QAD Adaptive Applications

Enterprise Edition

Installation Guide

QAD Production Execution

70-3489-4.0.1
QAD Production Execution 4.0.1
QAD Enterprise Edition
February 2023

QADProductionExecution_IG_v401.pdf/r9m/r9m

QAD Inc.
100 Innovation Place
Santa Barbara, California 93108
Phone (805) 566-6000
http://www.qad.com

Contents

Change Summary

Change Summary

The following table summarizes significant differences between this document and previous versions.

Date/Version	Description
February 2023/v4.0.1	Installing MariaDB Using the Script section: Added a step detailing a possible error that may arise.
	Manually Installing Maria DB section: Added a step to restart firewall.
	Removed the Restart Firewall section
	Added Post Installation Steps chapter
	Post Installation Steps chapter: Added Setup Instances section
	Post Installation Steps chapter: Added Starting PE and Setting up PE Admin Permissions section
December 2022/v4.0.0-Rev1	Added PE Hardware and Sizing Requirements section
	Added the Create Internal User Source section
	Updated the User Source Table
October 2022/v4.0.0	Second Release
December 2021/v3.5.0	First release

Chapter 1:

Introduction

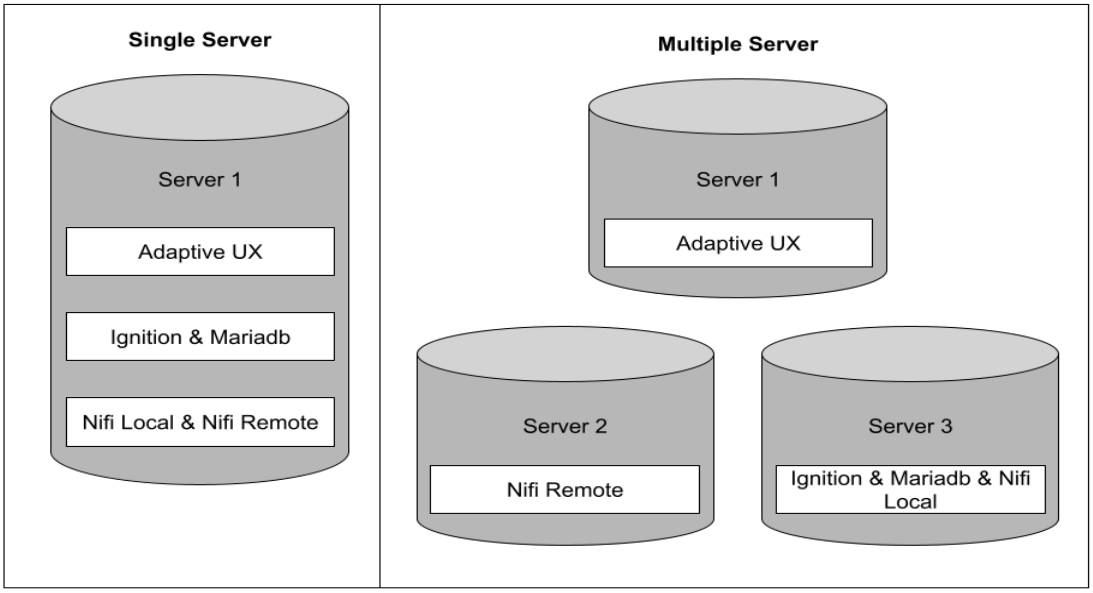
This chapter provides an overview of the QAD Production Execution installation process and the software and environment requirements:

Overview

Installing QAD Production Execution requires the following three sets of artifacts:

1. Adaptive UX-related components. Installed with the `peaux-app` YAB package.
2. Ignition and MariaDB related components. Installed with the manufacturer's installation package.
3. Nifi flow related-components. Installed with the `penifilocal and penifiremote` YAB package.

Depending on the implementation needs and goals, each of these pieces can be installed on a single server or on multiple servers. The following diagram shows the installation options.



The `peaux-app`, `penifiremote and penifilocal` packages can be installed using yab. Ignition and MariaDB-related artifacts require manual installation.

Cloud Installation

QAD recommends installing QAD Production Execution in the Cloud; therefore, the instructions in this installation guide are for Cloud installations only. For information about On-Premise installations, please contact QAD Services.

Software and Environment Requirements

The following sections describe the software and environment and other requirements for installing QAD Production Execution on a single server or on multiple servers.

Single-Server Requirements

If you are installing all the sets of artifacts on a single server, the server must have the following software and environments to install software for QAD Production Execution:

- Enterprise Edition 2019EE or above
- QAD Automation Solutions: Data Collection v3.2.2.10 or above
- QAD Automation Solutions: Label Printing Services v3.2.0.0 or above
- QAD Production Orders 3.2.4.0 or above
- Nifi 1.15.1 or above
- Nifi QRA Authentication 1.1.0.9 or above
- Java Runtime Environment (JRE) 1.8 or above
- Progress 11.7 or above
- YAB 1.14.1 or above (yab-ee-app 1.14.1.0 or above)
- qad-enterprise-platform v3.14.1.12 and above
- If installing on a UNIX Server, the zip and unzip packages must be installed
- Production Execution supports Ignition v8.0.15 or above. For the version of Ignition being installed, see the Ignition website for software and hardware requirements: https://inductiveautomation.com/downloads/
- Production Execution supports MariaDB v10.5.x. See the MariaDB website for software and hardware requirements: https://mariadb.com/

Multiple Server Requirements

If you are installing the artifacts on separate servers, you must have the following software and environment to install software for QAD Production Execution:

Adaptive UX Components

The server where the Adaptive UX components are installed must have the following software and environment to install software for QAD Production Execution:

- Enterprise Edition 2019EE or above
- QAD Automation Solutions: Data Collection v3.2.2.10 or above
- QAD Automation Solutions: Label Printing Services v3.2.0.0 or above
- QAD Production Orders 3.2.4.0 or above

- Java Runtime Environment (JRE) 1.8 or above
- Progress 11.7 or above
- YAB 1.14.1 or above (yab-ee-app 1.14.1.0 or above)
- qad-enterprise-platform v3.14.1.12 or above

Nifi Flow Components

The Server where Nifi Local or Nifi Remote components are installed must have the following software and environment to install software for QAD Production Execution:

- Yab Core 1.16 or above (if installing using yab)
- Nifi 1.11.3.0 or above
- Nifi QRA Authentication 1.1.0.9 or above
- Nifi Toolkit 1.15.1.0
- Java Runtime Environment (JRE) 1.8 or above

Ignition and MariaDB Components

The server where the Ignition and MariaDB components are installed must have the following software and environment to install software for QAD Production Execution:

- Supports Ignition v8.0.15 or above. For the version of Ignition being installed, see the Ignition website for software and hardware requirements: https://inductiveautomation.com/downloads/
- Supports MariaDB v10.5.x. See the MariaDB website for software and hardware requirements: https://mariadb.com/
- PE Ignition package (downloaded from the QAD store during the installation process). This package contains artifacts required for Ignition and MariaDB configuration.
- Python version 3 (if using MariaDB installation script)

Recommended Memory Allocation

Ignition. Recommended to run with at least 8GB of memory. Ignition.conf file needs to be updated for providing the recommended memory.

Mariadb. Recommended configuration. This is set in mysql configuration file (my.cnf):

innodb_buffer_pool_size = 11G

read_buffer_size = 1G

read_rnd_buffer_size = 1G

sort_buffer_size = 1G

max_connections = 500

PE Hardware and Software Sizing Requirements

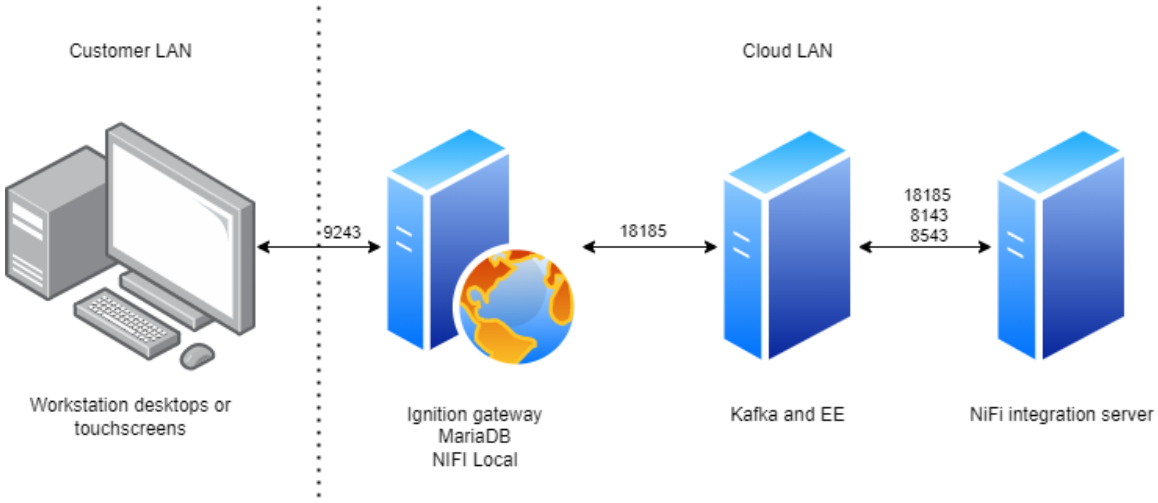
Overview

All hardware specifications in this section are minimum requirements for a single instance of QAD Production Execution. These specifications assume that certain server components (NiFi, Ignition) are installed on their own dedicated server. Combining these server components (or multiple QAD Production execution software instances) on a single server will negatively impact performance and require additional hardware resources to run appropriately.

If additional concurrent users are using the system, additional resources will be required. Similarly, if the servers are used for applications or purposes outside of dedicated hosting of QAD Production Execution, the server resource requirements should be scaled accordingly.

Network Topology Standards and Requirements

The standard network configuration for QAD Production Execution is to have a Production Execution Ignition gateway, also containing the MariaDB server, the NiFi server, and a NiFi integration server within the private Cloud network.



Server Hardware and Software Requirements

The customer's IT department should be aware of the minimum requirements for the installation and support of QAD Production Execution. If any of the following requirements are lacking, QAD Production Execution cannot be installed.

Hardware Requirements

Concurrent Clients	Production Note: Production servers must be dedicated to production.	Development / Test Note: Ignition gateway server should be dedicated to a non-production environment. NiFi and integration servers can be combined for non-production environments, but must be separated from production.
1-50	Ignition gateway server ● Hard Drive Free Space [100 GB] ● Processor [2 CPUs (4 cores total)] ● Memory [16 GB] ○ 8 GB Ignition ○ 4 GB MariaDB ○ 4 GB NiFi **Assumption: Combination of customer released Work Orders and Operations is less than 2500 instances. Note: If the number of instances is greater than described in the assumption above, please consult the architecture team on recommendations forward.	Ignition gateway server ● Hard Drive Free Space [100 GB] ● Processor [2 CPUs (4 cores total)] ● Memory [12 GB] ○ 4 GB Ignition ○ 4 GB MariaDB ○ 4 GB NiFi **Assumption: Combination of customer released Work Orders and Operations is less than 2500 instances. Note: If the number of instances is greater than described in the assumption above, please consult the architecture team on recommendations forward.
	NiFi Integration server ● Hard Drive Free Space [100 GB] ● Processor [2 CPUs (4 cores total)] ● Memory [8 GB]	NiFi Integration server ● Hard Drive Free Space [100 GB] ● Processor [2 CPUs (4 cores total)] ● Memory [4 GB]
	EE/AUX Server ● Extra Hard Drive Free Space [100 GB] ● Memory [2 GB] ○ 1 GB Kafka ○ 1 GB WSA and AppServer (with 5 agents)	EE/AUX Server ● Extra Hard Drive Free Space [100 GB] ● Memory [2 GB] ○ 1 GB Kafka ○ 1 GB WSA and AppServer (with 5 agents)

<table>
<tr><td rowspan="3">50-250</td><td>Ignition gateway server
Hard Drive Free Space [250 GB]
Processor [4 CPUs (8 cores total)]
Memory [32 GB]
16 GB Ignition
8 GB MariaDB
8 GB NiFi
**Assumptions

Combination of customer released Work Orders and Operations is less than 10,000 instances.

Note: If the number of instances is greater than described in the assumption above, please consult the architecture team on recommendations forward.</td><td>Ignition gateway server
Hard Drive Free Space [250 GB]
Processor [3 CPUs (6 cores total)]
Memory [24 GB]
12 GB Ignition
6 GB MariaDB
6 GB NiFi
**Assumptions

Combination of customer released Work Orders and Operations is less than 10,000 instances.

Note: If the number of instances is greater than described in the assumption above, please consult the architecture team on recommendations forward.</td></tr>
<tr><td>NiFi Integration server
Hard Drive Free Space [250 GB]
Processor [4 CPUs (8 cores total)]
Memory [16 GB]</td><td>NiFi Integration server
Hard Drive Free Space [250 GB]
Processor [2 CPUs (4 cores total)]
Memory [8 GB]</td></tr>
<tr><td>EE/Adaptive UX Server
Extra Hard Drive Free Space [100 GB]
Memory [2 GB]
1 GB Kafka
1 GB WSA and AppServer (with 10 agents)</td><td>EE/Adaptive UX Server
Extra Hard Drive Free Space [100 GB]
Memory [2 GB]
1 GB Kafka
1 GB WSA and AppServer (with 10 agents)</td></tr>
<tr><td rowspan="3">250+</td><td colspan="2">Ignition gateway server

Note: With 250+ clients on one gateway it is advised to cluster Ignition into a gateway network consisting of multiple gateways of 250 clients. For this setup, please contact your QAD representative.</td></tr>
<tr><td>NiFi Integration server

Double the resources per 250 clients:
Hard Drive Free Space [250 GB]
Processor [4 CPUs (8 cores total)]
Memory [16 GB]</td><td>NiFi Integration server

Double the resources per 250 clients:
Hard Drive Free Space [250 GB]
Processor [2 CPUs (4 cores total)]
Memory [8 GB]</td></tr>
<tr><td>EE/Adaptive UX Server
Extra Hard Drive Free Space [100 GB]</td><td>EE/Adaptive UX Server
Extra Hard Drive Free Space [100 GB]</td></tr>
</table>

	• Memory [2 GB]
	  ○ 1 GB Kafka
	  ○ 1 GB WSA and AppServer (with 10 + n * 5 agents, n = number of clients divided by 250 clients)
		• Memory [2 GB]
		  ○ 1 GB Kafka
		  ○ 1 GB WSA and AppServer (with 10 + n * 5 agents, n = number of clients divided by 250 clients)

Software Requirements

Ignition Gateway Server Requirements Note: The following requirements apply to production, dev, and test servers.	**Notes**
Linux 64 bit Supported Operating Systems: • Red Hat V 8 • CentOS V 8	Both virtual servers and physical servers are supported.
Ignition V 8.1.17	An external license is required. Currently this is not provided with the package and needs to be downloaded at https://inductiveautomation.com/downloads/
NiFi V 1.15.1	This is only provided in a package when installed with YAB. Note: This is not recommended for customer installs. This needs to be downloaded at https://archive.apache.org/dist/nifi/1.15.1/nifi-1.15.1-bin.tar.gz
MariaDB V 10.5+	Any version higher than version 10.5 will work. Note: This is not provided in the packages, but it can be installed with the yum package manager.
SSL Certificate	All QAD Production Execution traffic takes place over the SSL/TLS protocol and therefore requires that a Digital Certificate be assigned.

NiFi Integration Server Requirements Note: The following requirements apply to production, dev, and test servers.	Notes
Linux 64-bit Supported Operating Systems: ● Red Hat V 8 ● CentOS V 8	Both virtual servers and physical servers are supported.
NiFi V 1.15.1	Only provided in a package when installed with YAB. Note: This is not recommended for customer installs. This needs to be downloaded at https://archive.apache.org/dist/nifi/1.15.1/nifi-1.15.1-bin.tar.gz
TCP/IP	The TCP/IP protocol must be available. Firewall needs to be set open on ports 18185, 8143, and 8543

Client Hardware and Software Requirements

Hardware Requirements

Client Requirements Note: The following requirements apply to running against production, dev, and/or test sites.	Client Recommended Note: The following requirements apply to running against production, dev, and/or test sites.	Notes
Desktop / Laptop ● Hard Drive Free Space [10 GB] ● Processor [dual-core] ● Memory [8 GB] ● 1920x1080 resolution ● 10 mb connection to Ignition server	Desktop / Laptop ● Hard Drive Free Space [10 GB] ● Processor [quad-core] ● Memory [16 GB] ● 1920x1080 resolution ● 25 mb connection to Ignition server	
Mobile Note: Currently mobile devices are not supported.	Mobile Note: Currently mobile devices are not supported.	

