



Monitoring MySQL

MySQL PowerPack version 103

Table of Contents

Introduction	3
What Does the MySQL PowerPack Monitor?	3
Installing the MySQL PowerPack	4
Configuration and Discovery	5
Prerequisites for Monitoring MySQL	5
Creating a SOAP/XML Credential for MySQL	6
Creating a SOAP/XML Credential for MySQL in the SL1 Classic User Interface	8
Configuring the Credential to Read the MySQL Error Log	10
Configuring the Credential to Read the MySQL Error Log in the SL1 Classic User Interface	13
Creating a SOAP/XML Credential for an SSL Certificate	16
Creating a SOAP/XML Credential for an SSL Certificate in the SL1 Classic User Interface	18
Discovering MySQL Servers	20
Discovering MySQL Servers in the SL1 Classic User Interface	22
Verifying Discovery and Dynamic Application Alignment	24
Enabling the Slow Query Log in MySQL or MariaDB	26
Viewing MySQL Component Devices	27
MySQL Dashboards	30
Device Dashboards	30
MySQL: Instance	31

Chapter

1

Introduction

Overview

This manual describes how to monitor MySQL servers and instances in SL1 using the MySQL PowerPack.

The following sections provide an overview of MySQL and the MySQL PowerPack:

This chapter covers the following topics:

<i>What Does the MySQL PowerPack Monitor?</i>	3
<i>Installing the MySQL PowerPack</i>	4

NOTE: ScienceLogic provides this documentation for the convenience of ScienceLogic customers. Some of the configuration information contained herein pertains to third-party vendor software that is subject to change without notice to ScienceLogic. ScienceLogic makes every attempt to maintain accurate technical information and cannot be held responsible for defects or changes in third-party vendor software. There is no written or implied guarantee that information contained herein will work for all third-party variants. See the End User License Agreement (EULA) for more information.

What Does the MySQL PowerPack Monitor?

To monitor MySQL servers using SL1, you must install the MySQL PowerPack. This PowerPack enables you to discover, model, and collect data about MySQL servers and instances.

The MySQL PowerPack includes:

- Dynamic Applications to discover, model, and monitor performance metrics and collect configuration data for MySQL servers and instances

- Device Classes for MySQL Servers and MySQL instances
- A sample credential for discovering MySQL servers
- Event Policies and corresponding alerts that are triggered when MySQL servers and instances meet certain status criteria
- A Device Dashboard for viewing data about MySQL servers and instances

Installing the MySQL PowerPack

Before completing the steps in this manual, you must import and install the latest version of the MySQL PowerPack.

TIP: By default, installing a new version of a PowerPack overwrites all content from a previous version of that PowerPack that has already been installed on the target system. You can use the **Enable Selective PowerPack Field Protection** setting in the **Behavior Settings** page (System > Settings > Behavior) to prevent new PowerPacks from overwriting local changes for some commonly customized fields. (For more information, see the **System Administration** manual.)

IMPORTANT: The minimum required MySQL version is 5.6.0.

To download and install the PowerPack:

1. Search for and download the PowerPack from the **PowerPacks** page (Product Downloads > PowerPacks & SyncPacks) at the [ScienceLogic Support Site](#).
2. In SL1, go to the **PowerPacks** page (System > Manage > PowerPacks).
3. Click the **[Actions]** button and choose *Import PowerPack*. The **Import PowerPack** dialog box appears.
4. Click **[Browse]** and navigate to the PowerPack file from step 1.
5. Select the PowerPack file and click **[Import]**. The **PowerPack Installer** modal displays a list of the PowerPack contents.
6. Click **[Install]**. The PowerPack is added to the **PowerPacks** page.

NOTE: If you exit the **PowerPack Installer** modal without installing the imported PowerPack, the imported PowerPack will not appear in the **PowerPacks** page. However, the imported PowerPack will appear in the **Imported PowerPacks** modal. This page appears when you click the **[Actions]** menu and select *Install PowerPack*.

Chapter

2

Configuration and Discovery

Overview

The following sections describe how to configure and discover MySQL for monitoring by SL1 using the MySQL PowerPack:

This chapter covers the following topics:

<i>Prerequisites for Monitoring MySQL</i>	5
<i>Creating a SOAP/XML Credential for MySQL</i>	6
<i>Configuring the Credential to Read the MySQL Error Log</i>	10
<i>Creating a SOAP/XML Credential for an SSL Certificate</i>	16
<i>Discovering MySQL Servers</i>	20
<i>Verifying Discovery and Dynamic Application Alignment</i>	24
<i>Enabling the Slow Query Log in MySQL or MariaDB</i>	26
<i>Viewing MySQL Component Devices</i>	27

Prerequisites for Monitoring MySQL

To configure the SL1 system to monitor MySQL servers and instances using the MySQL PowerPack, you must first create a read-only MySQL user for each instance to be monitored. For discovery of multiple instances on the same IP address, ScienceLogic recommends creating the same user and password on each instance. The user must have the minimum following privileges:

Privilege	Definition	Level(s)
SELECT	Enables the use of SELECT.	Global, database, table, column.
EXECUTE	Enable the use of statements that execute stored routines (stored procedures and functions). This is necessary for queries on the system database.	

Creating a SOAP/XML Credential for MySQL

To configure SL1 to monitor MySQL, you must create a SOAP/XML credential. This credential allows the Dynamic Applications in the MySQL PowerPack to communicate with your MySQL server and instances.

The MySQLPowerPack includes an example SOAP/XML credential that you can use as a template for creating SOAP/XML credentials for MySQL.

NOTE: If you are using an SL1 system prior to version 11.1.0, the new user interface does not include the **Duplicate** option for sample credential(s). ScienceLogic recommends that you use [the classic user interface and the Save As button](#) to create new credentials from sample credentials. This will prevent you from overwriting the sample credential(s).

To configure a SOAP/XML credential to access your MySQL server:

1. Go to the **Credentials** page (Manage > Credentials).
2. Locate the **MySQL Example Credential** click its **[Actions]** icon (☰) and select **Duplicate**. A copy of the credential appears called **MySQL Example Credential copy**.

- Click the **[Actions]** icon (☰) for the **MySQL Example Credential copy** credential and select **Edit**. The **Edit Credential** modal page appears:

- **Name.** Type a new name for the MySQL credential.
- **All Organizations.** Toggle on (blue) to align the credential to all organizations, or toggle off (gray) and then select one or more specific organizations from the **Select the organizations the credential belongs to** drop-down field to align the credential with those specific organizations.
- **Timeout (ms).** It is recommended to enter a timeout of at least 2000 milliseconds.
- **URL.** Type "https://%D".
- **HTTP Auth User.** Type the username for your MySQL server.
- **HTTP Auth Password.** Type the password for your MySQL server.

NOTE: To discover multiple MySQL instances on the same IP address, ScienceLogic recommends creating the same user and password on each instance, so the user will need to create only one credential.

