Installing and Updating an Extension

MySQL supports a variety of **plugins** that extend the functionality of the database engine. These plugins add features like authentication methods, full-text search improvements, audit logging, and more. Popular examples include `auth_socket`, `validate_password`, and `audit_log`. In Elestio-hosted MySQL instances, many common plugins are already available and can be enabled or disabled as needed. This guide explains how to install, manage, and troubleshoot MySQL plugins and verify compatibility with different MySQL versions.

Installing and Enabling Plugins

In MySQL, plugins are usually installed globally at the server level, not per-database. If the plugin binaries are available, they can be loaded dynamically at runtime without restarting the server.

Start by connecting to your MySQL database using a client like mysql:

```
mysql -u root -p -h your-elestio-hostname
```

To enable a plugin, use the INSTALL PLUGIN command. For example, to enable the `validate_password` plugin:

```
INSTALL PLUGIN validate_password SONAME 'validate_password.so';
```

You can verify that the plugin is installed by checking the plugin list:

```
SHOW PLUGINS;
```

To enable a plugin automatically at server startup, add its configuration in the my.cnf file. However, for managed Elestio instances, this may require support team intervention unless custom configuration access is provided.

Checking Plugin Availability & Compatibility

Plugins must be compiled for the specific MySQL version and platform. Before upgrading MySQL or installing a new plugin, verify that a compatible version is available for your target setup. You can find plugin binaries under the plugin_dir, which you can locate with:

```
SHOW VARIABLES LIKE 'plugin_dir';
```

To check if a specific plugin is installed and active:

```
SELECT * FROM INFORMATION_SCHEMA.PLUGINS WHERE PLUGIN_NAME = 'validate_password';
```

If a plugin is incompatible or missing from the plugin_dir, the server will return an error when you attempt to install it. In this case, contact Elestio support to request installation or confirm version compatibility.

Updating or Uninstalling Plugins

After a MySQL version upgrade, some plugins may need to be reinstalled or updated. If a plugin is malfunctioning after an upgrade, it is good practice to uninstall and reinstall it:

```
UNINSTALL PLUGIN validate_password;
INSTALL PLUGIN validate_password SONAME 'validate_password.so';
```

Not all plugins support automatic upgrades. You should consult plugin-specific documentation or Elestio's compatibility matrix before proceeding.

Troubleshooting Common Plugin Issues

Issue	Cause	Resolution
Can't open shared library	Plugin binary not found in plugin_dir	Check if the `.so` file exists and has correct permissions; contact Elestio if needed
Plugin already installed	Attempting to install a plugin that is already active	Use `SHOW PLUGINS` to verify and avoid duplicate installation
Permission denied	Current user lacks `SUPER` privilege	Log in as a user with `SUPER` or administrative rights
Plugin is not loaded at startup	Plugin not defined in configuration file	Contact Elestio to add it to the MySQL startup config (`my.cnf`)

Security Considerations

Plugins can have significant control over database behavior. Only enable trusted plugins from verified sources. Avoid enabling plugins you do not need, as they can introduce security or performance risks. Always test plugin behavior in a staging environment before deploying to production.