Software Requirements

Client Requirements Note: The following requirements apply to running against production, dev, and/or test sites.	**Notes**
Desktop / Laptop Supported Client Operating Systems: ● Windows Server 2008/2012/2016/2019 ● Windows 7.8, and 10 ● MacOS ● Ubuntu 12.4 or related linux distro's	

Technical Installation Checklist

Note: This should be completed by the customer's IT department.

Customer Name:	
IT Contact Name:	IT Contact Number:

The purpose of this section is to collect information about the systems in which QAD Production Execution will be installed and to reduce the possibility of exceptions during the installation and configuration of the system.

Production Note: Production servers must be dedicated to production.	**Development / Test** Note: The Ignition gateway server should be dedicated per non production environment. NiFi integration server can be combined for non production environments but must be separated from production.
Ignition gateway server ● Name: [ ] ● Operating System: [ ] ● Hard Drive Free Space [ ] ● Processor (include speed and total number of cores) [ ] ● Memory [ ] ● Preferred Drives/Folders: ○ Ignition [ ]	Ignition gateway server ● Name: [ ] ● Operating System: [ ] ● Hard Drive Free Space [ ] ● Processor (include speed and total number of cores) [ ] ● Memory [ ] ● Preferred Drives/Folders: ○ Ignition [ ]

○ MariaDB [ ] ○ NiFi [ ]	○ MariaDB [ ] ○ NiFi [ ]
NiFi Integration server ● Name: [ ] ● Operating System: [ ] ● Hard Drive Free Space [ ] ● Processor (include speed and total number of cores) [ ] ● Memory [ ] ● Preferred Drive/Folder [ ]	NiFi Integration server ● Name: [ ] ● Operating System: [ ] ● Hard Drive Free Space [ ] ● Processor (include speed and total number cof cores) [ ] ● Memory [ ] ● Preferred Drive/Folder [ ]

Chapter 2:

Installing QAD Production Execution

This chapter provides detailed instructions on installing QAD Production Execution:

Installation Overview

This chapter describes how to install the three sets of artifacts for QAD Production Execution:

- Ignition
- Maria DB
- Adaptive UX

The installation instructions in this install guide are for Cloud installations. The installation packages for each set of artifacts do not need to be downloaded and are installed from the Cloud catalog.

Note: For information about On-Premise installations, please contact QAD Services.

Installing QAD Production Execution

Pre-Installation Tasks

Back Up Databases

Before installing the three sets of artifacts for QAD Production Execution, back up your databases and supporting environments.

To back up all databases, execute the command:

```
> yab database-backup
```

Alternatively, to back up a specific database, execute the command:

```
> yab database-[INSTANCE]-backup
```

Download PE Ignition Package

Download the PE Ignition package from the QAD Store onto the server where Ignition and MariaDB will be installed. The PE Ignition package contains artifacts for MariaDB and Ignition configuration.

Installing Ignition and MariaDB Artifacts

The installation steps for single server architecture and multiple server architecture are the same for Ignition and MariaDB artifacts.

Stop Firewall

Before installing Ignition and the Maria DB artifacts, stop the firewall with the following command. Make sure ports are open to red-hat linux repositories.

```
service firewalld stop
```

Installing MariaDB

You can install MariaDB by using the script or manually. You must choose *only one* of these options to install MariaDB.

Installing MariaDB Using the Script

1. Download the PEIgnition package on the server.
2. Use the script <download directory>/manualinstall/mariaDB/script/mariadb_install.py provided in the package for installation of MariaDB.
3. The script provides help on setting parameters that are provided as input to the script.

```
$ python3 mariadb_install.py --help

   -h, --help
```

Arguments	Description
-h --help	Show this help message and exit
-P <port>, --port <port>	MariaDB server port (default: 3306)
--mariadb-data-dir <MARIADB_DATA_DIR>	MariaDB data directory (default: /qond/apps/mysql)
--mariadb-mysqld-config <MARIADB_MYSQLD_CONFIG>	A JSON string to update MariaDB my.cnf mysqld section's configuration. (default: None) e.g {"max_connections": 500, "sort_buffer_size": "1G"}
--mariadb-version <MARIADB_VERSION>	MariaDB server version (default: 10.5)
--root-password <ROOT_PASSWORD>	MariaDB root user password (default: None)
-u <USER>, --user <USER>	MariaDB user for production order integration (default: None)

-p <PASSWORD>, --password <PASSWORD>	Integration MariaDB user's password (default: None)
--database <DATABASE>	QAD Production Execution database name (default: qad)
--schema-dir <SCHEMA_DIR>	Ignition schema files location (default: None)
--seed-data-dir <SEED_DATA_DIR>	Seed data files location (default: None)
--skip-mariadb-install	Skip MariaDB installation (default: False)
--reset	Uninstall and reset MariaDB before reinstall (default: False)

4. To install and configure MariaDB, run the following command:

   Note: The table above lists the parameter values for the script. For the parameters where the Default = None, you must replace the parameter in the following command, which is in brackets <xxx xxx>, with the appropriate values based on your environment. In the <Mariadb User id> parameter, you must enter a MariaDB User ID. This parameter cannot be blank.

   ```
   python3 mariadb_install.py --port <mariadb port>
   --mariadb-data-dir <Data directory> --root-password <root
   password for mariadb> --user <Mariadb User id> --password
   <Mariadb User's Password> --database <name of database to
   be created> --schema-dir <download
   dir/manualinstall/mariaDB/schema> --seed-data-dir
   <download dir/manualinstall/mariaDB/data>
   ```

5. During the execution of the script in the previous step, you may see the following error. Take the necessary steps to resolve this issue. If you did not receive this error, proceed to the next step.

   ```
   https://repo.saltstack.com/yum/redhat/7/x86_64/latest/repo
   data/repomd.xml: [Errno 14] HTTPS Error 404 - Not Found
   ```

Trying other mirror.

To address this issue please refer to the following wiki article: https://wiki.centos.org/yum-errors

If that article does not help to resolve the issue, refer to the following: https://bugs.centos.org/

One of the configured repositories failed (SaltStack Latest Release Channel for RHEL/Centos 7), and yum doesn't have enough cached data to continue. At this point the only safe thing yum can do is fail. There are a few ways to work "fix" this:

a. Contact the upstream for the repository and get them to fix the problem.
b. Reconfigure the baseurl/etc. for the repository, to point to a working upstream. This is most often useful if you are using a newer distribution release than is supported by the repository (and the packages for the previous distribution release still work).
c. Run the command with the repository temporarily disabled yum --disablerepo=salt-latest …
d. Disable the repository permanently, so yum won't use it by default. Yum will then just ignore the repository until you permanently enable it again or use --enablerepo for temporary usage: yum-config-manager --disable salt-latest or subscription-manager repos --disable=salt-latest
e. Configure the failing repository to be skipped, if it is unavailable.

Note that yum will try to contact the repo. When it runs most commands, it will have to try and fail each time and yum will be much slower. If it is a very temporary problem though, this is often a nice compromise:
yum-config-manager --save
--setopt=salt-latest.skip_if_unavailable=true
failure: repodata/repomd.xml from salt-latest: [Errno 256] No more mirrors to try.
https://repo.saltstack.com/yum/redhat/7/x86_64/latest/repodata/repomd.xml: [Errno 14] HTTPS Error 404 - Not Found

6. Provide access to the port for MariaDB using the command below:

   ```
   sudo firewall-cmd --zone=public --add-port=<MariaDB Port>/tcp --permanent
   ```

   ```
   sudo firewall-cmd --reload
   ```

7. The installation and configuration of MariaDB are complete.

Manually Installing MariaDB

1. Create the following `MariaDB.repo` file if it does not exist. If this file exists, skip to the next step to update the file.

   ```
   /etc/yum.repos.d/MariaDB.repo
   ```

2. Specify the version of MariaDB to be installed by adding the following lines to the `MariaDB.repo` file:

   ```
   [mariadb]
   ```

   ```
   name = MariaDB-10.5
   ```

   ```
   baseurl=https://yum.mariadb.org/10.5/centos7-amd64
   ```

   ```
   # alternative:
   baseurl=http://archive.mariadb.org/mariadb-10.5/yum/centos7-amd64
   ```

   ```
   gpgkey=https://yum.mariadb.org/RPM-GPG-KEY-MariaDB
   ```

   ```
   gpgcheck=1
   ```

3. Create a directory to store the MariaDB data and configuration files; for example, /dr01/mariadb/mysql.

   ```
   sudo mkdir -p /dr01/mariadb/mysql
   ```

4. Add user mysql if it does not exist.

   ```
   sudo useradd mysql
   ```

5. Update ownership of the folder: /dr01/mariadb to mysql user.

   ```
   sudo chown -R mysql: /dr01/mariadb
   ```

6. Run the following command to install MariaDB Server and MariaDB Client:

   ```
   sudo yum install MariaDB-server MariaDB-client
   ```

7. The following installation information is displayed. When prompted, enter “y” to confirm.

```
Install 3 Packages (+8 Dependent packages)

Total download size: 53 M

Is this ok [y/d/N]:  y

Retrieving key from
https://yum.mariadb.org/RPM-GPG-KEY-MariaDB

Importing GPG key 0x1BB943DB:

Userid: "MariaDB Package Signing Key
<package-signing-key@mariadb.org>"

Fingerprint: 1993 69e5 404b d5fc 7d2f e43b cbcb 082a 1bb9
43db

From: https://yum.mariadb.org/RPM-GPG-KEY-MariaDB

Is this ok [y/N]: y
```

8. Update `/etc/my.cnf` file for the following configuration:

```
[mysqld]

datadir = <Data Directory for Mariadb e.g- /dr01/mariadb>

port = <MariaDB Port e.g 3306>

character_set_server = utf8

collation_server = utf8_general_ci

innodb_buffer_pool_size = 11G

read_buffer_size = 1G

read_rnd_buffer_size = 1G

sort_buffer_size = 1G

max_connections = 500
```

9. Install the system table by entering:

```
sudo mysql_install_db
```

10. Change the ownership of the file under data directory (for example, `/dr01/mariadb/mysql`) by entering :

```
sudo chown -R mysql: <Data Directory e.g
/dr01/mariadb/mysql>
```

11. Enable MariaDB by entering:

```
sudo systemctl enable mariadb
```

12. Start MariaDB by entering:

```
sudo systemctl start mariadb
```

13. Secure the MariaDB installation by entering the following command:

```
sudo mariadb-secure-installation
```

    a. When prompted for root user password, press Enter (leave blank).

    ```
    Enter current password for root (enter for none):
    ```

    b. Enter the following when prompted:

    ```
    Switch to unix_socket authentication [Y/n] n

    Change the root password? [Y/n] Y

    Enter new password that you want to set for root user

    New password: <enter new root user password>

    Re-enter new password: <re-enter new root user password>

    Remove anonymous users? [Y/n] Y

    Disallow root login remotely? [Y/n] y

    Remove test database and access to it? [Y/n] Y

    Reload privilege tables now? [Y/n] Y
    ```

14. Create “database qad” and then load the schema and data:

    a. Create database qad by entering:

    ```
    mysql -u root -p
    ```

    b. Enter Password: `<enter root password>`

    c. Enter the following:

    ```
    MariaDB [(none)]> create database <database name>;

    MariaDB [(none)]> exit;
    ```

    d. Load the MariaDB schema for the database. The file needs to be manually edited to replace the `qad` value with name of database provided in previous step

(`<database Name>`)
Note: The symbol **(`)** is tick and not a single quote.

```
mysql -u root -p <database name> < <download
Location>/manualinstall/mariaDB/schema/qad.sql
```

The PE Ignition package that was downloaded contains the `schema/qad.sql` file.

e. Enter Password: `<enter root password>`
f. Load the MariaDB seed data for the database:

```
mysql -u root -p <database name> < <download
Location>/manualinstall/mariaDB/data/qad.sql
```

Note: The PE Ignition package that was downloaded contains the `data/qad.sql` file.

g. Enter Password: `<enter root password>`

15. Provide remote access for the user by entering the following:

```
sudo mysql -u root -p

Password:  <enter root password>

MariaDB [(none)]> CREATE USER '<user id>'@'%' IDENTIFIED
BY '<enter user id's password>';

MariaDB [(none)]> GRANT ALL PRIVILEGES ON *.* TO '<user
id>'@'%' WITH GRANT OPTION;

MariaDB [(none)]> CREATE USER '<user id>'@'localhost'
IDENTIFIED BY '<enter user id's password>';

MariaDB [(none)]> GRANT ALL PRIVILEGES ON *.* TO '<user
id>'@'localhost' WITH GRANT OPTION;

MariaDB [(none)]> FLUSH PRIVILEGES;

MariaDB [(none)]> exit;
```

16. Restart the firewall by entering the following yab command. Make sure ports are open to red-hat linux repositories.

```
service firewalld start
```

17. Provide access to the port for MariaDB using the command below:

```
sudo firewall-cmd --zone=public --add-port=<MariaDB
Port>/tcp --permanent
```

```
sudo firewall-cmd --reload
```

18. The installation of MariaDB is complete. Proceed to the next section and install Ignition.

Installing Ignition

1. Download the Ignition installer package v8.0.15:

   https://inductiveautomation.com/downloads/archive/8.0.15

2. Select the Ignition package based on the OS.

   Note: If installing on a Linux server, make sure the downloaded file has execute permissions.

3. Execute the install file. You can log in as root or use the sudo command to install:

   ```
   sudo ./ignition-8.0.15-linux-x64-installer.run
   ```

   a. Specify the username for installing Ignition.

      ```
      Username: <username to install ignition>
      ```

   b. Enter the Ignition Package installation location; for example, `[/usr/local/bin/ignition]: /dr01/ignition`:

      ```
      [/usr/local/bin/ignition]: <directory path of ignition install>
      ```

   c. Specify if the installation type is typical or custom. It is recommended that you select the “1 (Typical)” option:
      - [1] Typical - Includes Ignition with SQL Bridge, Perspective, Vision, OPC-UA, and driver modules for Allen-Bradley, Siemens, and MODBUS devices.
      - [2] Custom - Install additional modules and adjust the default modules to install.

   d. The setup is now ready to begin installing Ignition on your computer. Enter “Y” to begin installation:

      ```
      Ready to Install

      Do you want to continue? [Y/n]: Y
      ```

   e. A confirmation message is displayed when the installation is complete:

      ```
      Ignition Successfully Installed
      ```

   f. Enter “Y” to start Ignition:

      ```
      Start Ignition Now [Y/n]: Y
      ```

g. Enter “Y” to install service:

```
Install Service [Y/n]: Y
```

4. Copy the files from “ignitionserverfiles” directory available in peignition package into the following Ignition webserver location:

   `<Ignition Install Directory>/webserver/webapps/main`

   copy ignitionserverfiles/images into

   `<Ignition Install Directory>/webserver/webapps/main/images folder`

   copy ignitionserverfiles/docs into

   `<Ignition Install Directory>/webserver/webapps/main/docs folder`

   copy ignitionserverfiles/videos into

   `<Ignition Install Directory>/webserver/webapps/main/videos folder`

   copy ignitionserverfiles/sounds into

   `<Ignition Install Directory>/webserver/webapps/main/sounds folder`

5. Copy the following three Kafka scripts, which are available in the package, into `/usr/bin` or any other directory that is in the system PATH:
   - `rdkafka_simple_producer.sh`
   - `rdkafka_simple_producer_ssl`
   - `rdkafka_simple_producer`
6. Set execute permissions to the scripts by entering:

   ```
   sudo chmod +x
   <scripts-location>/rdkafka_simple_producer.sh
   rdkafka_simple_producer_ssl rdkafka_simple_producer
   ```

7. Proceed to the next section to install the Adaptive UX components and Nifi.

Single Server Installation

Installing Adaptive UX Components and Nifi Flow Components

The Adaptive UX and Nifi Flow components are installed using YAB.

YAB is a configuration management tool you can use to start, stop, reconfigure, and maintain the consistency of your Enterprise Edition environments. Your environment is defined in terms of discrete units, or packages, which you can replace and add as needed.

See *QAD Enterprise Edition Installation Guide* for more information about using YAB.

The `peaux-app-.x.x.x.x` package contains the Adaptive UX components. The `penifilocal-x.x.x.x` and `penifiremote-x.x.x.x` packages contain the Nifi Flow Components.

Note: If you are only updating to a new version of the Nifi Flow components, see Updating Nifi Flow Components.

Follow these steps to install the Adaptive UX and the Nifi Flow components:

1. Add the following properties to the `configuration.properties` file and customize its value as per your configuration needs.