- **HTTP Headers.** The following headers are in the example credential and are required:
 - Service:MySQL
 - Range:<port_begin>-<port_end>. Specify the range of ports on which your MySQL server is running. For example, "Range:3305-3310".
 - Linux:<ssh_cred_id>. If you have [configured credentials to read the error log](#), enter the credential ID for the SSH credential for a Linux server. For Windows servers, update the field to "Windows:<powershell_cred_id>".

NOTE: To discover a MySQL instance with one port, users must specify the "Range" header as "Range:<port_begin>-<port_end>". Meaning, "port_begin" and "port_end" must be the same port. For example, "Range:7706-7706".


4. For all other fields, use the default values.
5. Click the **[Save & Close]** button.

Creating a SOAP/XML Credential for MySQL in the SL1 Classic User Interface

To configure SL1 to monitor MySQL, you must create a SOAP/XML credential. This credential allows the Dynamic Applications in the *MySQL PowerPack* to communicate with your MySQL server and instances.

The *MySQLPowerPack* includes an example SOAP/XML credential that you can use as a template for creating SOAP/XML credentials for MySQL.

To configure a SOAP/XML credential to access your MySQL server:

1. Go to the **Credential Management** page (System > Manage > Credentials).
2. Locate the **MySQL Example Credential** and click its wrench icon (). The **Edit SOAP/XML Credential** modal page appears.

3. Enter values in the following fields:

The screenshot shows the 'Credential Editor [97]' window. The title bar includes 'New' and 'Reset' buttons. The main area is divided into several sections:

- Basic Settings:** Profile Name (MySQL Example Credential), Content Encoding (text/xml), Method (POST), HTTP Version (HTTP/1.1), URL (https://%D), HTTP Auth User (<user>), HTTP Auth Password (masked), Timeout (seconds) (2).
- Proxy Settings:** Hostname/IP, Port (0), User.
- CURL Options:** A list of options including CAINFO, CAPATH, CLOSEPOLICY, CONNECTTIMEOUT, COOKIE, COOKIEFILE, COOKIEJAR, COOKIELIST, CRLF, CUSTOMREQUEST, and DNSCACHETIMEOUT.
- Soap Options:** Embedded Password [%P], Embed Value [%1] through [%4].
- HTTP Headers:** A list of headers: Service:MySQL, Range:<port_begin>-<port_end>, and Linux:<ssh_cred_id>.

At the bottom, there are 'Save' and 'Save As' buttons.

Basic Settings

- **Profile Name.** Type a new name for the MySQL credential.
- **URL.** Type "%D".
- **HTTP Auth User.** Type the username for your MySQL server.
- **HTTP Auth Password.** Type the password for your MySQL server.
- **Timeout.** It is recommended to enter a timeout of at least 2 seconds.

NOTE: To discover multiple MySQL instances on the same IP address, ScienceLogic recommends creating the same user and password on each instance, so the user will need to create only one credential.

HTTP Headers

- **HTTP Headers.** The following headers are in the example credential and are required:
 - Service:MySQL
 - Range:<port_begin>-<port_end>. Specify the range of ports on which your MySQL server is running. For example, "Range:3305-3310".

- Linux: <ssh_cred_id>. If you have [configured credentials to read the error log](#), enter the credential ID for the SSH credential for a Linux server. For Windows servers, update the field to "Windows: <powershell_cred_id>".

NOTE: To discover a MySQL instance with one port, users must specify the "Range" header as "Range: <port_begin>-<port_end>". Meaning, "port_begin" and "port_end" must be the same port. For example, "Range:7706-7706".

4. For all other fields, use the default values.
5. Click the **[Save As]** button.

Configuring the Credential to Read the MySQL Error Log

In addition to the [SOAP/XML credential created to monitor MySQL](#), another credential must be created to read the MySQL Error Log. The credentials are configured differently for Linux and Windows servers.

NOTE: If you are using an SL1 system prior to version 11.1.0, the new user interface does not include the **Duplicate** option for sample credential(s). ScienceLogic recommends that you use [the classic user interface and the Save As button](#) to create new credentials from sample credentials. This will prevent you from overwriting the sample credential(s).

For Linux servers, you must create an SSH/Key credential. To create the credential:

1. Go to the **Credentials** page (Manage > Credentials).
2. Locate the **MySQL SSH Example** credential, click its **[Actions]** icon (☰) and select **Duplicate**. A copy of the credential appears.
3. Click the **[Actions]** icon (☰) for the credential copy and select **Edit**. The **Edit Credential** modal page appears:

- **Name**. Type a new name for the credential.
 - **All Organizations**. Toggle on (blue) to align the credential to all organizations, or toggle off (gray) and then select one or more specific organizations from the *What organization manages this service?* drop-down field to align the credential with those specific organizations.
 - **Hostname/IP**. Type "%D".
 - **Username**. Type the username for your Linux server.
 - **Password**. Type the password for your Linux server.
4. For all other fields, use the default values.
 5. Click the **[Save & Close]** button.

NOTE: The private key can have a maximum of 64 characters per line. Therefore, you cannot use keys in the OpenSSH format, because that format uses 70 characters per line. When you attempt to save the credential, SL1 will validate that the private key entered is in the correct format. You will be able to save the credential only if the private key is correctly formatted.

For Windows servers, you must create a PowerShell credential. To create the credential:

1. Go to the **Credentials** page (Manage > Credentials).
2. Locate the **MySQL PowerShell Example** credential, click its **[Actions]** icon (⋮) and select **Duplicate**. A copy of the credential appears.

3. Click the **[Actions]** icon (☰) for the credential copy and select **Edit**. The **Edit Credential** modal page appears:

- **Name**. Type a new name for the credential.
 - **All Organizations**. Toggle on (blue) to align the credential to all organizations, or toggle off (gray) and then select one or more specific organizations from the **What organization manages this service?** drop-down field to align the credential with those specific organizations.
 - **Hostname/IP**. Type "%D".
 - **Username**. Type your username for the Windows server.
 - **Password**. Type your password for the Windows server.
4. For all other fields, use the default values.
 5. Click the **[Save & Close]** button.

To configure the existing SOAP credential:

1. Go to the **Credentials** page (Manage > Credentials).
2. Locate the **MySQL credential you created**, click its **[Actions]** icon (☰) and select **Edit/Test**.
3. In the **HTTP Headers** section, enter the credential ID for the SSH credential for a Linux server. For Windows servers, update the field to "Windows:<powershell_cred_id>".

Edit Credential

HTTP Auth User
<user>

HTTP Auth Password

Proxy Hostname/IP
optional

Proxy Port
0

Proxy User
optional

Proxy Password

Embedded Password [%P]

Embed Value [%1]

Embed Value [%2]

Embed Value [%3]

Embed Value [%4]

HTTP Headers Add Header

Service:MySQL X

Range:<port_begin>-<port_end> X

Linux:<ssh_cred_id> X

CURL Options Add CURL Option

Close

4. Click the **[Save & Close]** button.

Configuring the Credential to Read the MySQL Error Log in the SL1 Classic User Interface

In addition to the [SOAP/XML credential created to monitor MySQL](#), another credential must be created to read the MySQL Error Log. The credentials are configured differently for Linux and Windows servers.

