Stackdriver Monitoring in the Cloud Console is now Generally Available and the default option. For a limited period of time, you al
he option to use the classic Stackdriver Monitoring console. For more information, see Monitoring in the Cloud Console
itoring/docs/monitoring_in_console).

This page helps you diagnose problems in the installation or running of the Monitoring agent.

f you see the following message on a Linux VM, you can safely ignore it.

essage is emitted by a legacy service that is gradually being replaced. This message is benign and will eventually disappear.

If you are having trouble installing or using the Monitoring agent, here are some things to check:

- If Linux installation commands result in errors, then make sure that you prefix the installation commands with `sudo`.
- Verify that the agent service is running on your VM instance:
    - For a Windows VM, use the following PowerShell command:

      Search for a service called **Stackdriver Monitoring**. If the agent is not running, you might need to restart it.
    - For a Linux VM, use the following command:

      If the agent is not running, you might need to restart it using the following command:

If the restart fails, and the log output shows "Disabled via metadata", you are likely running an image from Cloud Marketplace (/marketplace), where the Monitoring agent is disabled by default. This is controlled by the `google-monitoring-enable` instance metadata key (with the value `0`). To re-enable the agent, either remove that key or set the value to `1` (see Setting instance metadata (/compute/docs/storing-retrieving-metadata#updatinginstancemetadata)).

If the agent is not disabled via metadata, reinstall the agent. See the following section, Reinstalling the agent (#try-installing).

- See if the agent has written error messages to the logs.
  - On Windows, the Monitoring agent writes messages to the Windows Event log.
  - On Linux, the Monitoring agent is a `collectd` package and logs messages to `/var/log/syslog` or `/var/log/messages`. The log messages are prefixed by `collectd` or `stackdriver-agent`:
    - If you see HTTP 429 errors, you might have exceeded your Monitoring API quotas (/monitoring/quotas). You can see your available quota by selecting **APIs & services > Dashboard** in the Cloud Console. Choose the Monitoring API.
    - If you see proxy problems, check that you correctly configured you HTTP proxy. The instructions are part of Installing on Linux (/monitoring/agent/install-agent#agent-install-linux) or Installing on Windows (/monitoring/agent/install-agent#agent-install-windows).
    - If you see API access or authorization problems, or error messages such as "Unable to determine collectd endpoint", see the following section, Verifying project and credentials (#verify-project).
    - If you see "Unsupported collectd plugin/type combination" or "Unsupported collectd id" errors in the logs, you may be sending unsupported agent metrics. This can happen if you modified one of the agent Third-party Application configurations. To revert the changes, you can reinstall the configuration for the specific plug-in by following the instructions in the relevant documentation page (/monitoring/agent/plugins). If you want to use the agent to send that metric to Monitoring, look into converting them to custom metrics (/monitoring/agent/custom-metrics-agent).
- If the agent seems to be running normally, but you are not getting data or your alerting policies are not acting as you think they should, then you should check that the agent is sending data to the correct project. See the following section, Verifying project and credentials (#verify-project).

If the agent is reporting access or authorization errors, or if the agent seems to be running normally but there is no data or your alerting policies are not working as you expect, then you should check if your VM instance's

credentials are correct, including if they specify the correct project:

- To see if data is arriving in Monitoring, try to read some of the time series data. For instructions, see Verifying the agent-to-project connection (#verify-running). If you do see data, then the problem is not with the agent.
- If you are using a Google Compute Engine VM instance with standard (not private-key) credentials, then it is unlikely that data is going to the wrong project, but your credentials might still be deficient. For information about credentials, see Authorizing the agent (/logging/docs/agent/authorization#private_key_authorization). To verify your credentials, see Verifying Compute Engine credentials (#verify-creds).
- If you are using an Amazon EC2 VM instance, or if you are using private-key credentials on your Google Compute Engine instance, then the credentials could be invalid or they could be from the wrong project. For AWS accounts, the project used by the agent must be the AWS connector project (/monitoring/accounts/#account-project), typically named `"AWS Link..."`. For information about credentials, see Authorizing the agent (/logging/docs/agent/authorization#private_key_authorization). To verify your credentials, see Verifying private-key credentials (#verify-key).

If you still have not resolved your problem, see Reinstalling the agent (#try-installing).