   ```
   pe.mariadb.name=<Name of the MariaDB database>

   pe.mariadb.host=<Hostname where MariaDB database is installed>

   pe.mariadb.port=<Port configured for MariaDB>

   pe.mariadb.user=<MariaDB login credentials for user with write access to Production Execution tables>

   pe.mariadb.password=<MariaDB login credentials for user with write access to Production Execution tables>

   pe.integration.user=<AUX login credentials for user that is used by Nifi to login to AUX and invoke APIs>

   pe.integration.password=<AUX login credentials for user that is used by Nifi to login to AUX and invoke APIs>

   pe.qms.images.targetdirectory=<Target Directory for storing QMS artifacts>

   pe.qms.images.sourcedirectory=<Source Directory for getting QMS artifacts>
   ```

2. To install both packages via the Cloud catalog, enter the following yab command:

   ```
   yab install peaux-app-x.x.x.x penifilocal-x.x.x.x penifiremote-x.x.x.x
   ```

3. Provide access to Restapi port, if blocked by firewall.

a. To find the rest api port, enter the following command:

```
yab config pe.restapi.port
```

b. Enter the following command to provide access to Rest API port, if it is blocked by firewall:

```
<Rest API Port> is derived in earlier step

sudo firewall-cmd --zone=public
--add-port=<Rest API Port>/tcp --permanent

sudo firewall-cmd --reload
```

Multiple Server Installation

Installing Adaptive UX Components on AUX server

The Adaptive UX component is installed using YAB.

YAB is a configuration management tool you can use to start, stop, reconfigure, and maintain the consistency of your Enterprise Edition environments. Your environment is defined in terms of discrete units, or packages, which you can replace and add as needed.

See *QAD Enterprise Edition Installation Guide* for more information about using YAB.

The `peaux-app-.x.x.x.x package` contains the Adaptive UX components.

Follow these steps to install the Adaptive UX components:

1. Add the following properties to the `configuration.properties` file and customize its value as per your configuration needs.
2. To install the package via the Cloud catalog, enter the following yab command:

   ```
   yab install peaux-app-x.x.x.x
   ```

Installing Nifi Remote Components on Standalone Server

Follow these steps to install Nifi remote components on a standalone server using yab:

1. Download the file `yab/config/docs/penifiremote-standalone.properties` from the `penifiremote-x.x.x.x` package.
2. Update the file with the correct yab properties value, some of the yab properties value needs to come from AUX server.

3. The `yab/config/docs/penifiremote-standalone.properties` file provides details on which properties need to be updated and from where the value should come.
4. Run the following yab command:
   `-p:penifiremote-standalone.properties -a:penifiremote-env create`.

Installing Nifi Local Components on Standalone Server/Ignition and Mariadb Server

Follow these steps to install Nifi Local Components on a Standalone Server/Ignition and Mariadb Server using yab:

1. Download the file `yab/config/docs/penifilocal-standalone.properties` file from `penifilocal-x.x.x.x package`.
2. Update the file with the correct yab properties value. Some of the yab property values need to come from the AUX server.
3. The `yab/config/docs/penifilocal-standalone.properties` file provides details on which properties need to be updated and from where value should come.
4. Run the following yab command:
   `-p:penifilocal-standalone.properties -a:penifilocal-env create`.
5. Provide access to Restapi port, if blocked by firewall.
   a. To find the rest api port, enter the following command:

      ```
      yab config pe.restapi.port
      ```

   b. Enter the following command to provide access to Rest API port, if it is blocked by firewall:

      ```
      <Rest API Port> is derived in earlier step

      sudo firewall-cmd --zone=public
      --add-port=<Rest API Port>/tcp --permanent

      sudo firewall-cmd --reload
      ```

Configure Ignition Gateway

This section describes the steps required to configure the Ignition Gateway after it has been installed. During this configuration procedure you will:

- Create the Admin user and set the password.
- Update the port numbers used to access Ignition Gateway.
- Import project files, which contain PE functionality.
- Create a database connection for MariaDB.
- Create a new User Source, which connects the user roles provided with MariaDB with the Ignition project.
- Create a new Realtime Tag Provider, which provides additional functionality for PE.
- Link project files with User Source and Tags.

Follow these steps to configure the Ignition Gateway:

1. Check if the Ignition Gateway is started by entering:

   ```
   sudo <Ignition installation directory>/ignition.sh status
   ```

   Note: If the Ignition Gateway is stopped, enter the following command to start it:

   ```
   sudo <Ignition installation directory>/ignition.sh start
   ```

2. Ignition Gateway is configured from a web browser. You will need the Ignition Gateway port, which is located in the following file:

   ```
   <Ignition installation directory>/data/gateway.xml.
   ```

   For new installations, the default port is 8088:

   ```
   <entry key="gateway.port">8088</entry>
   ```

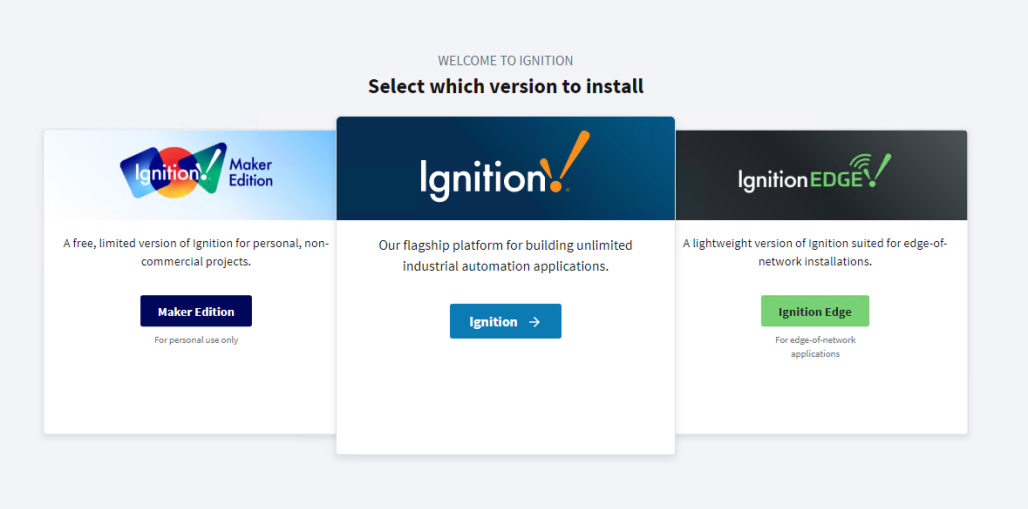
3. Access the Ignition Gateway from a web browser by entering:

   ```
   http://<hostname>:<port>
   ```

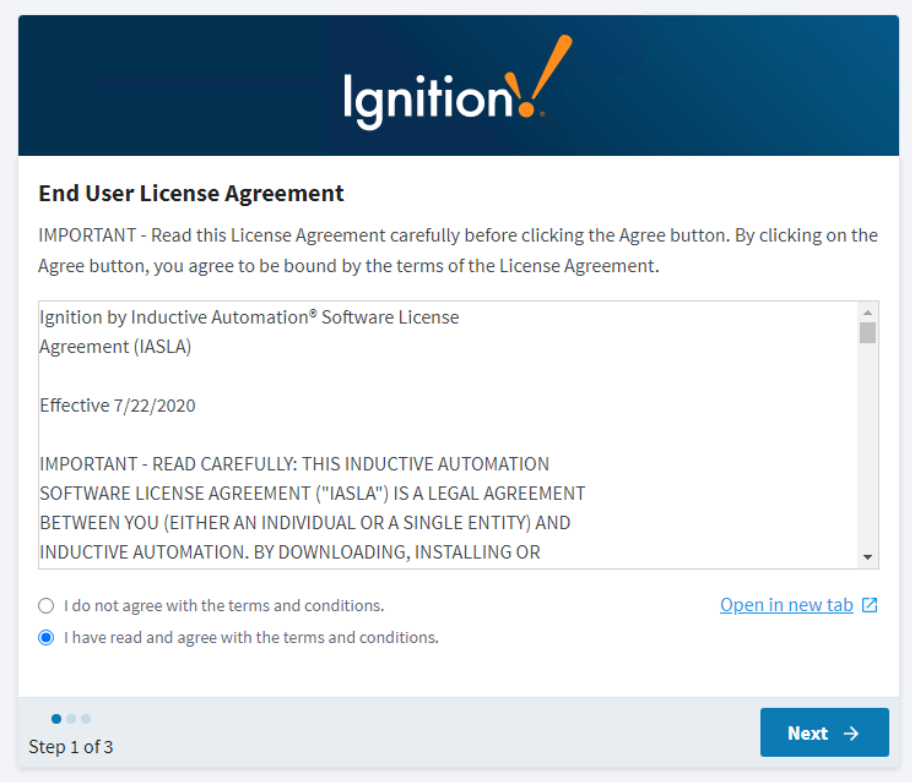
   Note: If it is not accessible, the port needs to be updated by entering:

   ```
   sudo  <Ignition installation directory>/gwcmd.sh -k <new port>
   ```

4. On the welcome page, select Ignition.

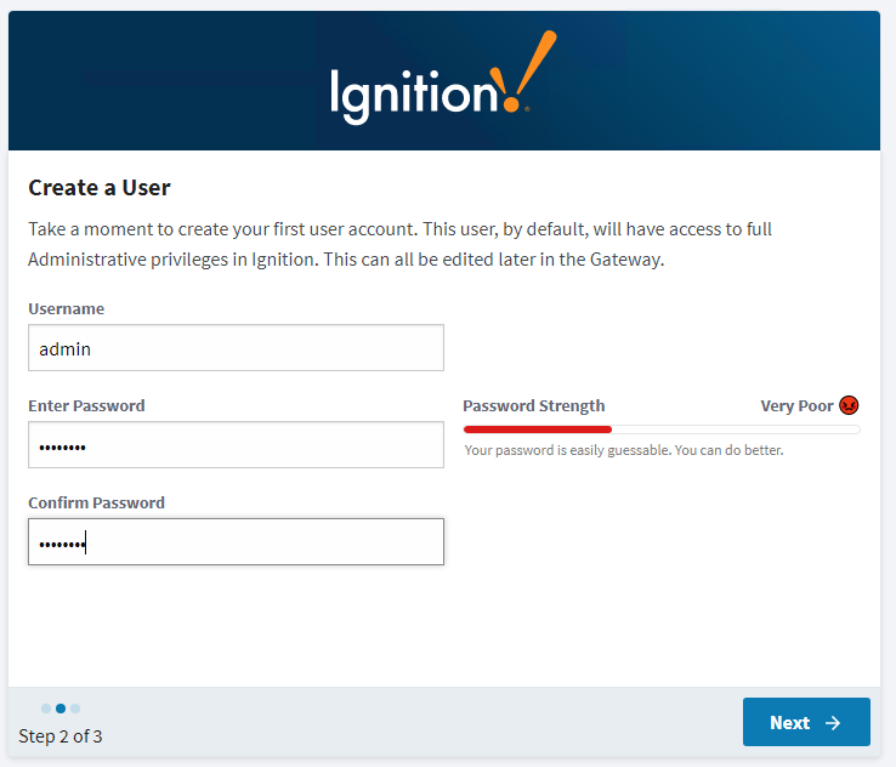


5. Navigate through the terms and conditions page.

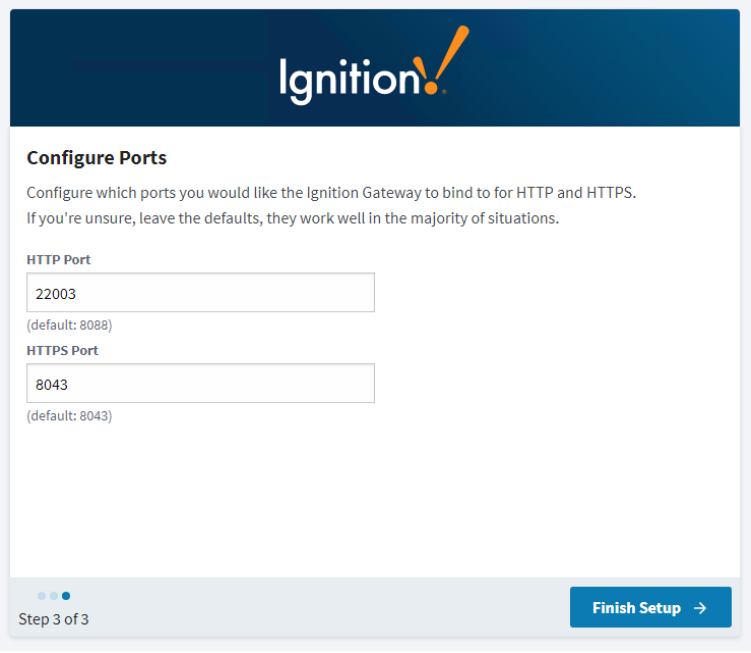


6. Create a user in Ignition for the Admin role and then set the password.

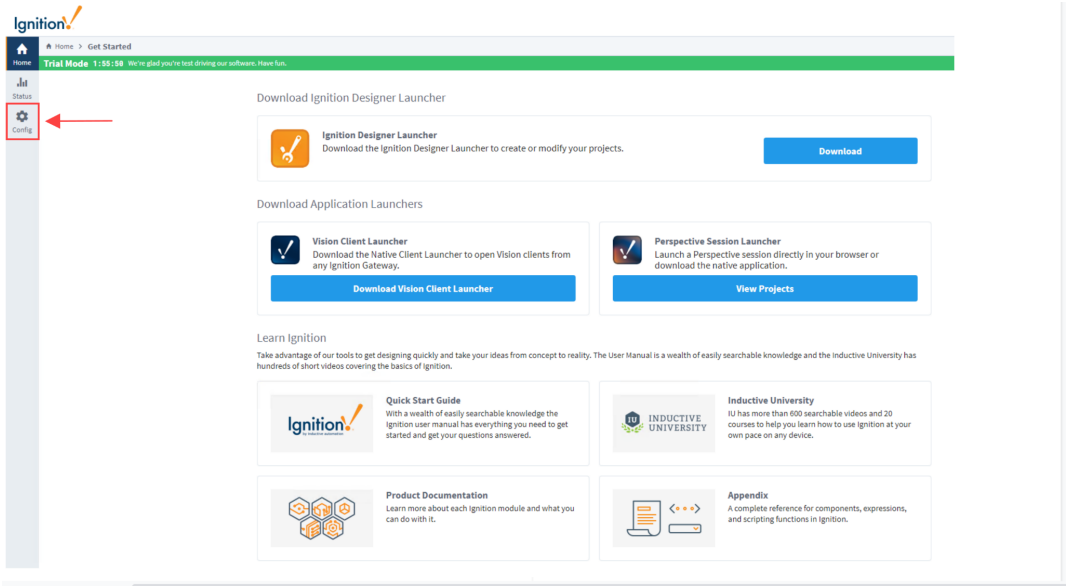
   Note: Make a record of the username and password because these credentials are used each time you access the Ignition Gateway.



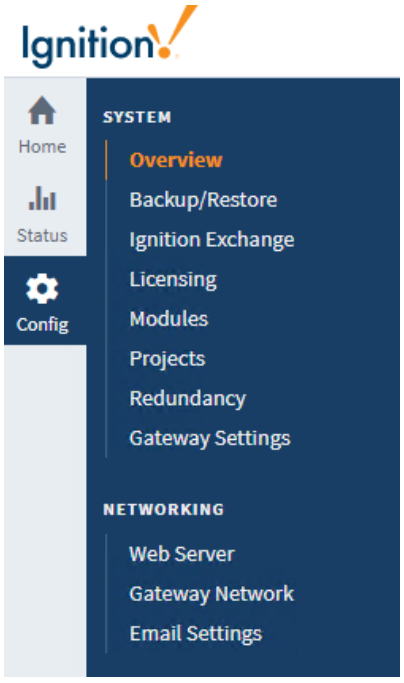
7. Update the port numbers for HTTP Port and HTTPS Port access. The port numbers that are entered are used to access the Ignition Gateway using the URL.

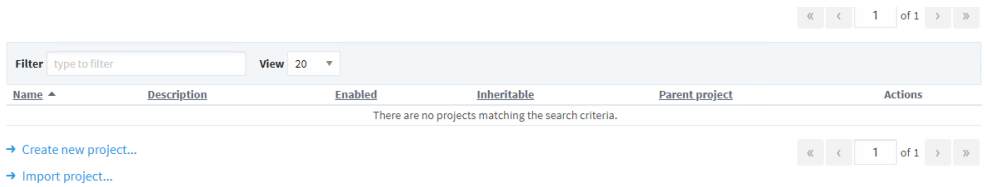


8. Next, import the Project `.zip` file and connect MariaDB.
9. Select the Config menu icon on the Ignition Gateway Home Page.



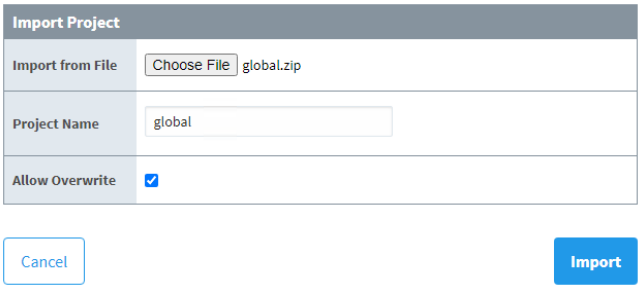
10. Select Projects from the Config menu. The Projects option allows users to import projects into the environment.



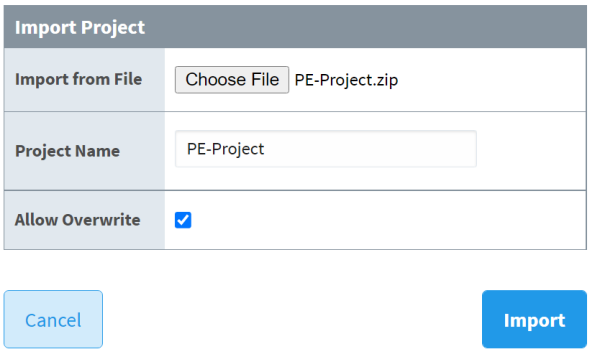


11. Import the `global.zip` Project file by selecting Choose File and then locating the file. The Project file is provided in the PE Ignition installation package.

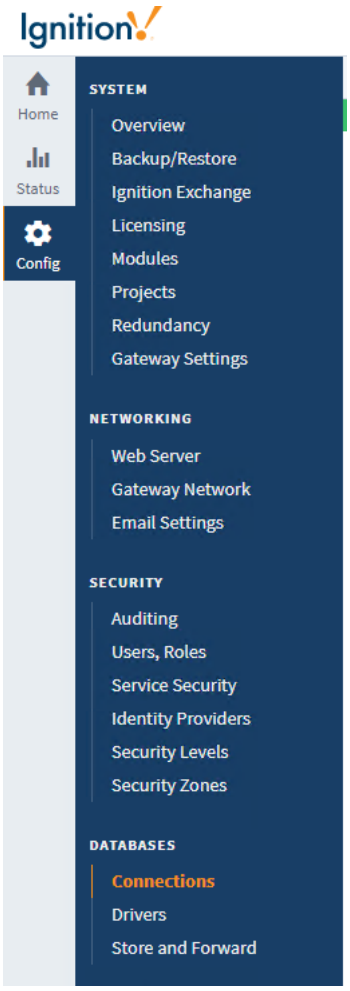
12. In the Project Name field, enter “global” and then select the Allow Overwrite check box.
13. Select Import.



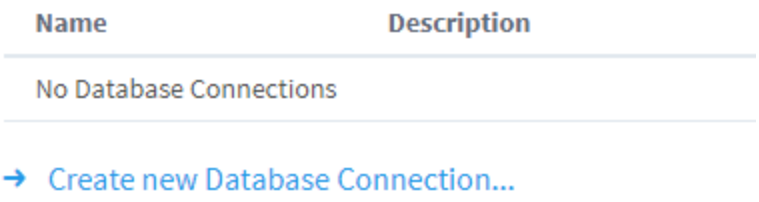
14. Import the `PE-Project.zip` Project file, which is also provided in the PE Ignition installation package.
15. In the Project Name field, enter “PE-Project” and select the Allow Overwrite check box.
16. Select Import.

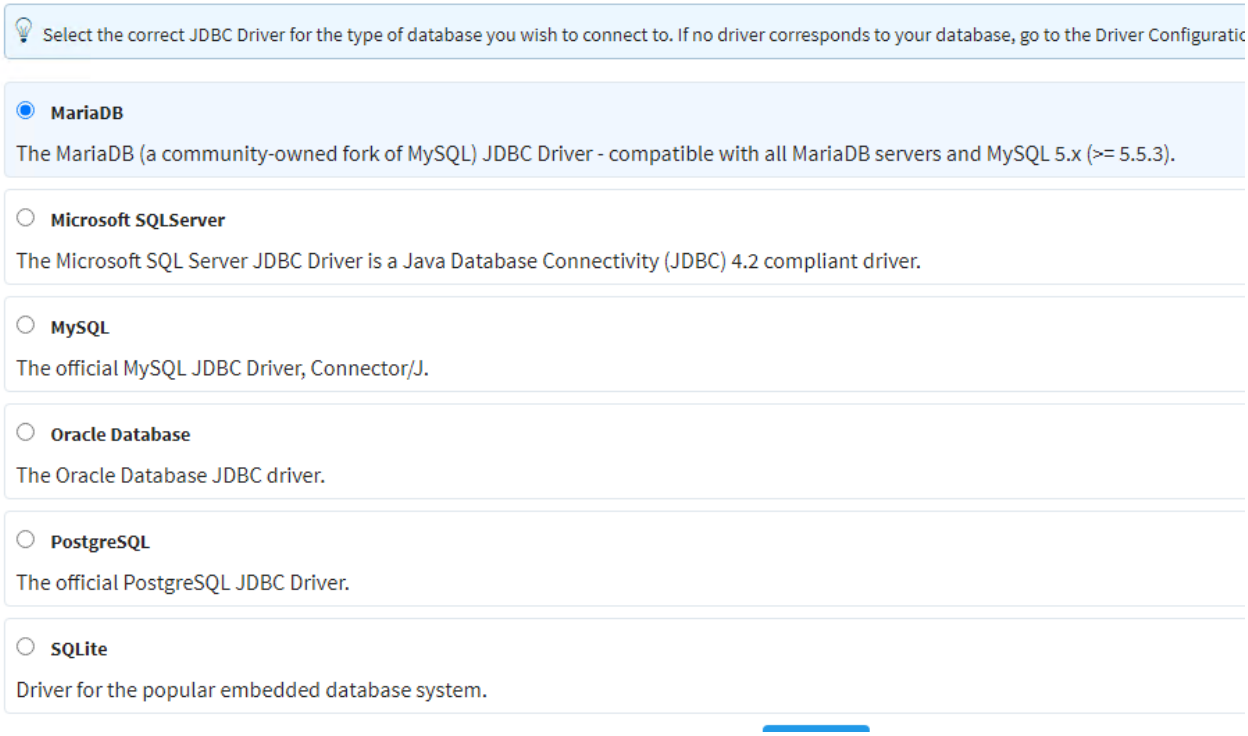


17. Create a new database connection for MariaDB. From the Config menu, select Databases > Connections.



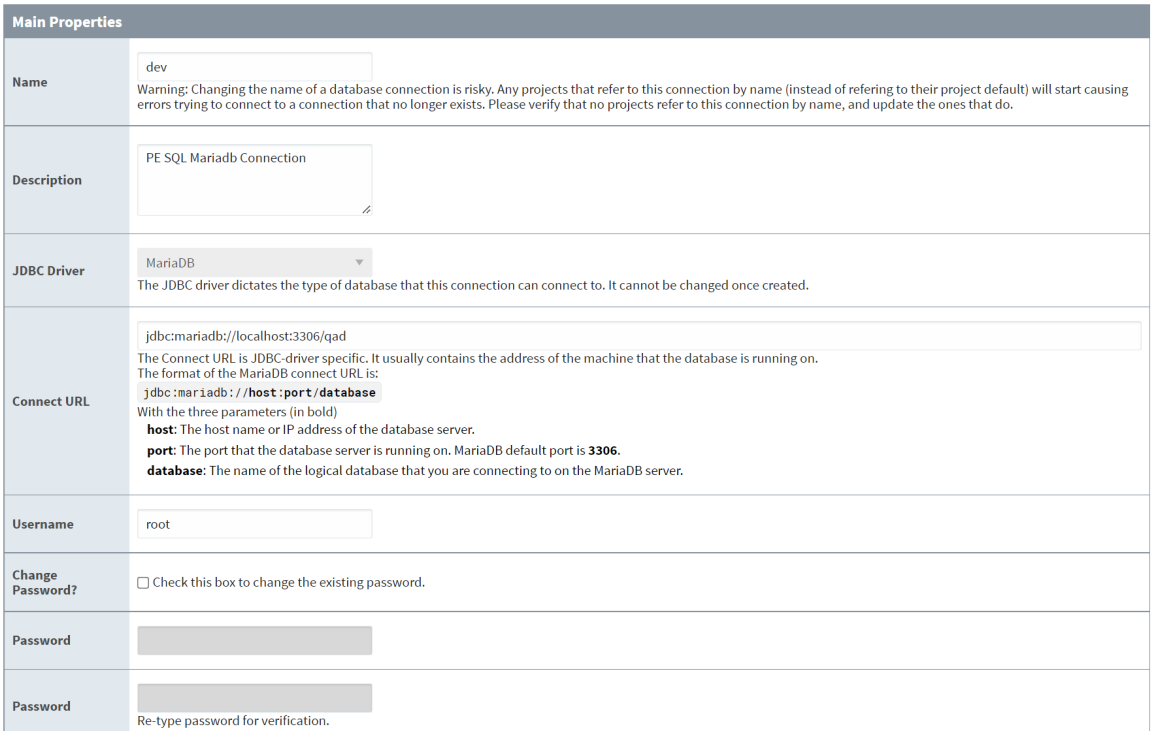
18. Select Create new Database Connection and then select the MariaDB database connection.



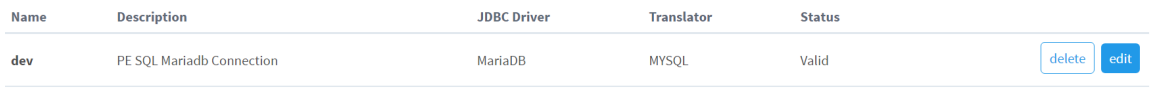


19. Enter the Connection details for MariaDB:

Field	Setting
Name	dev
Description	PE SQL MariaDB Connection
JDBC Driver	MariaDB
Connect URL	Make sure the port and database name are specified correctly.
Username	<username for mariaDB database>
Password	<mariaDB password for username>

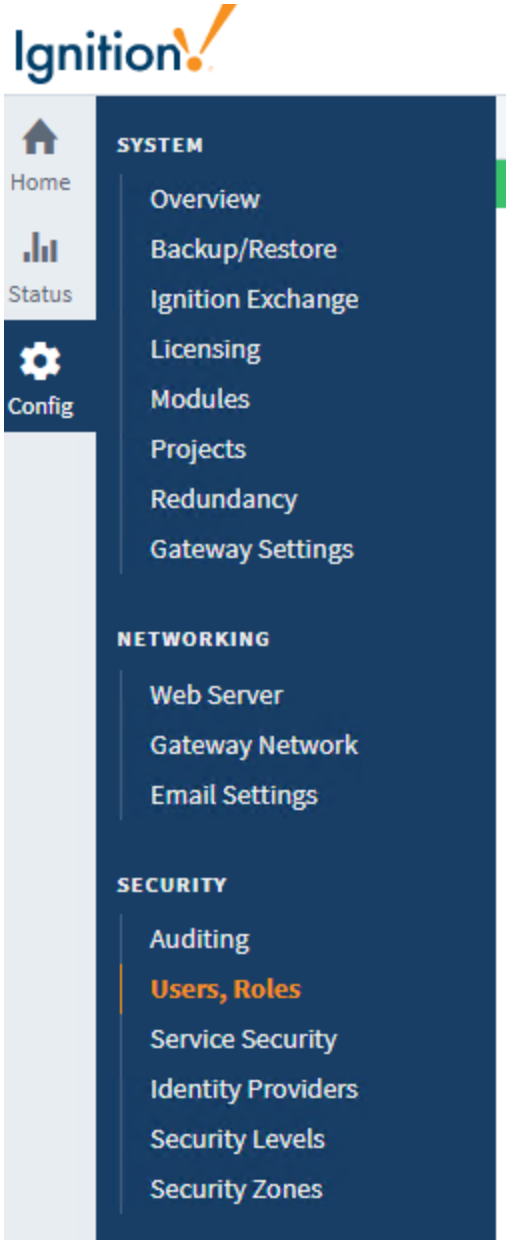


20. After the connection details are saved, the connection Status should display as Valid.

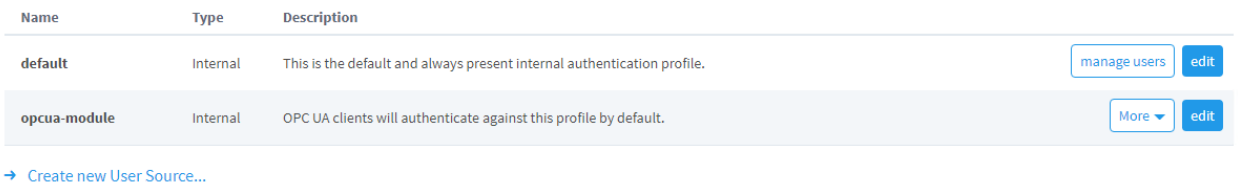


Create Internal User Source

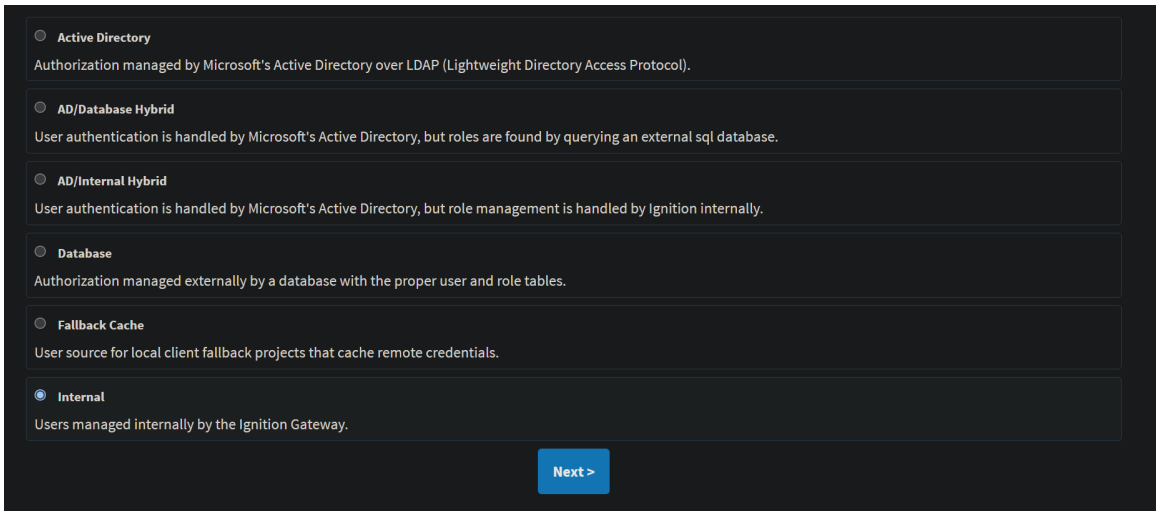
21. Next, create a New Internal User Source by selecting Security > Users, Roles in the Config menu.



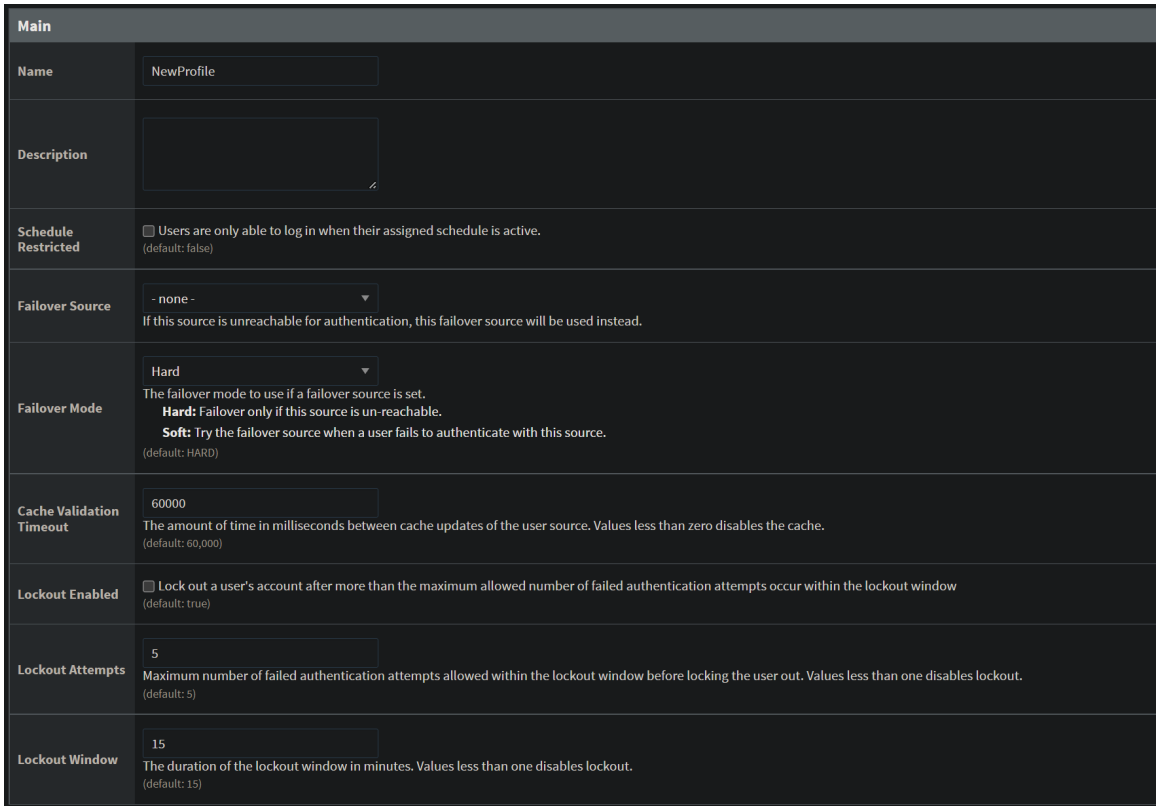
22. Select Create new User Source.

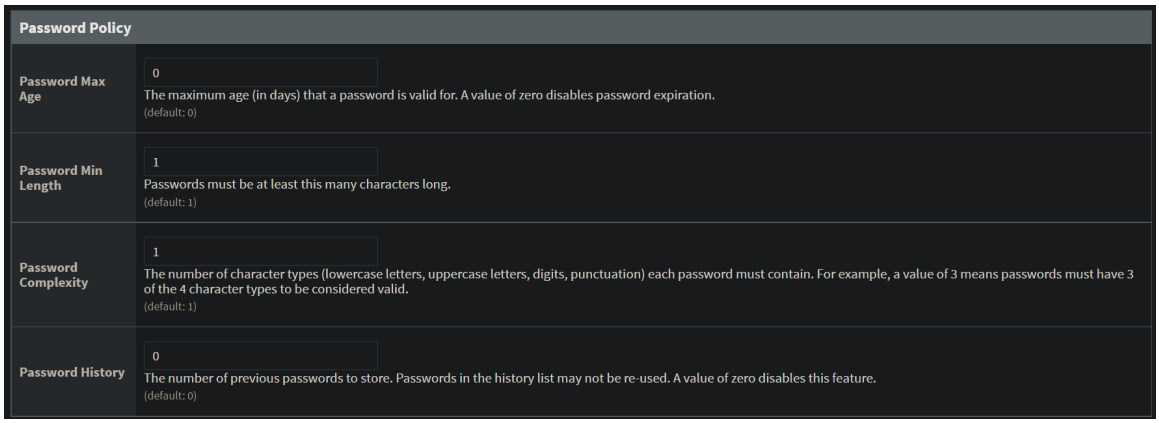


23. Select Internal as the new User Source.



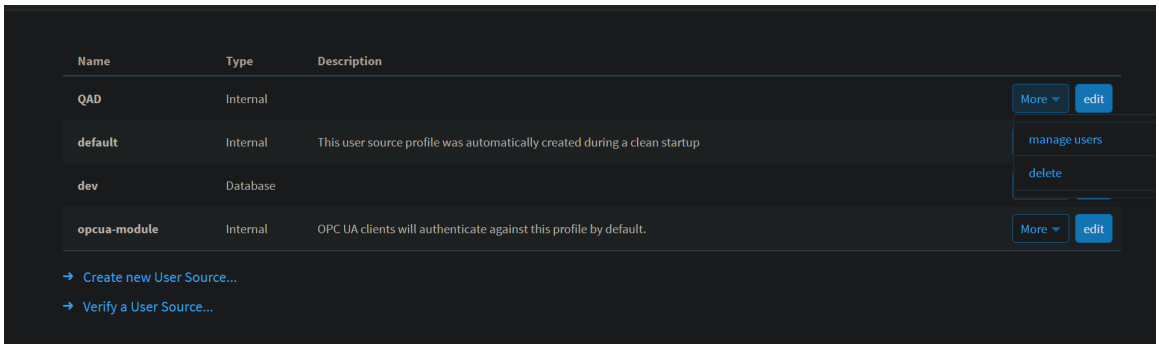
24. Enter the details for the new User Source. In the Name field, enter “QAD”.



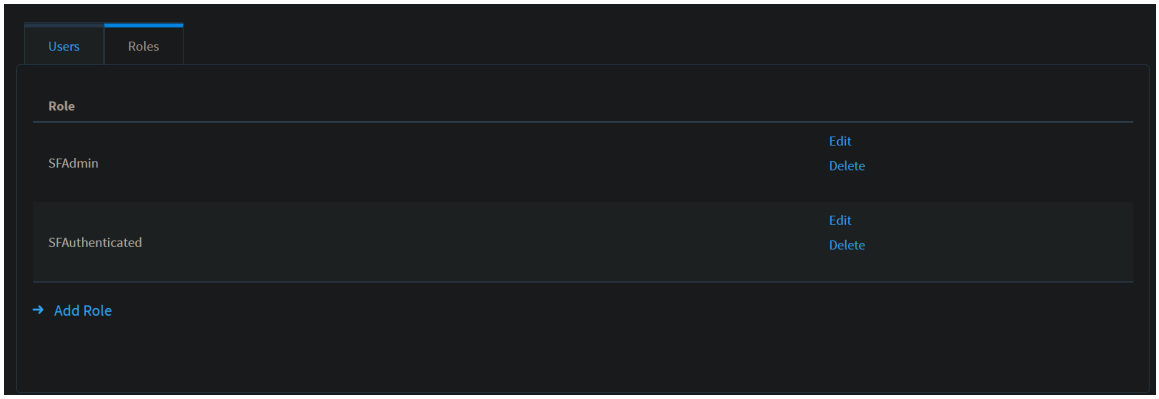


25. After the User Source is saved, a confirmation message is displayed.

26. For the QAD user source, click More and then select Manage Users.

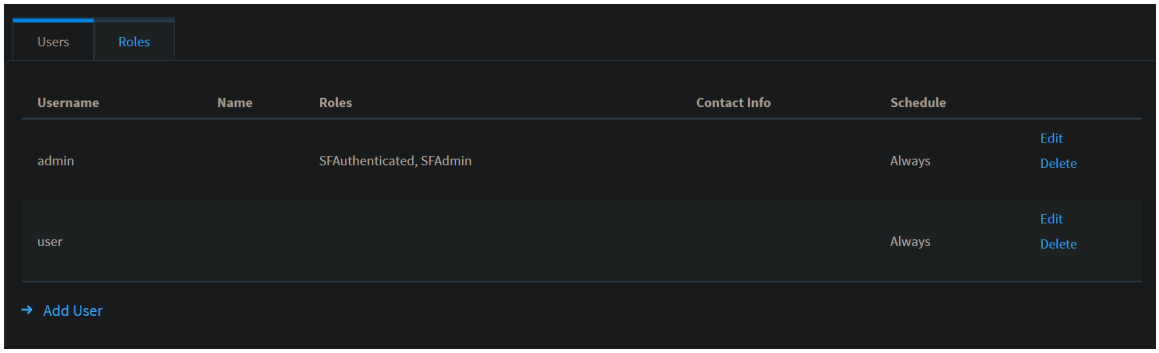


27. Select the Roles tab and create the following two roles:
    - SFAuthenticated
    - SFAdmin



28. Once both roles are created, select the Users tab.
29. Click the Add User button to create two new users with the following properties:

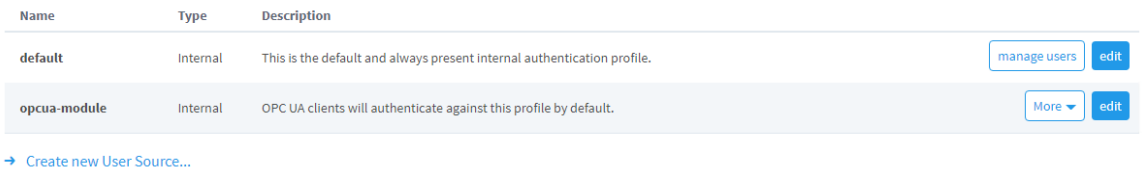
Field	User 1	User 2
Username	user	admin
Password	user	password
Roles	None	SFAdmin, SFAuthenticated



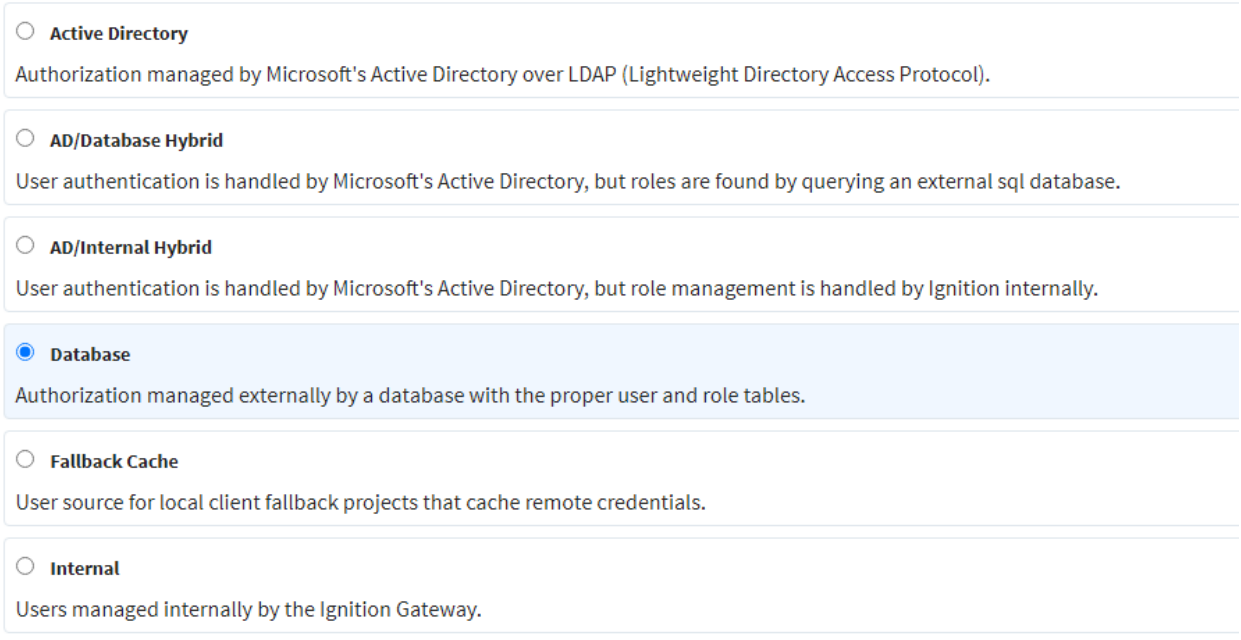
Create Database User Source

30. Next, create a New Database User Source by selecting Security > Users, Roles in the Config menu.

31. Select Create new User Source.



32. Selecting Database as the new User Source.



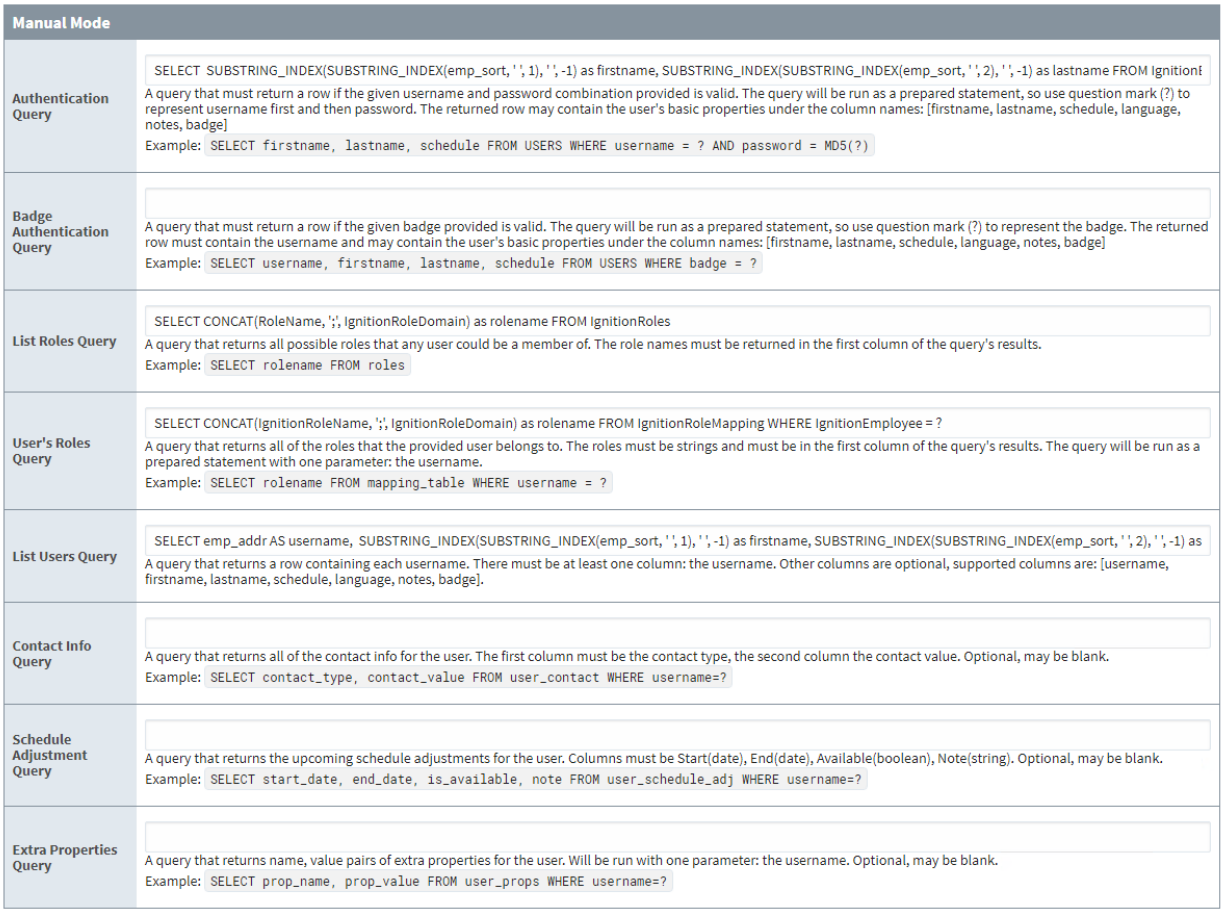
33. Enter the details for the new User Source:

Field	Setting
Authentication Query	SELECT SUBSTRING_INDEX(SUBSTRING_INDEX(emp_sort, ' ', 1), ' ', -1) as firstname, SUBSTRING_INDEX(SUBSTRING_INDEX(emp_sort, ' ', 2), ' ', -1) as lastname FROM IgnitionEmployee emp WHERE emp_addr = ? AND IgnitionPassword = CONVERT(PBKDF2('SHA512', ?, (SELECT IgnitionSalt FROM IgnitionEmployee ie WHERE emp.emp_addr = ie.emp_addr AND emp.emp_domain=ie.emp_domain LIMIT 1), 250, 128, false) USING utf8)
List Role Query	SELECT CONCAT(RoleName, ';', IgnitionRoleDomain) as rolename FROM IgnitionRoles UNION SELECT RoleName as rolename FROM IgnitionRoles WHERE rolename = 'SFAuthenticated'
User's Roles Query	SELECT rolename FROM (SELECT CONCAT(IgnitionRoleName,';',IgnitionRoleDomain) as rolename, IgnitionEmployee FROM IgnitionRoleMapping UNION SELECT IgnitionRoleName as rolename, IgnitionEmployee FROM IgnitionRoleMapping WHERE IgnitionRoleName = 'SFAuthenticated') sub WHERE IgnitionEmployee = ?
List Users Query	SELECT emp_addr AS username, SUBSTRING_INDEX(SUBSTRING_INDEX(emp_sort, ' ', 1), ' ', -1) as firstname, SUBSTRING_INDEX(SUBSTRING_INDEX(emp_sort, ' ', 2), ' ', -1) as lastname FROM IgnitionEmployee
Failover Mode	Soft
Failover Source	QAD

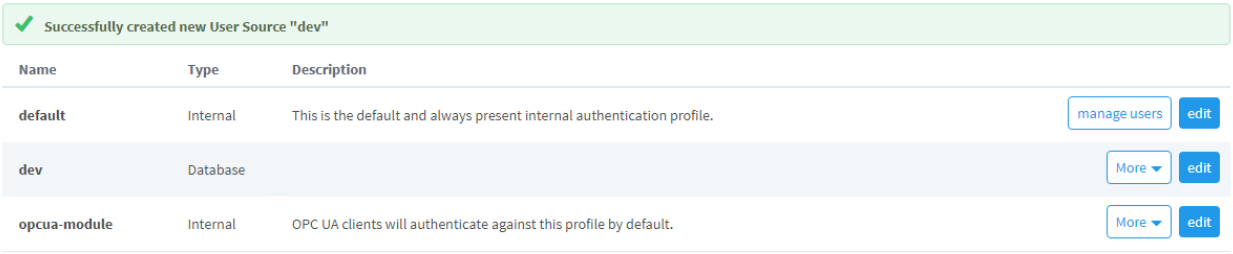
Main	
Name	dev
Description	
Schedule Restricted	☐ Users are only able to log in when their assigned schedule is active. (default: false)
Failover Source	- none - If this source is unreachable for authentication, this failover source will be used instead.
Failover Mode	Hard The failover mode to use if a failover source is set. **Hard:** Failover only if this source is un-reachable. **Soft:** Try the failover source when a user fails to authenticate with this source. (default: HARD)
Cache Validation Timeout	60000 The amount of time in milliseconds between cache updates of the user source. Values less than zero disables the cache. (default: 60,000)
Lockout Enabled	☐ Lock out a user's account after more than the maximum allowed number of failed authentication attempts occur within the lockout window (default: true)
Lockout Attempts	5 Maximum number of failed authentication attempts allowed within the lockout window before locking the user out. Values less than one disables lockout. (default: 5)
Lockout Window	15 The duration of the lockout window in minutes. Values less than one disables lockout. (default: 15)