For Linux servers, you must create an SSH/Key credential. To create the credential:

1. Go to the **Credential Management** page (System > Manage > Credentials).
2. Click the **[Actions]** button and select the option *Create SSH/Key Credential* for Linux servers or *Create PowerShell Credential* for Windows Servers.

3. Enter values in the following fields:

For Linux Servers:

The screenshot shows a window titled "Credential Editor [107]" with a sub-header "Edit SSH/Key Credential #107". The window contains a "Basic Settings" section with the following fields and values:

- Credential Name:** MySQL SSH EM7
- Hostname/IP:** %D
- Port:** 22
- Timeout(ms):** 10
- Username:** em7admin
- Password:** (masked with dots)
- Private Key (PEM Format):** (empty text area)

Buttons for "New", "Reset", "Save", and "Save As" are located at the bottom of the window.

- **Credential Name.** Type a new name for the credential.
- **Hostname/IP.** Type "%D".
- **Username.** Type the username for your Linux server.
- **Password.** Type the password for your Linux server.

NOTE: The private key can have a maximum of 64 characters per line. Therefore, you cannot use keys in the OpenSSH format, because that format uses 70 characters per line. When you attempt to save the credential, SL1 will validate that the private key entered is in the correct format. You will be able to save the credential only if the private key is correctly formatted.

4. For all other fields, use the default values.

5. Click the **[Save]** button.

For Windows servers, you must create a PowerShell credential. To create the credential:

1. Go to the **Credential Management** page (System > Manage > Credentials).
2. Click the **[Actions]** button and select *Create PowerShell Credential*.

3. Enter values in the following fields:

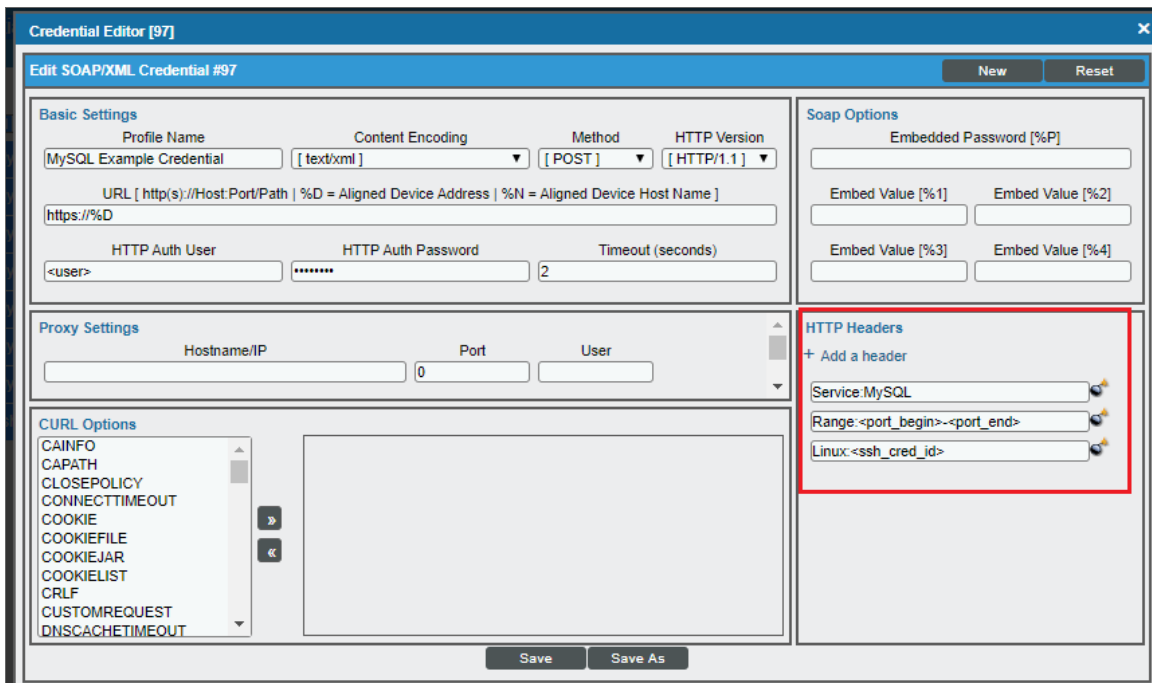
The screenshot shows the 'Credential Editor [100]' window. The title bar includes 'Credential Editor [100]' and a close button. Below the title bar is a sub-header 'Edit PowerShell Credential #100' with 'New' and 'Reset' buttons. The main area is divided into two sections: 'Basic Settings' and 'Active Directory Settings'.
Basic Settings:
- Profile Name: Windows-Powershell
- Account Type: [Local] (dropdown)
- Hostname/IP: %D
- Timeout(ms): 180000
- Username: Administrator
- Password: [masked with dots]
- Encrypted: [no] (dropdown)
- Port: 5985
- PowerShell Proxy Hostname/IP: [empty]
Active Directory Settings:
- Active Directory Hostname/IP: [empty]
- Domain: [empty]
At the bottom are 'Save' and 'Save As' buttons.

- **Profile Name.** Type a new name for the credential.
- **Hostname/IP.** Type "%D".
- **Username.** Type your username for the Windows server.
- **Password.** Type your password for the Windows server.

4. For all other fields, use the default values.
5. Click the **[Save]** button.

To configure the existing SOAP credential:

1. Go to the **Credential Management** page (System > Manage > Credentials).
2. Locate the **MySQL credential you created** and click its wrench icon (🔧). The **Edit SOAP/XML Credential** modal page appears.
3. In the **HTTP Headers** pane, enter the credential ID for the SSH credential for a Linux server. For Windows servers, update the field to "Windows:<powershell_cred_id>".



4. Click the **[Save]** button.

Creating a SOAP/XML Credential for an SSL Certificate

In addition to the [SOAP/XML credential created to monitor MySQL](#), another credential must be created to support loading your SSL certificate on a database connection.

NOTE: If you are using an SL1 system prior to version 11.1.0, the new user interface does not include the **Duplicate** option for sample credential(s). ScienceLogic recommends that you use [the classic user interface and the Save As button](#) to create new credentials from sample credentials. This will prevent you from overwriting the sample credential(s).

To create the credential:

1. Go to the **Credentials** page (Manage > Credentials).
2. Locate the **MySQL SSH Example Credential SSL** credential, click its **[Actions]** icon (☰) and select **Duplicate**. A copy of the credential appears.
3. Click the **[Actions]** icon (☰) for the credential copy and select **Edit**. The **Edit Credential** modal page appears:

- **Name.** Type a new name for the credential.
- **All Organizations.** Toggle on (blue) to align the credential to all organizations, or toggle off (gray) and then select one or more specific organizations from the **What organization manages this service?** drop-down field to align the credential with those specific organizations.
- **Timeout (ms).** It is recommended to enter a timeout of at least 2000 milliseconds.
- **URL.** Type "https://%D".
- **HTTP Auth User.** Type the username for your server.
- **HTTP Auth Password.** Type the password for your server.

NOTE: To discover multiple MySQL instances on the same IP address, ScienceLogic recommends creating the same user and password on each instance, so the user will need to create only one credential.