To verify that the agent is sending metrics correctly, use the timeseries.list (/monitoring/api/ref_v3/rest/v3/projects.timeSeries/list) method of the Monitoring API to look for recent time series data from the VM instance. You can call the method using the APIs Explorer form at the bottom of the method's documentation page. If you do not see any data, it may be that the agent is sending data to the wrong project. To check that, see Verifying project and credentials (#verify-project).

Here are detailed instructions for using the timeseries.list (/monitoring/api/ref_v3/rest/v3/projects.timeSeries/list) method:

1. Determine the instance ID of the VM instance where you installed the agent:
   - **Compute Engine**: Go to Compute Engine's detail page for your instance. At the bottom of the page, click **Equivalent REST**. The ID is a 19-digit number.
   - **Amazon EC2**: The ID for each instance is shown in the list of instances. The ID looks like `i-1a2b3c4d`.
2. Go to the documentation page for the timeseries.list (/monitoring/api/ref_v3/rest/v3/projects.timeSeries/list) method:

   Open the timeseries.list page (https://cloud.google.com/monitoring/api/ref_v3/rest/v3/projects.timeSeries/list#try-it)
3. In the **Try it!** section, click the switch **Authorize requests using OAuth 2.0**. Accept the form without changes and click **Authorize**.

4. Fill out the APIs Explorer form:

    a. Set **name** to the project containing your VM instance, prefixed by `projects/`. For example, `projects/[YOUR_PROJECT_ID]`. For Amazon EC2 instances, you must use the AWS connector project (/monitoring/accounts/#account-project) for your Amazon account, which is typically has a name beginning with `"AWS Link"`.

    b. Set **filter** to the following line to choose an agent metric from your VM instance. Copy and paste it into the APIs Explorer, and then change the VM instance ID:

    c. Set the search time interval. You want approximately a five-minute interval:

    - Set **interval.endTime** to the current GMT time, which you can find at time.is/GMT (https://time.is/GMT). The time must be formatted like the following example. Do not enclose the time in quotation marks:

    - Set **interval.startTime** to approximately five minutes before the end time, using the same format.

    d. Leave all the other fields blank.

5. Click **Execute**.

You should see output like the following:

If the API call returns any time series data from your VM instance, as shown above, then your agent is working properly and you are finished.

If you do not see any time series data, check the following:

- If your API call results in an error message, this does *not* indicate an agent problem. Check that the APIs Explorer fields are filled properly. Check the spelling and format of the project ID, filter, and the two time stamps. "Not authorized" errors can mean you misspelled the project ID. "Not found" errors can indicate that you've omitted the required `projects/` prefix in the "name" field. Fix the problems and try the API call again.
- If the API call succeeds but you see only an empty response, `{ }`, then check that your filter and time interval are correct. Errors in formatting the timestamps can result in no data being returned. If everything seems correct but you are getting no data, then the agent is not sending metric data, or at least not to the project you are expecting it to. This might indicate a credentials problem; see Verifying private-key credentials (#verify-key).

Use the **Compute Engine > VM instances** page of the Cloud Console to verify that your Compute Engine VM instance has adequate credentials for the Monitoring agent. The credentials are typically added in the default service account of all new Compute Engine VM instances, but it is possible to overwrite those defaults when creating an instance.

Go to Compute Engine instances page (https://console.cloud.google.com/compute/instances)

1. If necessary, change the current GCP project to be the one associated with your Compute Engine VM instance. For example, if you are prompted to **Enable billing**, it means the current project does not have any Compute Engine VM instances in it.
2. In the **VM Instances** page, click the name of your VM instance. The detail page for your VM instance appears.

3. In the **VM instance details** page, look under the **Cloud API access scopes** heading:
    a. If you see "Allow full access to all Cloud APIs," then you have adequate credentials.
    b. If you see next to **Stackdriver Monitoring API** that you have **Write Only** or **Full** permission, then you have adequate credentials.
    c. Otherwise, your instance's default service account does not have the credentials needed by the agent. To use the agent on your instance, you must add private-key service account credentials. For instructions, see Adding credentials (/logging/docs/agent/authorization#private_key_authorization).

If you have the correct default credentials, skip ahead to Installing on Linux (/monitoring/agent/install-agent#agent-install-linux) or Installing on Microsoft Windows (/monitoring/agent/install-agent#agent-install-windows).

To verify that valid private-key credentials are installed on your VM instance, first verify that the credentials file exists in its expected location, and then verify that the information in the credentials file is valid. Previously-valid credentials can be revoked using the **IAM & Admin > Service accounts** section of the Cloud Console. If valid credentials aren't present, see Adding credentials (/logging/docs/agent/authorization#private_key_authorization) to replace the existing credentials or to add new ones.

n: Other services besides Stackdriver Monitoring might use private-key credentials on your instance. Replacing existing credenti
prevent other services from working.