Main Properties	
Database	dev Choose the database connection this user source will use.
Mode	Manual In Automatic mode, tables will be created for you and all interaction with the database is handled automatically. In Manual mode, you write queries by hand against tables that you've created. Management of users (adding, removing etc) is not supported in manual mode. (default: Automatic)

Automatic Mode	
Tablename Prefix	scada_ When in automatic mode, tables will be created to store the users and roles. They will have this prefix. (default: scada_)

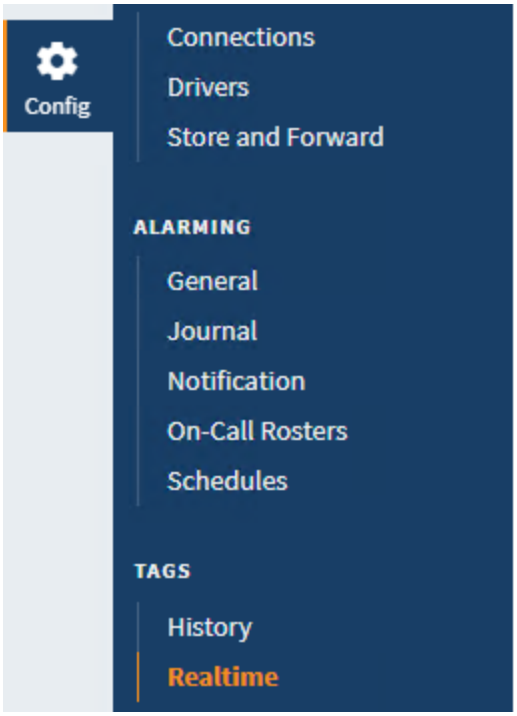


34. After the User Source is saved, a confirmation message is displayed.

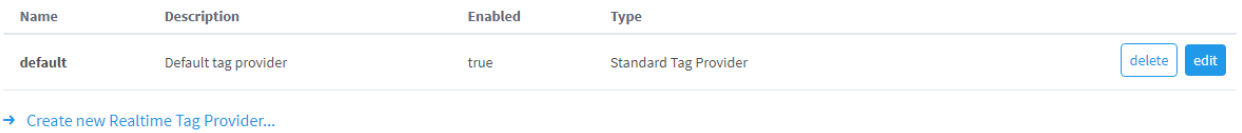


Create a Realtime Tag Provider

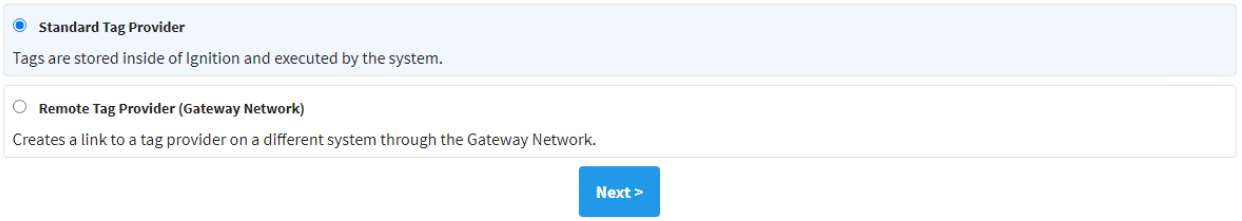
35. Create a new Realtime Tag Provider by selecting Tags > Realtime from the Config menu.



36. Select Create new Realtime Tag Provider.

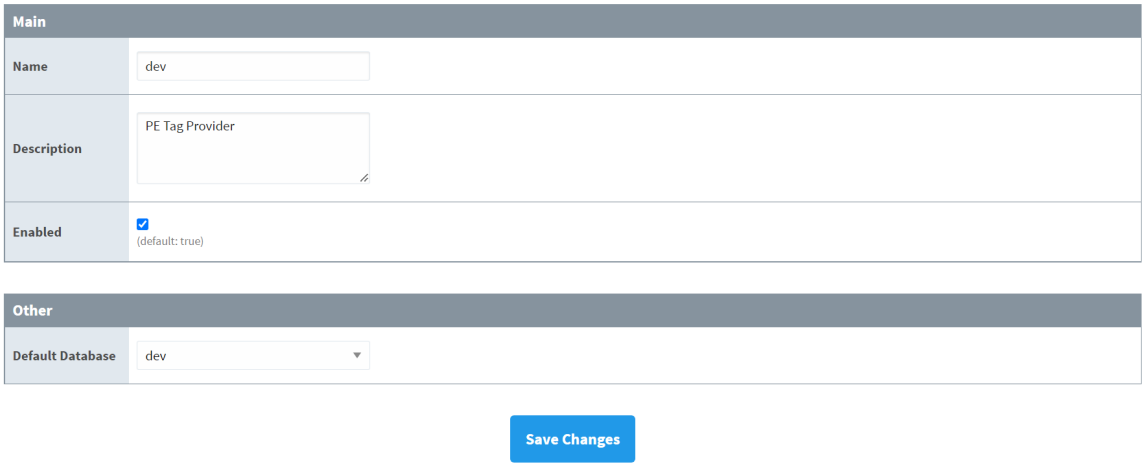


37. Select Standard Tag provider.

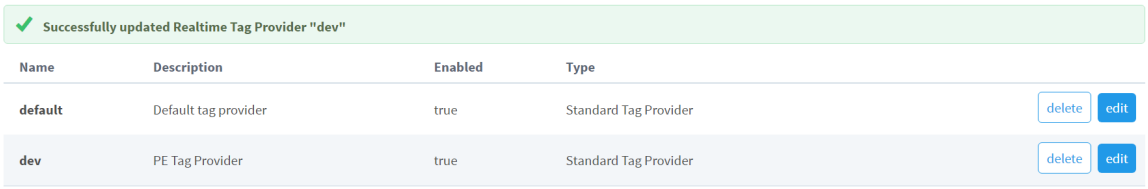


38. Enter the Realtime Tag Provider information:

Field	Setting
Name	dev
Description	PE Tag Provide
Enabled	Yes/Check box selected
Default Database	dev

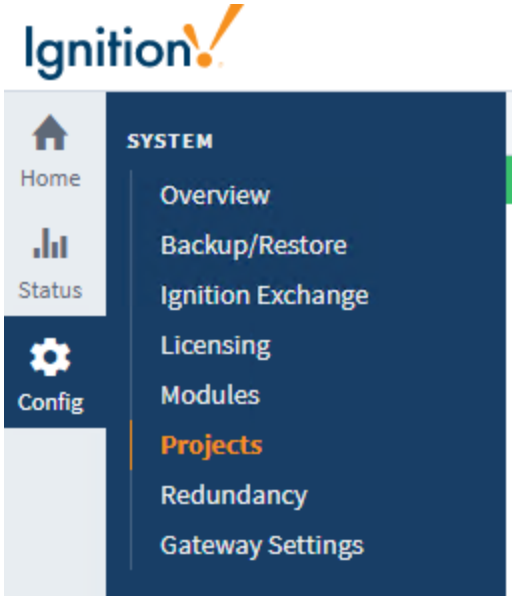


39. After the Realtime Tag Provider is saved, a confirmation message is displayed.

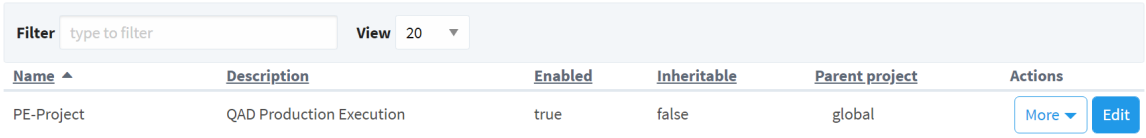


Set Up Project to Use the pesql Database and peTag

40. Next, edit the Project so that it can use the pesql database and peTag. Select Projects from the Config menu.

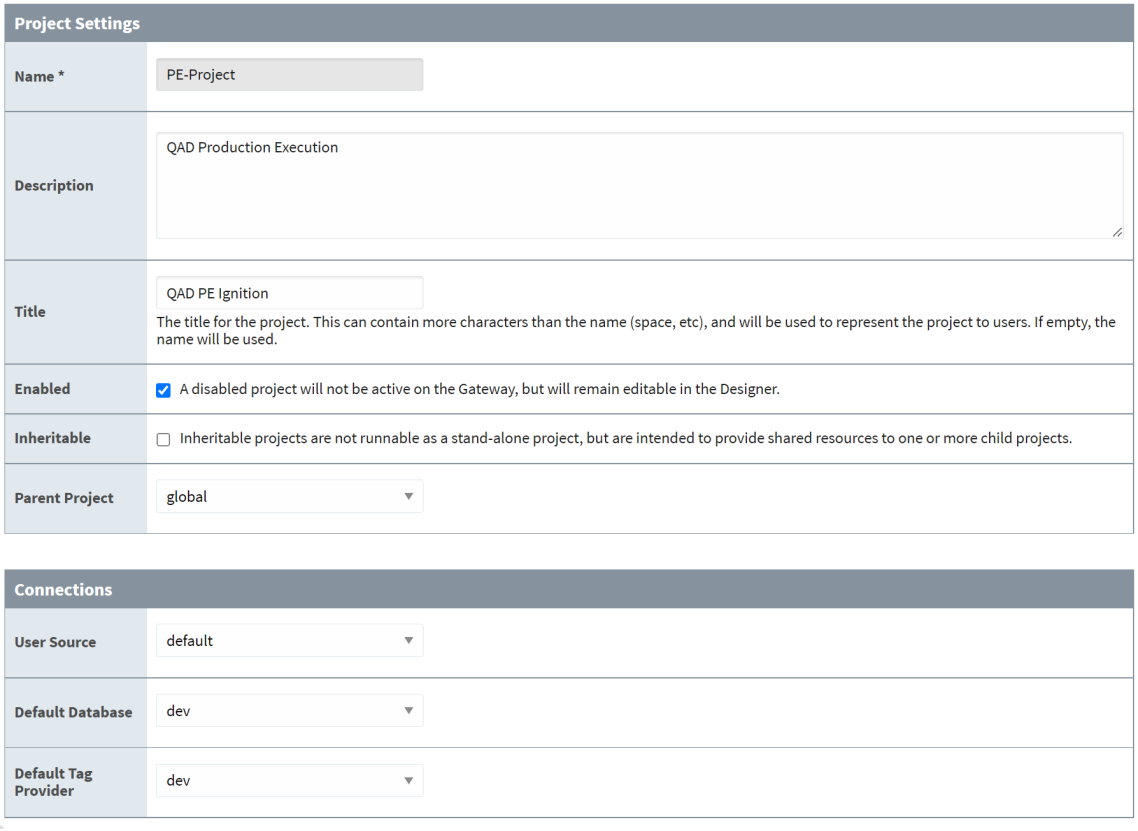


41. For the PE-Project, select Edit.



42. Enter the following information:

Field	Setting
User Source	default
Default Database	dev
Default Tag Provider	dev



Install and Configure the Ignition Designer

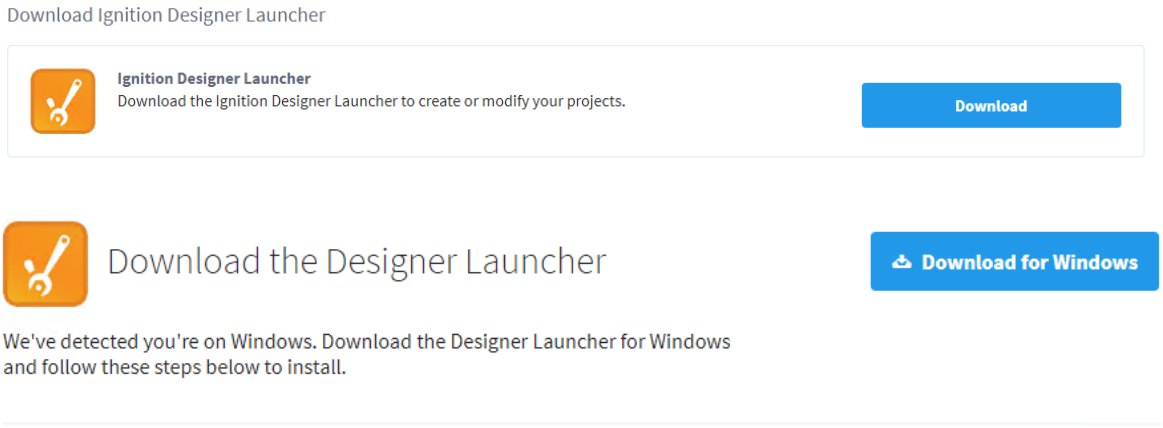
This section contains instructions to configure the Ignition Designer.

In this procedure, you will,

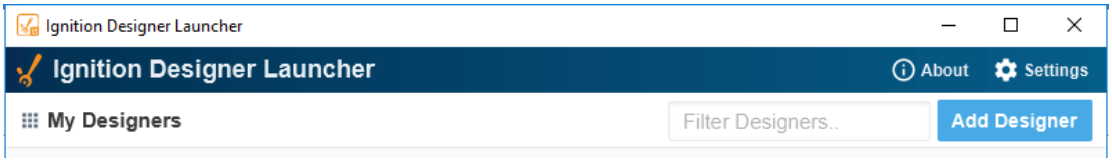
- Download and install Ignition Designer.

- Import Tags and Tag Groups, which provide additional functionality for PE.
- Import the images/icons that are displayed on the Production Execution screens.

1. Download the `DesignerLauncherSetup.exe` file from the Ignition Gateway homepage. Open the `.exe` file and follow the install instructions on the screens.



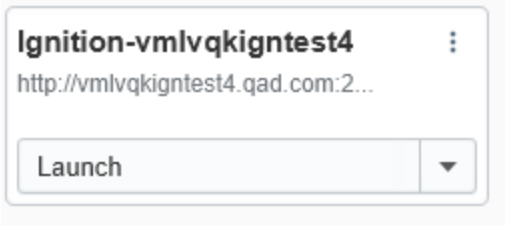
2. Once the installation is complete, open the Ignition Designer and select Add Designer.



3. Select Manually Add Gateway.

4. Enter the Gateway URL: `http://<hostname>:<gateway port>` and then select Add Gateway.

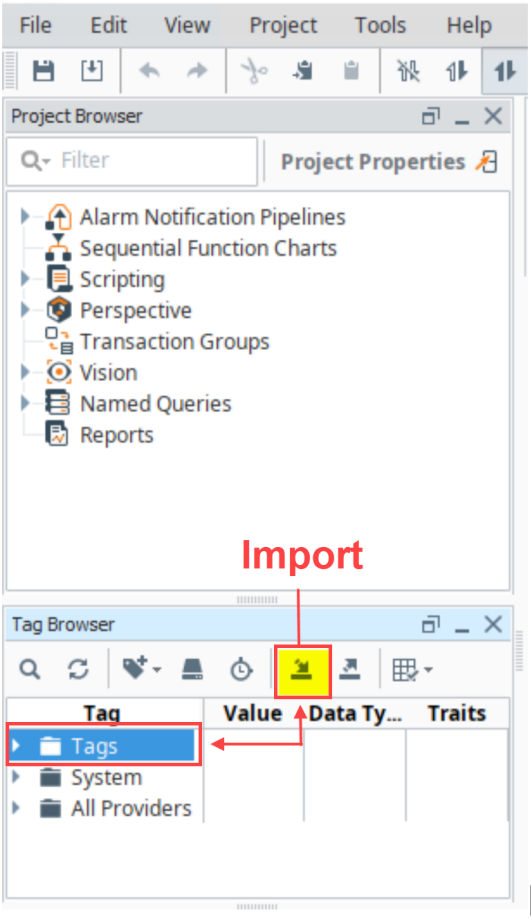
5. Select Launch.



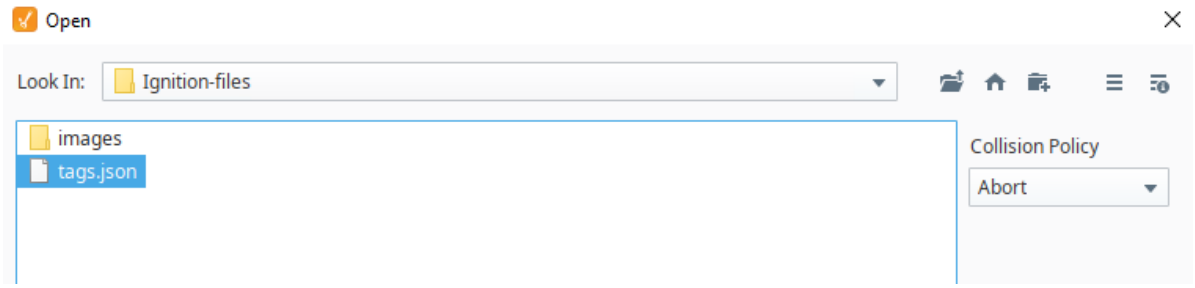
6. Enter the Ignition Gateway user credentials.

7. Select the `PE-Project` and then select Open.

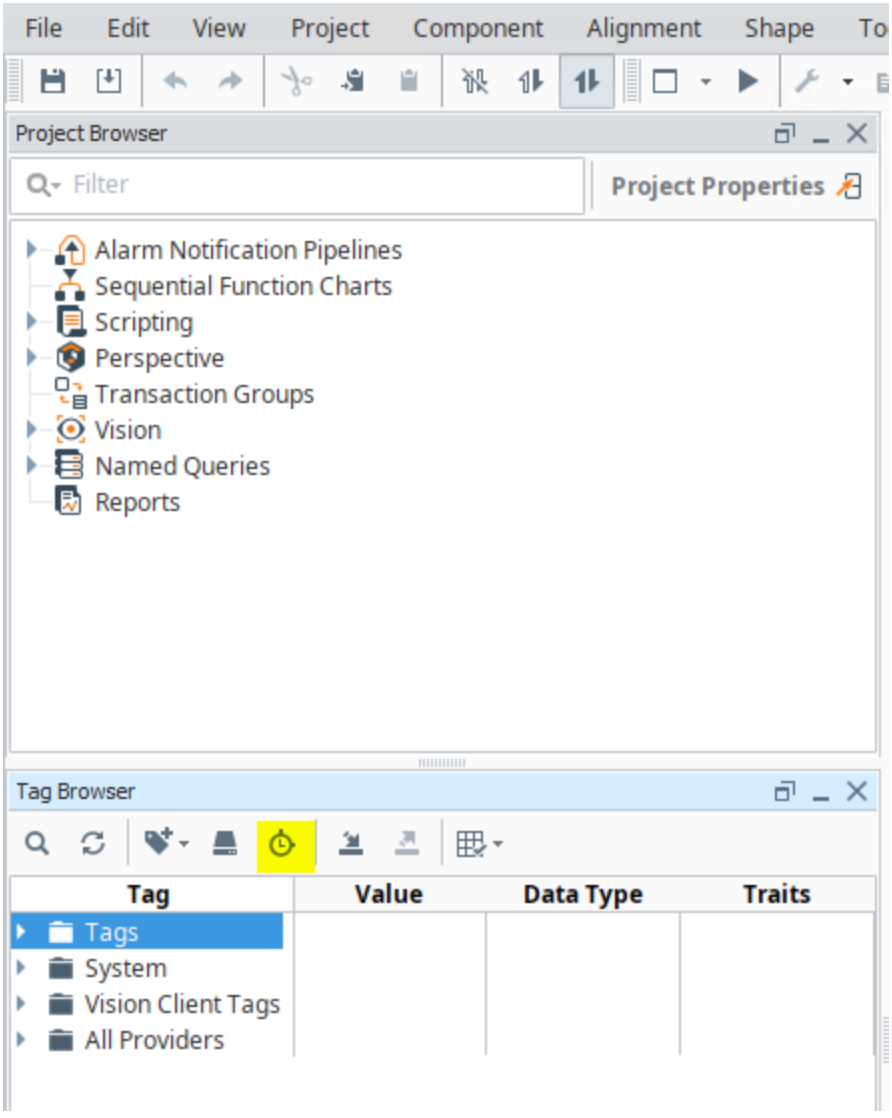
8. Import the Tags by highlighting Tag and selecting Import.



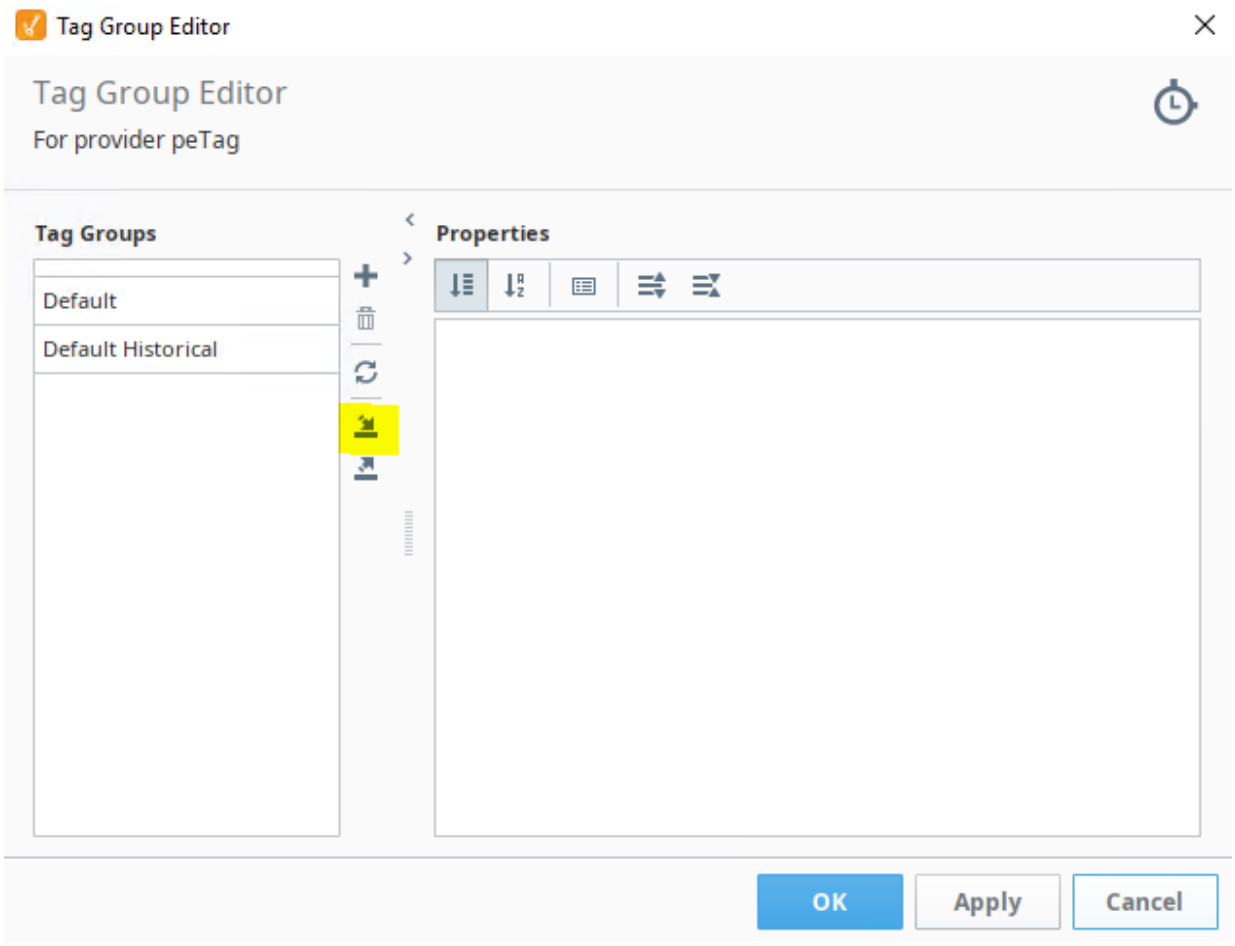
9. Select the `tags.json` file to import. This file is available in the PE Ignition package.



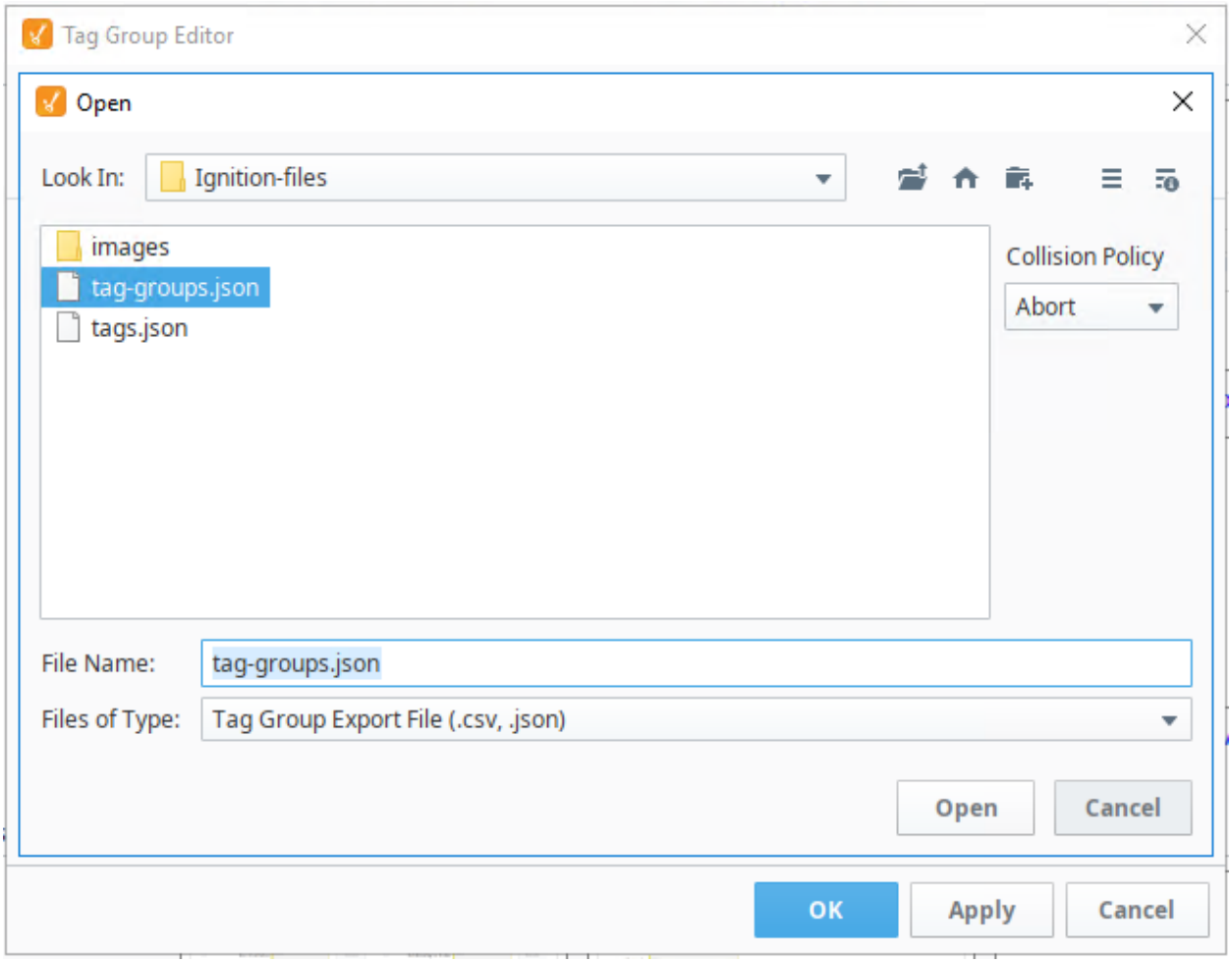
10. Select Tags and then the Edit Tag Group.



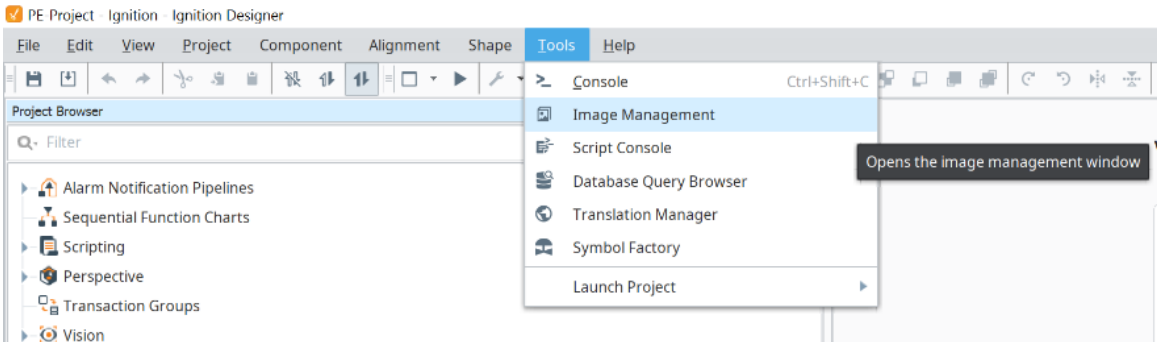
11. From the Tag Group Editor Dialog box, select Import.



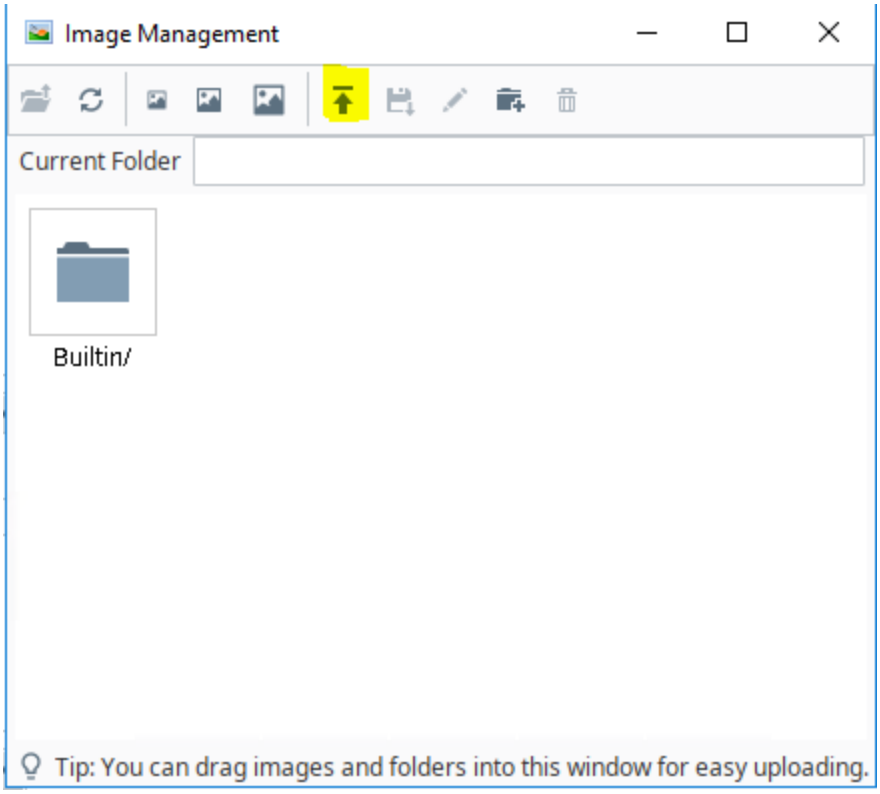
12. Select the location where the `tag-groups.json` file is saved. This file is available in the PE Ignition package.