- **HTTP Headers.** The following headers are in the example credential and are required:
 - Service:MySQL
 - Range:<port_begin>-<port_end>. Specify the range of ports on which your MySQL server is running. For example, "Range:3305-3310".
 - Linux:<ssh_cred_id>. If you have [configured credentials to read the error log](#), enter the credential ID for the SSH credential for a Linux server. For Windows servers, update the field to "Windows:<powershell_cred_id>".

NOTE: To discover a MySQL instance with one port, users must specify the "Range" header as "Range:<port_begin>-<port_end>". Meaning, "port_begin" and "port_end" must be the same port. For example, "Range:7706-7706".

- **CURL Options.** Edit the following fields in this section:
 - **CAPATH.** Type the CA path for your SSL certificate.
 - **SSLKEY.** Type the key path for your SSL certificate.
 - **SSLPEERCERT.** Type the certificate path for your SSL certificate.

The screenshot shows the 'Edit Credential' modal window. It has a title bar with a close button (X). The main content area is divided into several sections. At the top is 'Embedded Password [69]' with a masked input field. Below that are four 'Embed Value' fields labeled [%1] through [%4]. The 'HTTP Headers' section has an 'Add Header' button and three header entries: 'Service:MySQL', 'Range:<port_begin>-<port_end>', and 'Linux<ssh_cred_id>', each with a delete (X) icon. The 'CURL Options' section is highlighted with a red box and contains three entries: 'CAPATH' with '<ca_path>', 'SSLKEY' with '<key_path>', and 'SSLPEERCERT' with '<certificate_path>', each with a delete (X) icon. To the right of this section is an 'Add CURL Option' dropdown. At the bottom right of the modal are 'Save & Test' and 'Save & Close' buttons.

4. For all other fields, use the default values.
5. Click the [Save & Close] button.

Creating a SOAP/XML Credential for an SSL Certificate in the SL1 Classic User Interface

In addition to the [SOAP/XML credential created to monitor MySQL](#), another credential must be created to support loading your SSL certificate on a database connection.

To configure the credential:

1. Go to the **Credential Management** page (System > Manage > Credentials).
2. Locate the **MySQL SSH Example Credential SSL** and click its wrench icon (🔧). The **Edit SOAP/XML Credential** modal page appears.

3. Enter values in the following fields:

The screenshot shows the 'Credential Editor [82]' window. The title bar indicates it is editing 'SOAP/XML Credential #82'. The interface is organized into several panels:

- Basic Settings:** Profile Name: MySQL Example Credential SSL; Content Encoding: [text/xml]; Method: [POST]; HTTP Version: [HTTP/1.1]; URL: [https://%D]; HTTP Auth User: <user>; HTTP Auth Password: *****; Timeout (seconds): 2.
- Proxy Settings:** Hostname/IP, Port: 0, User.
- CURL Options:** A list of options on the left (CAINFO, CLOSEPOLICY, CONNECTTIMEOUT, COOKIE, COOKIEFILE, COOKIEJAR, COOKIELIST, CRLF, CUSTOMREQUEST, DNSCACHETIMEOUT, DNSUSEGLOBALCACHE) and fields for CAPATH (<ca_path>), SSLKEY (<key_path>), and SSLPEERCERT (<certificate_path>).
- Soap Options:** Embedded Password [%P] (*****); Embed Value [%1] through [%4] (empty).
- HTTP Headers:** + Add a header; Service:MySQL; Range:<port_begin>-<port_end>; Linux:<ssh_cred_id>.

Buttons for 'Save' and 'Save As' are located at the bottom center.

Basic Settings

- **Profile Name.** Type a new name for the MySQL credential.
- **URL.** Type "%D".
- **HTTP Auth User.** Type the username for your MySQL server.
- **HTTP Auth Password.** Type the password for your MySQL server.
- **Timeout.** It is recommended to enter a timeout of at least 2 seconds.

NOTE: To discover multiple MySQL instances on the same IP address, ScienceLogic recommends creating the same user and password on each instance, so the user will need to create only one credential.

HTTP Headers

- **HTTP Headers.** The following headers are in the example credential and are required:
 - Service:MySQL
 - Range:<port_begin>-<port_end>. Specify the range of ports on which your MySQL server is running. For example, "Range:3305-3310".

- Linux: <ssh_cred_id>. If you have [configured credentials to read the error log](#), enter the credential ID for the SSH credential for a Linux server. For Windows servers, update the field to "Windows: <powershell_cred_id>".

NOTE: To discover a MySQL instance with one port, users must specify the "Range" header as "Range:<port_begin>-<port_end>". Meaning, "port_begin" and "port_end" must be the same port. For example, "Range:7706-7706".

CURL Options


- **CAPATH.** Type the CA path for your SSL certificate.
 - **SSLKEY.** Type the key path for your SSL certificate.
 - **SSLPEERCERT.** Type the certificate path for your SSL certificate.
4. For all other fields, use the default values.
 5. Click the **[Save As]** button.

Discovering MySQL Servers

To model and monitor your MySQL servers and instances, you must run a discovery session to discover the MySQL server that SL1 will use as the root device for monitoring the MySQL instances.

Several minutes after the discovery session has completed, the "MySQL: Discovery" Dynamic Application in the MySQL PowerPack should automatically align to the MySQL server, creating the MySQL server container. The remaining Dynamic Applications in the PowerPack will then discover, model, and monitor the remaining MySQL instances.

To discover the MySQL server that you want to monitor, perform the following steps:

1. On the **Devices** page () or the **Discovery Sessions** page (Devices > Discovery Sessions), click the **[Add Devices]** button. The **Select** page appears.
2. Click the **[Unguided Network Discovery]** button. Additional information about the requirements for discovery appears in the **General Information** pane to the right.
3. Click **[Select]**. The **Basic Information** page appears. Complete the following fields:

Step 1 Basic Information

Name *

Description (Optional)

Select the organization to add discovered devices to...

← Back

Next >

- **Name.** Type a unique name for this discovery session. This name is displayed in the list of discovery sessions on the **[Discovery Sessions]** tab.
- **Description.** Optional. Type a short description of the discovery session. You can use the text in this description to search for the discovery session on the **[Discovery Sessions]** tab.
- **Select the organization to add discovered devices to.** Select the name of the organization to which you want to add the discovered devices

4. Click **[Next]**. The **Credential Selection** page of the **Add Devices** wizard appears:

Step 2 Credential Selection

Choose credentials that connect your devices

Q. Type to search credentials

Create New Test Credentials

NAME	TYPE	TIMEOUT (MS)	LAST EDIT
<input type="checkbox"/> Auto_DB2_176	SOAP/XML	2000	May 17, 2022, 5:23 PM
<input type="checkbox"/> Auto_DB2_178	SOAP/XML	2000	May 17, 2022, 5:23 PM
<input type="checkbox"/> Auto_DB2_67	SOAP/XML	2000	May 17, 2022, 5:23 PM
<input type="checkbox"/> Auto_DB2_AZ_14	SOAP/XML	2000	May 17, 2022, 5:23 PM
<input type="checkbox"/> Auto_DB2_AZ_166	SOAP/XML	2000	May 17, 2022, 5:23 PM
<input type="checkbox"/> Auto_DB2_PS_AZ_14	SSH/Key	4000	May 17, 2022, 5:23 PM
<input type="checkbox"/> Auto_DB2_PS_AZ_166	PowerShell	18000	May 17, 2022, 5:23 PM
<input type="checkbox"/> Auto_DB2_SSH_176	SSH/Key	4000	May 17, 2022, 5:23 PM
<input type="checkbox"/> Auto_DB2_SSH_178	SSH/Key	4000	May 17, 2022, 5:23 PM
<input type="checkbox"/> Auto_DB2_SSH_67	SSH/Key	4000	May 17, 2022, 5:23 PM
<input type="checkbox"/> Auto_Microsoft_Azure	SOAP/XML	120000	May 3, 2022, 3:03 PM
<input type="checkbox"/> AUTO_MySQL_PS	PowerShell	180000	Aug 2, 2020, 5:13 PM
<input type="checkbox"/> AUTO_MySQL_SSH	SSH/Key	1500	May 9, 2022, 3:16 PM
<input type="checkbox"/> AWS Credential	SOAP/XML	2000	May 3, 2022, 11:45 AM
<input type="checkbox"/> AWS Credential - Proxy	SOAP/XML	2000	May 3, 2022, 11:45 AM
<input type="checkbox"/> AWS Credential - Specific Region	SOAP/XML	2000	May 3, 2022, 11:45 AM
<input type="checkbox"/> Azure Classic Credential SOAP	SOAP/XML	60000	May 3, 2022, 11:44 AM

← Back


Next >

5. On the **Credential Selection** page, locate and select the **credential** you created for the MySQL server.

6. Click **[Next]**. The **Discovery Session Details** page of the **Add Devices** wizard appears:

7. Complete the following fields:

- **List of IPs/Hostnames.** Type the IP address(es) of the MySQL server you want to discover.
- **Which collector will monitor these devices?** Required. Select an existing collector to monitor the discovered devices.
- **Run after save.** Select this option to run this discovery session as soon as you save the session.

In the **Advanced options** section, click the down arrow icon () to complete the following fields:

- **Discover Non-SNMP.** Enable this setting.
 - **Model Devices.** Enable this setting.
8. Click **[Save and Run]** if you enabled the Run after save setting, or **[Save and Close]** to save the discovery session. The **Discovery Sessions** page (Devices > Discovery Sessions) displays the new discovery session.
9. If you selected the **Run after save** option on this page, the discovery session runs, and the **Discovery Logs** page displays any relevant log messages. If the discovery session locates and adds any devices, the **Discovery Logs** page includes a link to the **Device Investigator** page for the discovered device.

Discovering MySQL Servers in the SL1 Classic User Interface

To model and monitor your MySQL servers and instances, you must run a discovery session to discover the MySQL server that SL1 will use as the root device for monitoring the MySQL instances.

Several minutes after the discovery session has completed, the "MySQL: Discovery" Dynamic Application in the MySQL PowerPack should automatically align to the MySQL server, creating the MySQL server container. The remaining Dynamic Applications in the PowerPack will then discover, model, and monitor the remaining MySQL instances.

To discover the MySQL server that you want to monitor, perform the following steps:

1. Go to the **Discovery Control Panel** page (System > Manage > Classic Discovery).
2. In the **Discovery Control Panel**, click the **[Create]** button.

- The **Discovery Session Editor** page appears. In the **Discovery Session Editor** page, define values in the following fields:

The screenshot shows the 'Discovery Session Editor | Editing Session [1]' window. It is divided into four main sections:

- Identification Information:** Includes fields for 'Name' (SL1) and 'Description'.
- IP and Credentials:**
 - IP Address/Hostname Discovery List:** A text field containing '10.2.21.17' and a 'Browse...' button.
 - SNMP Credentials:** A list of credential types including 'Cisco SNMPv2 - Example', 'Cisco: CSP SNMP Port 1610 Example', 'Dell EMC: Isilon SNMPv2 Example', 'EM7 Default V2', 'EM7 Default V3', and 'IPSLA Example'.
 - Other Credentials:** A list of credential types including 'Cisco CE Series status', 'Cisco VOS SOAP - Example', 'Cisco: Conductor Example (Discov', 'Cisco: Conductor Example (Virtua', 'Dell EMC XtremIO Example', 'Dell EMC: Isilon SOAP Example', 'Dynatrace Credential Example', 'MySQL Example Credential', and 'MySQL SL1'.
- Detection and Scanning:**
 - Initial Scan Level:** '[System Default (recommended)]'
 - Scan Throttle:** '[System Default (recommended)]'
 - Port Scan All IPs:** '[System Default (recommended)]'
 - Port Scan Timeout:** '[System Default (recommended)]'
 - Detection Method & Port:** A list of methods including 'UDP: 161 SNMP', 'TCP: 1 - tcpmux', 'TCP: 2 - compressnet', 'TCP: 3 - compressnet', 'TCP: 5 - rje', 'TCP: 7 - echo', 'TCP: 9 - discard', 'TCP: 11 - systat', 'TCP: 13 - daytime', and 'TCP: 15 - netstat'.
 - Interface Inventory Timeout (ms):** '600000'
 - Maximum Allowed Interfaces:** '10000'
 - Bypass Interface Inventory:** An unchecked checkbox.
- Basic Settings:**
 - Discover Non-SNMP:** Checked checkbox.
 - Model Devices:** Checked checkbox.
 - DHCP:** Unchecked checkbox.
 - Device Model Cache TTL (h):** '2'
 - Collection Server PID:** '[50C-MUD-DCU-19]'
 - Organization:** '[System]'
 - Add Devices to Device Group(s):** A list containing 'None' and 'Servers'.
 - Apply Device Template:** '[Choose a Template]'

At the bottom, there are 'Save' and 'Save As' buttons, and a 'Log All' checkbox which is checked.

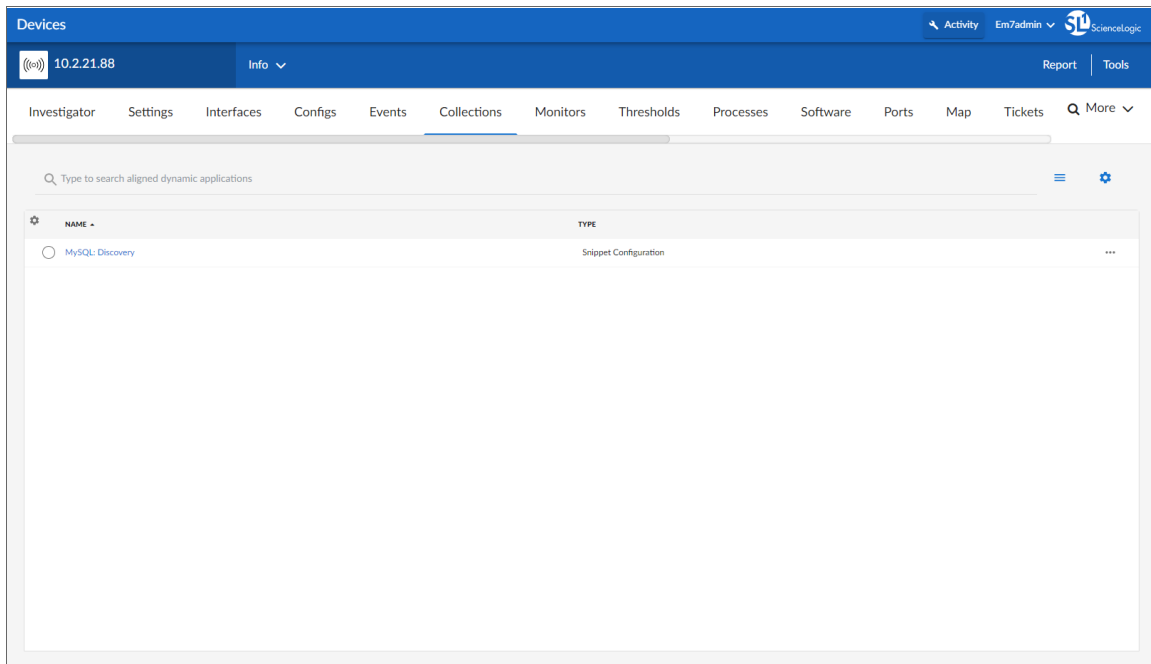
- **IP Address/Hostname Discovery List.** Type the IP address(es) of the MySQL server you want to discover.
 - **Other Credentials.** Select the SOAP/XML credential(s) you created for the MySQL server.
 - **Discover Non-SNMP.** Select this checkbox.
 - **Model Devices.** Select this checkbox.
- Optionally, you can enter values in the other fields on this page. For more information about the other fields on this page, see the **Discovery & Credentials** manual.
 - Click the **[Save]** button to save the discovery session and then close the **Discovery Session Editor** window.
 - The discovery session you created appears at the top of the **Discovery Control Panel** page. Click its lightning-bolt icon (⚡) to run the discovery session.
 - The **Discovery Session** window appears. When the cluster root device(s) are discovered, click the device icon (🖨) to view the **Device Properties** page for each device.

Verifying Discovery and Dynamic Application Alignment

To verify that SL1 has automatically aligned the correct Dynamic Applications during discovery, perform the following steps:

1. After discovery has completed, locate the device for the MySQL server in the **Devices** page.
2. From the **Device Investigator** page for the MySQL server, click the **[Collections]** tab.
3. The "MySQL: Discovery" Dynamic Application for the server is automatically aligned during discovery.

NOTE: It can take several minutes after the discovery session has completed for Dynamic Applications to appear in the **Dynamic Application Collections** page.



The MySQL server container will then be created and the "MySQL: Instance Discovery" Dynamic Application will auto-align to the server container. The MySQL server container will then discover, model, and monitor the remaining MySQL instances.

The following Dynamic Applications will auto-align to the MySQL instances:

- MySQL: Instance Commands Performance
- MySQL: Instance Handler Performance
- MySQL: Instance InnoDB Buffer Pool Performance
- MySQL: Instance InnoDB Data Performance
- MySQL: Instance InnoDB Row Performance
- MySQL: Instance Overall Performance

- MySQL: Instance Sort and Select Performance
- MySQL: Instance Table Locking Performance
- MySQL: Instance Threads and Connections Performance
- MySQL: Instance Configuration
- MySQL: Instance InnoDB Configuration

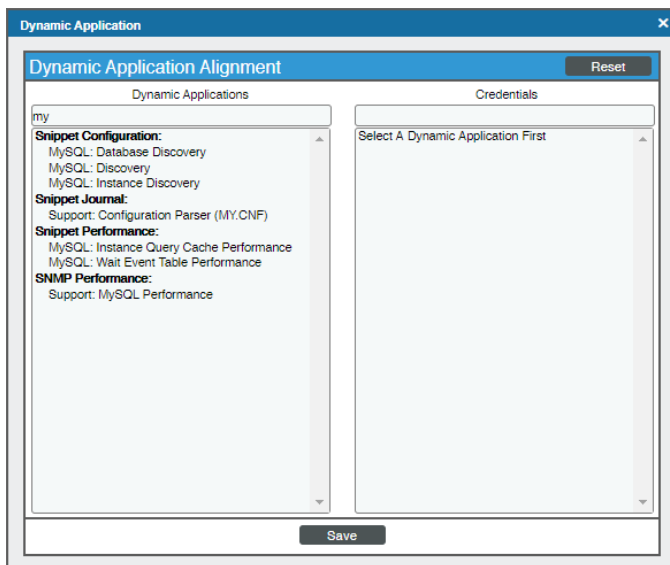
The following Dynamic Applications will not automatically align during discovery and will need to be manually aligned:

- MySQL: Events Errors Summary Configuration
- MySQL: Performance Schema Statements Configuration
- MySQL: Performance Schema Summary Statement Configuration
- MySQL: Process List Configuration
- MySQL: Statements With Error/Warning Configuration

NOTE: To collect data for the manually-aligned Dynamic Applications, you will need to enable the system database and performance_schema in the MySQL instance.

To manually align Dynamic Applications, perform the following steps:

1. Go to the **Device Manager** page (Devices > Device Manager) and click the wrench icon (🔧) for the device you want to manually align Dynamic Applications to.
2. Click the **[Collections]** tab, then click the **[Action]** button and select *Add Dynamic Application*. The **Dynamic Application Alignment** page appears:



3. In the **Dynamic Applications** field, select the Dynamic Application you want to align.
4. In the **Credentials** field, select the credential specified in the table.

5. Click the **[Save]** button.
6. Repeat steps 1-4 for the other unaligned Dynamic Applications.

Enabling the Slow Query Log in MySQL or MariaDB

To view the metrics related to slow queries collected in the "MySQL: Instance Configuration" Dynamic Application, you will need to enable the Slow Query Log. To enable the slow query log:

1. Log in to your server as the `root` user via SSH.
2. Open the `my.cnf` file with a text editor and add the following under the `mysqld` section:

```
slow_query_log = 1
```

```
slow-query_log_file = /var/log/mysql-slow.log
```

```
long_query_time = 2
```

NOTE: If you are using MySQL version 5.6 or older, use the `log-slow-queries` variable instead of the `slow-query_log_file` variable.

3. Run the following commands to create the `/var/log/mysql-slow.log` file and set its user as the `mysql` user:

```
touch /var/log/mysql-slow.log
```

```
chown mysql:mysql /var/log/mysql-slow.log
```

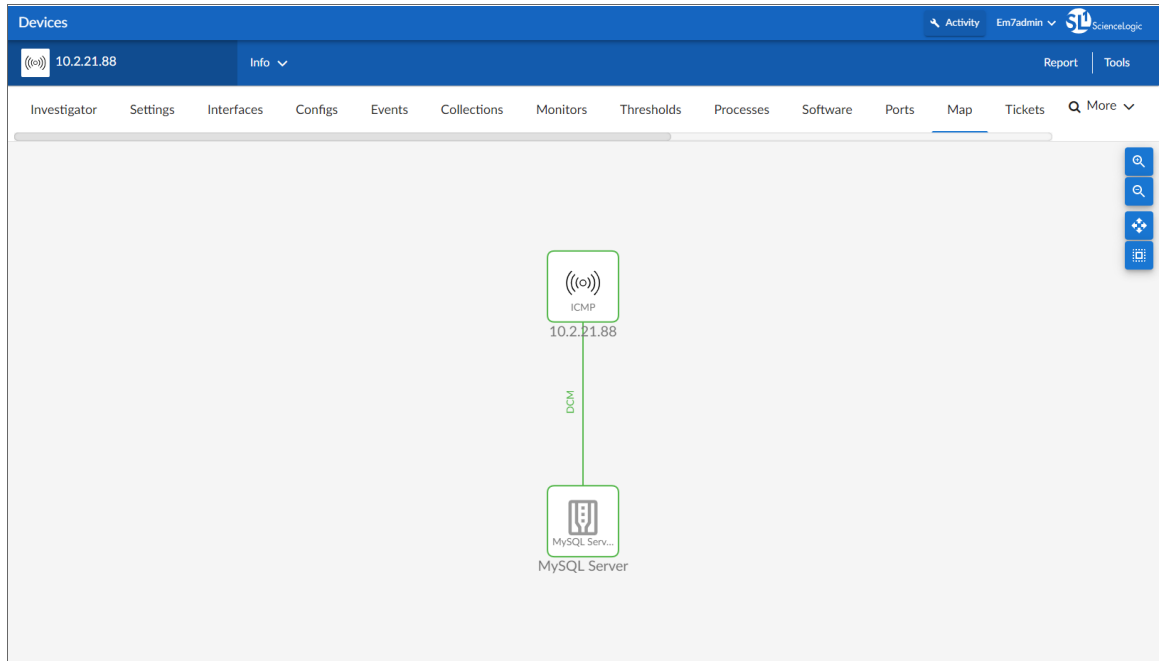
4. Restart MySQL or MariaDB by running the following command:

```
/usr/local/cpanel/scripts/restartsrv_mysql
```

Viewing MySQL Component Devices

In addition to the **Device Manager** page (Registry > Devices > Device Manager), you can view the MySQL server and all associated component devices in the following places in the user interface:

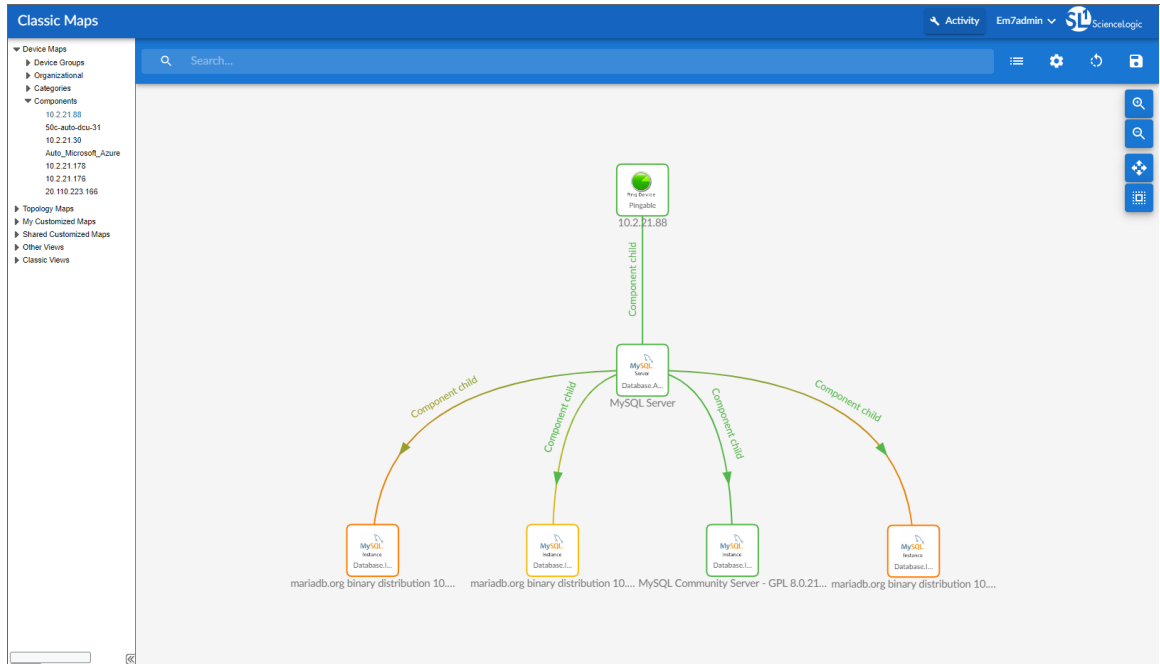
- The **Device Investigator** Map page (click **Map** in the **Device Investigator** page) displays a map of a particular device and all of the devices with which it has parent-child relationships. Double-clicking any of the listed devices reloads the page to make the selected device the primary device.



- The **Device Components** page (Devices > Device Components) displays a list of all root devices and component devices discovered by SL1 in an indented view, so you can easily view the hierarchy and relationships between child devices, parent devices, and root devices. To view the component devices associated with a MySQL server, find the MySQL server and click its plus icon (+):

Device Name	IP Address	Device Category	Device Class Sub-class	DID	Organization	Current State	Collection Group	Collection State
10.2.21.176	10.2.21.176	Pingable	Ping ICMP	921	Auto_IBM_DB2	Healthy	CUG	Active
10.2.21.178	10.2.21.178	Pingable	Ping ICMP	920	Auto_IBM_DB2	Healthy	CUG	Active
10.2.21.30	10.2.21.30	Pingable	Ping ICMP	351	AUTO_MYSQL_ORG	Healthy	CUG	Active
MySQL Server	--	Application	Oracle MySQL Server	353	AUTO_MYSQL_ORG	Healthy	CUG	Active
MariaDB Server 10.4.18-MariaDB 7706	--	Instance	Oracle MySQL Instance	357	AUTO_MYSQL_ORG	Healthy	CUG	Active
10.2.21.88	10.2.21.88	Pingable	Ping ICMP	349	AUTO_MYSQL_ORG	Healthy	CUG	Active
MySQL Server	--	Application	Oracle MySQL Server	352	AUTO_MYSQL_ORG	Healthy	CUG	Active
mariadb.org binary distribution 10.4.13-MariaDB	--	Instance	Oracle MySQL Instance	358	AUTO_MYSQL_ORG	Minor	CUG	Active
mariadb.org binary distribution 10.3.13-MariaDB	--	Instance	Oracle MySQL Instance	356	AUTO_MYSQL_ORG	Major	CUG	Unavailable
mariadb.org binary distribution 10.5.4-MariaDB	--	Instance	Oracle MySQL Instance	360	AUTO_MYSQL_ORG	Major	CUG	Unavailable
MySQL Community Server - GPL 8.0.21.3308	--	Instance	Oracle MySQL Instance	359	AUTO_MYSQL_ORG	Healthy	CUG	Active
20.110.223.166	20.110.223.166	Pingable	Ping ICMP	923	Auto_IBM_DB2	Healthy	CUG	Active
50c-auto-dcu-31	10.2.21.31	EM7	ScienceLogic, Inc. EM7 Data Collector	350	AUTO_MYSQL_ORG	Major	CUG	Active
MySQL Server	--	Application	Oracle MySQL Server	354	AUTO_MYSQL_ORG	Healthy	CUG	Active