To see if private-key service account credentials are on your instance, run the following Linux commands on your instance:

If either command displays a file like the one shown below, then your instance might have valid private-key credentials. If both commands display a file, then the file denoted by `GOOGLE_APPLICATION_CREDENTIALS` is used.

If there are no credential files present, then see Adding credentials (/logging/docs/agent/authorization#private_key_authorization).

In the credentials file, **project_id** is your GCP project, **client_email** identifies the service account in the project, and **private_key_id** identifies the private key in the service account. Match this information with what is shown in the **IAM & Admin > Service accounts** section of the Cloud Console.

The credentials file is *not* valid if any of the following are true:

- You are checking a Compute Engine instance, but the GCP project in the credentials file is not the project that contains your instance.
- You are checking an Amazon EC2 instance, but the GCP project in the credentials file is not the connector project (/monitoring/accounts/#account-project) (named `AWS Link...`) for your AWS account.
- The listed service account doesn't exist. It might have been deleted.
- The listed service account doesn't have the right roles enabled. It should have at least `roles/monitoring.metricWriter` (Monitoring Metric Writer) for the Monitoring agent and `roles/logging.logWriter` (Logs Writer) for the Logging agent.
- The private key doesn't exist. It might have been revoked.

If the service account is all right but the private key has been revoked, then you can create a new private key and copy it to your instance. Otherwise, you must create a new service account as described in the following section, Adding credentials (/logging/docs/agent/authorization#private_key_authorization).

If the credentials are not valid, take the following steps:

1. For each connected project containing instances that need to be authorized with a private key (all AWS connector projects and Compute Engine instances created without the monitoring scope), create a service account and generate a private key, if they do not already exist. Follow the steps below:
    a. To see the list of projects connected to your Workspace, go to **Monitoring**:

    Go to Monitoring (https://console.cloud.google.com/monitoring)

b. If **Settings** appears in the navigation pane, select **Settings** and then select the **Summary** tab. Otherwise, in the toolbar, click **Menu** ▾, select **Workspace settings**, and then select **Monitored accounts**:

- For AWS, use the link to navigate directly to the Cloud Console for the project in question.
- For GCP, identify the project containing the Compute Engine resources in question and navigate to the Cloud Console (https://console.cloud.google.com/).

c. From the Cloud Console, navigate to the IAM Service Accounts (https://console.cloud.google.com/iam-admin/serviceaccounts/project?project=) page.

d. Create a new service account and generate a new private key for it.

The simplest approach is to download a private key with the correct configuration. This key can be obtained by modifying the URL from the current page: append `&createStackdriverServiceAccount` to the end of the URL. For more information, see Creating a service account (/logging/docs/agent/authorization#create-service-account).

★ **Note:** The URL parameter above will automatically download a key; service accounts can have a maximum of 10 keys.

2. Replace the private key on the instances that correspond to the service account in question.

- On Linux, replace the private key located in `/etc/google/auth/application_default_credentials.json`
- On Windows, replace the private key located in `C:\ProgramData\Google\Auth\application_default_credentials.json` See Copying the private key to your instance (/logging/docs/agent/authorization#copy-private-key) for more details.

3. Restart the agent

- On Linux, run `sudo service stackdriver-agent restart`
- On Windows, go into the service management console and restart the `Stackdriver Monitoring` service.

If you have multiple projects that need new private keys, repeat this procedure for each of them.

To verify that the private key is correct, see Are the credentials present? (#present-creds). Specifically:

- Read the private key JSON file on the instance, for example (on Linux): `sudo cat /etc/google/auth/application_default_credentials.json`
- Ensure that the value of the `project-id` matches that of the monitored project for which you just generated credentials.

Installing the most recent version of the agent can solve many problems:

- If you are sure that the problem is not related to credentials, you can skip ahead to Installing on Linux (/monitoring/agent/install-agent#agent-install-linux) or Installing on Windows (/monitoring/agent/install-agent#agent-install-windows).
- For a full install of the agent and any needed credentials, see Installing the Monitoring Agent (/monitoring/agent/install-agent).

You can set up a script to check if the agent is running and then restart the agent in the event that it crashed.

For example, on Linux, you can create the following crontab entry to check the agent status every 5 minutes: