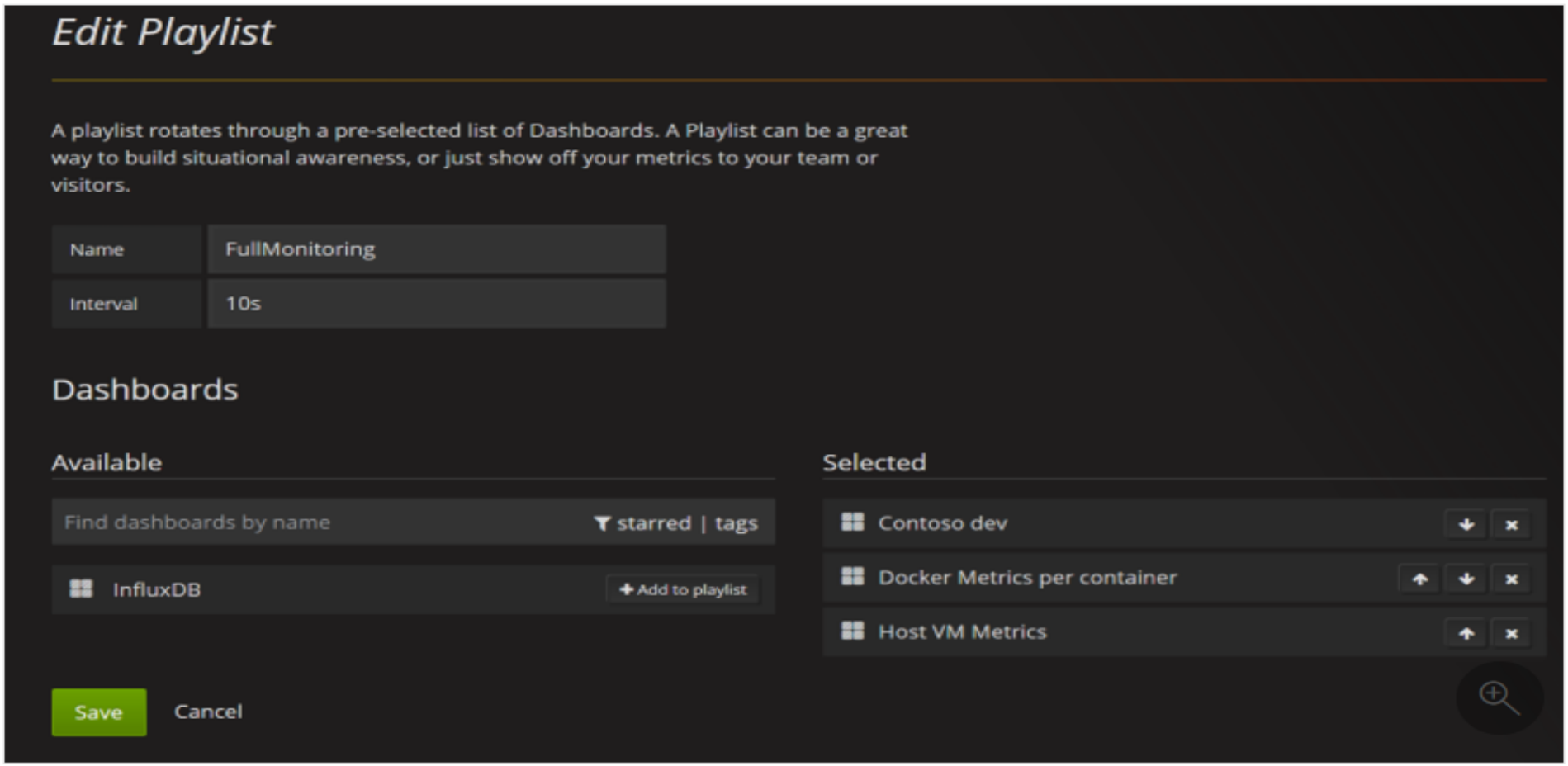
```
Usage
| where $__timeFilter(TimeGenerated)
| summarize total_KBytes=sum(Quantity)*1024 by bin(TimeGenerated, $__interval)
| sort by TimeGenerated
```

You can configure a variable that will list all available **Solution** values and then update your query to use it. To create a new variable, select the dashboard's **Settings** button in the top right area, select **Variables**, and then select **New**. On the variable page, define the data source and query to run to get the list of values.



# Create dashboard playlists

One of the many useful features of Grafana is the dashboard playlist. You can create multiple dashboards and add them to a playlist configuring an interval for each dashboard to show. Select **Play** to see the dashboards cycle through. You might want to display them on a large wall monitor to provide a status board for your group.



# Optional: Monitor other datasources in the same Grafana dashboards

There are many data source plug-ins that you can use to bring these metrics together in a dashboard.

Here are good reference articles on how to use Telegraf, InfluxDB, Azure Monitor managed service for Prometheus, and Docker:

- How to configure data sources for Azure Managed Grafana
- Use Azure Monitor managed service for Prometheus as data source for Grafana using managed system identity
- How to monitor system Metrics with the TICK Stack on Ubuntu 16.04
- A monitoring solution for Docker hosts, containers, and containerized services

Here's an image of a full Grafana dashboard that has metrics from Azure Monitor and Application Insights.

# Clean up resources

If you've set up a Grafana environment on Azure, you're charged when resources are running whether you're using them or not. To avoid incurring additional charges, clean up the resource group created in this article.

1. On the left menu in the Azure portal, select **Resource groups** > **Grafana**.
2. On your resource group page, select **Delete**, enter **Grafana** in the text box, and then select **Delete**.