- The **Component Map** page (Classic Maps > Device Maps > Components) allows you to view devices by root node and view the relationships between root nodes, parent components, and child components in a map. This makes it easy to visualize and manage root nodes and their components. SL1 automatically updates the **Component Map** as new component devices are discovered. The platform also updates each map with the latest status and event information. To view the map for a MySQL server, go to the **Component Map** page and select the map from the list in the left NavBar. To learn more about the **Component Map** page, see the **Maps** manual.



Chapter

3

MySQL Dashboards

Overview

The following sections describe the device dashboards that are included in the MySQLPowerPack:

NOTE: This dashboard can be viewed only in the SL1 classic user interface.

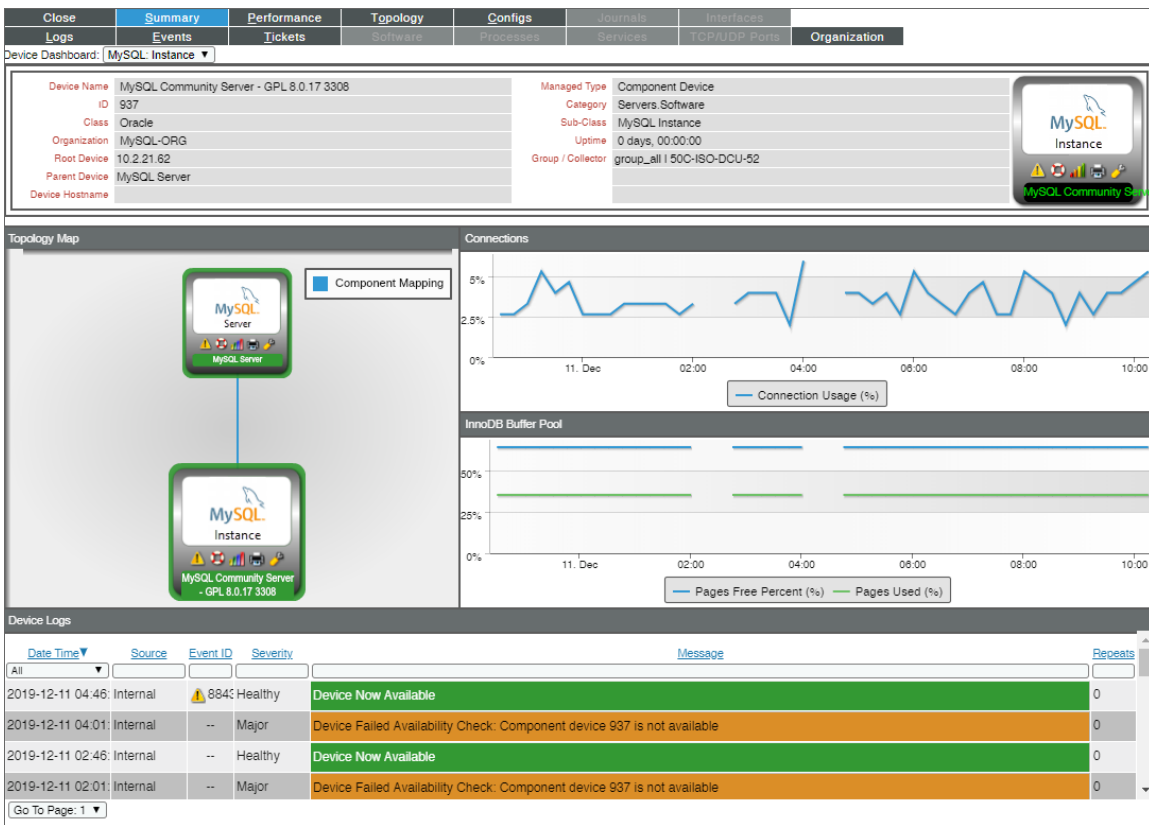
This chapter covers the following topics:

<i>Device Dashboards</i>	30
--------------------------------	----

Device Dashboards

The MySQL PowerPack includes a device dashboard that provides summary information for MySQL instances.

MySQL: Instance



The MySQL: Instance device dashboard displays the following information:

- A topology map
- A Connections line chart that displays connection usage over a specified period of time
- An InnoDB Buffer Pool line chart that displays Pages Free Percent and Pages Used over a specified period of time
- Device logs

© 2003 - 2024, ScienceLogic, Inc.

All rights reserved.

LIMITATION OF LIABILITY AND GENERAL DISCLAIMER

ALL INFORMATION AVAILABLE IN THIS GUIDE IS PROVIDED "AS IS," WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED. SCIENCELOGIC™ AND ITS SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT.

Although ScienceLogic™ has attempted to provide accurate information on this Site, information on this Site may contain inadvertent technical inaccuracies or typographical errors, and ScienceLogic™ assumes no responsibility for the accuracy of the information. Information may be changed or updated without notice. ScienceLogic™ may also make improvements and / or changes in the products or services described in this Site at any time without notice.

Copyrights and Trademarks

ScienceLogic, the ScienceLogic logo, and EM7 are trademarks of ScienceLogic, Inc. in the United States, other countries, or both.

Below is a list of trademarks and service marks that should be credited to ScienceLogic, Inc. The ® and ™ symbols reflect the trademark registration status in the U.S. Patent and Trademark Office and may not be appropriate for materials to be distributed outside the United States.

- ScienceLogic™
- EM7™ and em7™
- Simplify IT™
- Dynamic Application™
- Relational Infrastructure Management™

The absence of a product or service name, slogan or logo from this list does not constitute a waiver of ScienceLogic's trademark or other intellectual property rights concerning that name, slogan, or logo.

Please note that laws concerning use of trademarks or product names vary by country. Always consult a local attorney for additional guidance.

Other

If any provision of this agreement shall be unlawful, void, or for any reason unenforceable, then that provision shall be deemed severable from this agreement and shall not affect the validity and enforceability of any remaining provisions. This is the entire agreement between the parties relating to the matters contained herein.

In the U.S. and other jurisdictions, trademark owners have a duty to police the use of their marks. Therefore, if you become aware of any improper use of ScienceLogic Trademarks, including infringement or counterfeiting by third parties, report them to Science Logic's legal department immediately. Report as much detail as possible about the misuse, including the name of the party, contact information, and copies or photographs of the potential misuse to: legal@sciencelogic.com. For more information, see <https://sciencelogic.com/company/legal>.

ScienceLogic

800-SCI-LOGIC (1-800-724-5644)

International: +1-703-354-1010