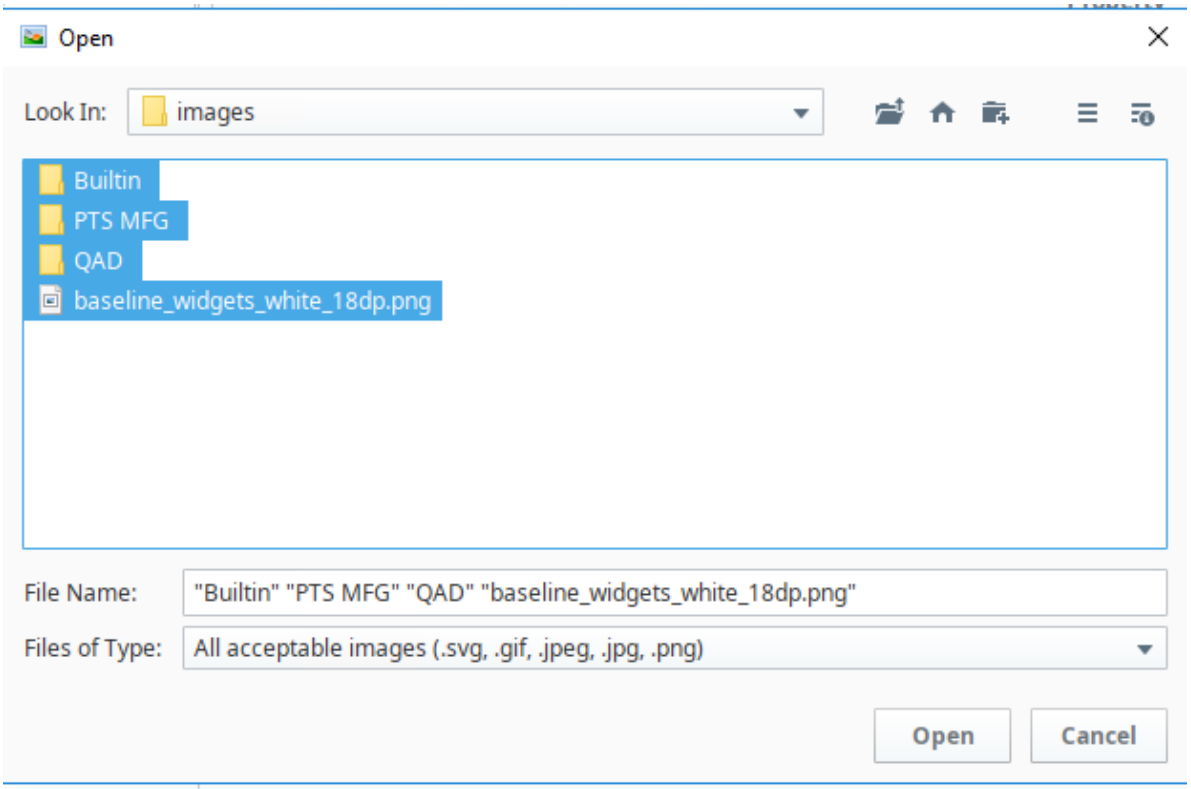
13. Next, import `images/icons` by extracting the `image.zip` file from the package into a temporary location. Select Tools from the menu and then select Image Management.



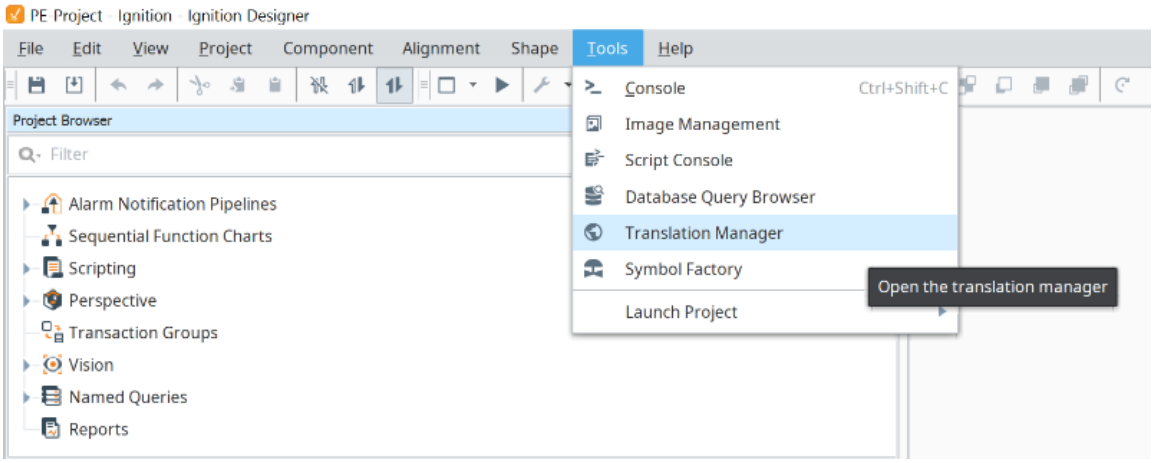
14. In the Image Management window, select Upload new image.



15. From the temporary location where the unzipped images are located, select all files and folders and then select Open. The `Images.zip` file is provided with the PE Ignition package.

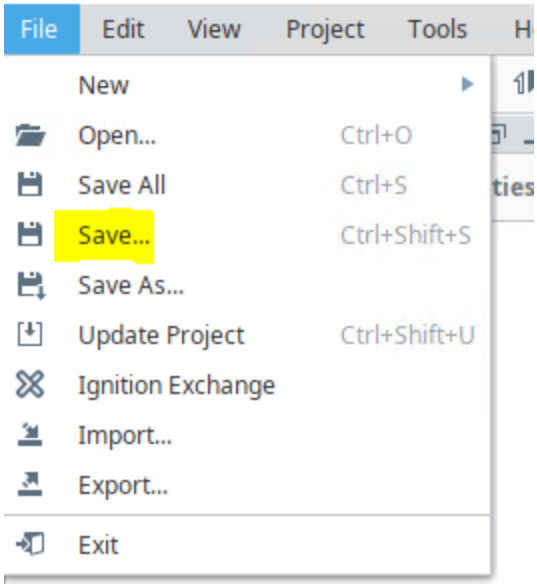


16. It may take a few minutes for Ignition to import the images. Once the import is completed, close the Image Management window.
17. To load the translation for different languages, open Ignition Designer and navigate to Tools → Translation Manager.

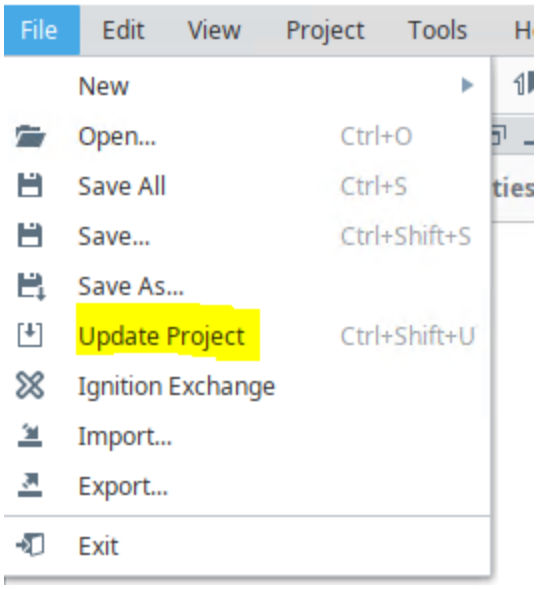


18. In the Translation Manger screen, select Import the Translation file. The translation files are available in the PE Ignition package.

19. Save the Project changes by selecting Save.



20. Then select Update Project.



EQMS PE Integration Configuration

Note: This section only applies if EQMS is installed.

1. Download and install the Web Service module in Ignition. Make sure to install the same version as the Ignition version that is installed:

   https://inductiveautomation.com/downloads/third-party-modules

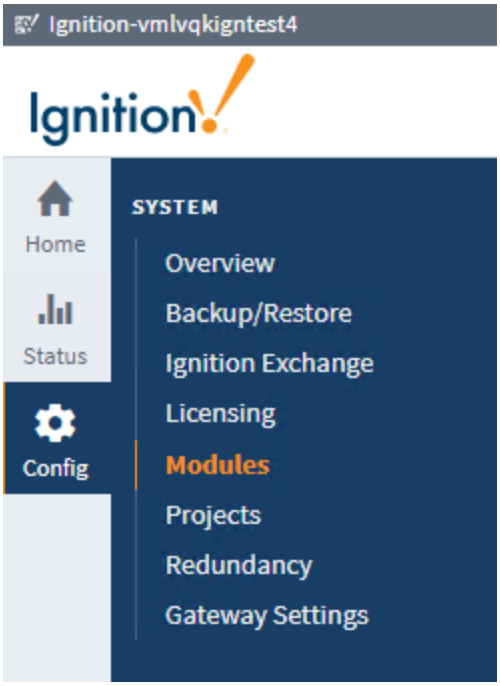
2. Download the Web Services Module file (`Web Service-module.modl`) for Sepasoft, Inc.

Sepasoft, Inc. MES Modules for Ignition	Version	Checksum
Web Services Module (4.7 MB)	2.80.0.2006162356 STABLE	sha-256

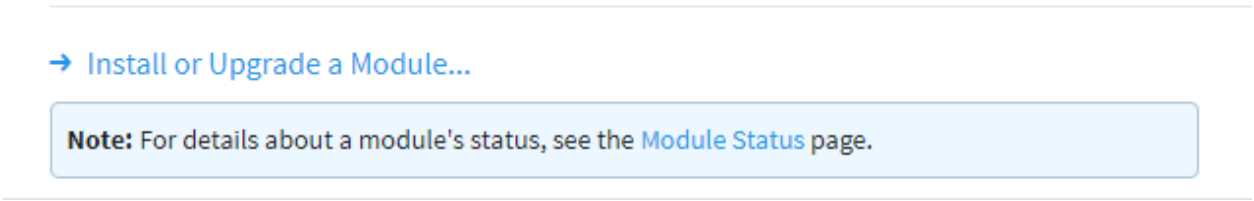
3. Open Ignition Gateway in the web browser.

   `http://<ignitionhostname>:<port>`

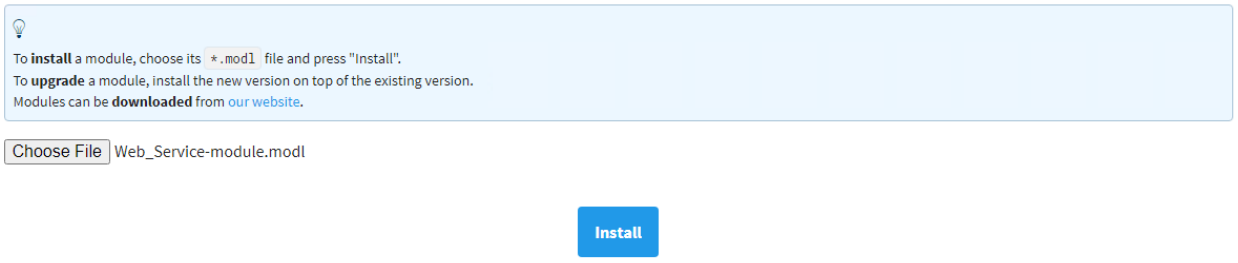
4. Select Modules from the Config menu.



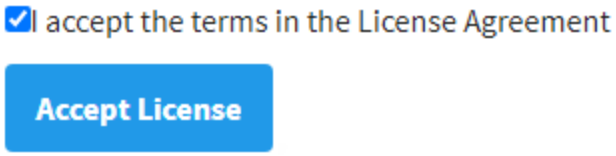
5. Scroll to the bottom of the page and select Install or Upgrade a Module.



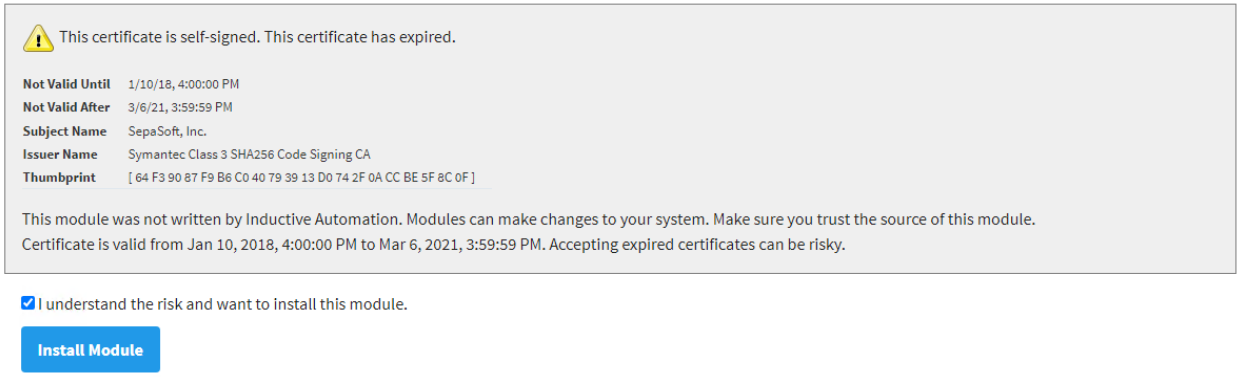
6. Select Choose File and then select `Web_Service-module.modl`, which was downloaded. Select Install to install the module.



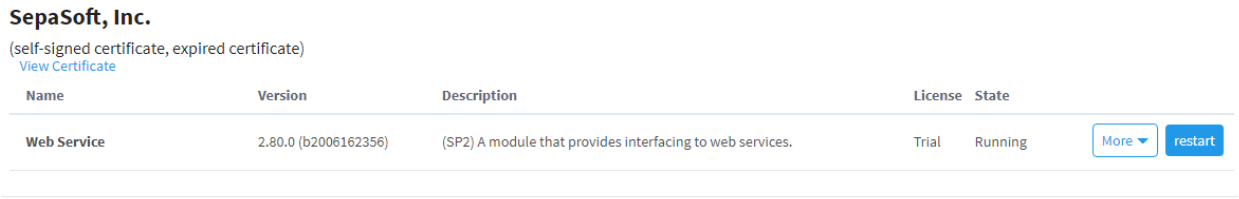
7. Read and accept the terms in the License Agreement and then select Accept License.



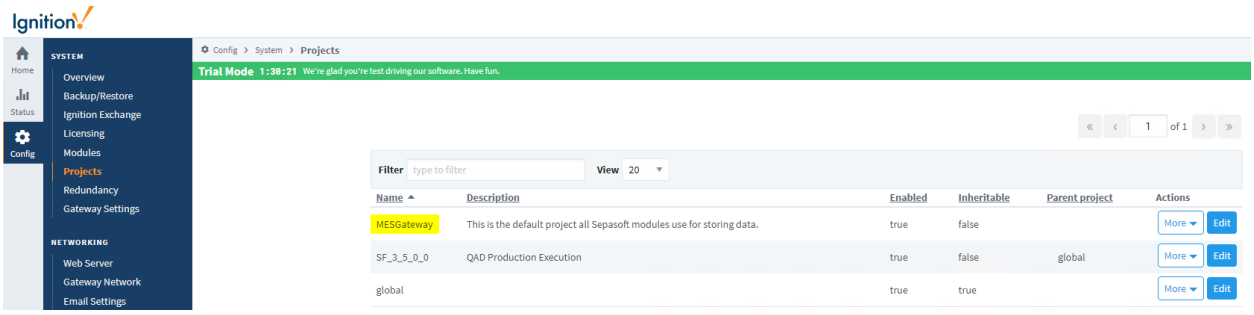
8. Accept the self-signed certificate and then select Install Module.



9. After the Module is installed, the Config > Module screen will display the SepaSoft, Inc. module for web service.

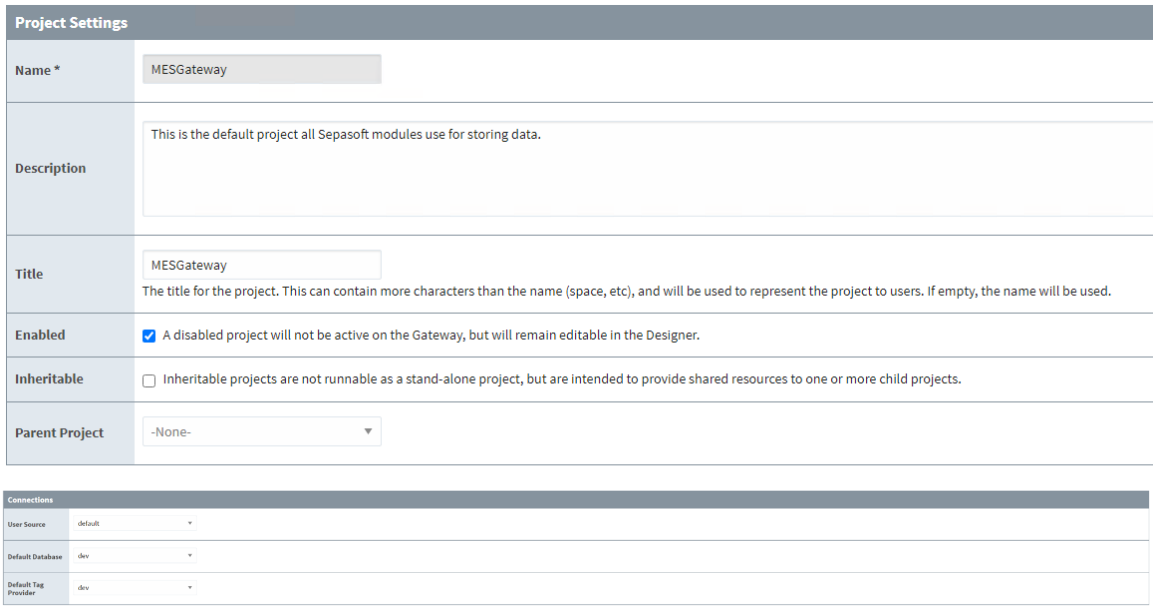


10. The new MESGateway project, which is now available in the Config >Project Web screen, needs to be updated to assign the correct Database and Tags.
11. Select Edit for the MESGateway project.

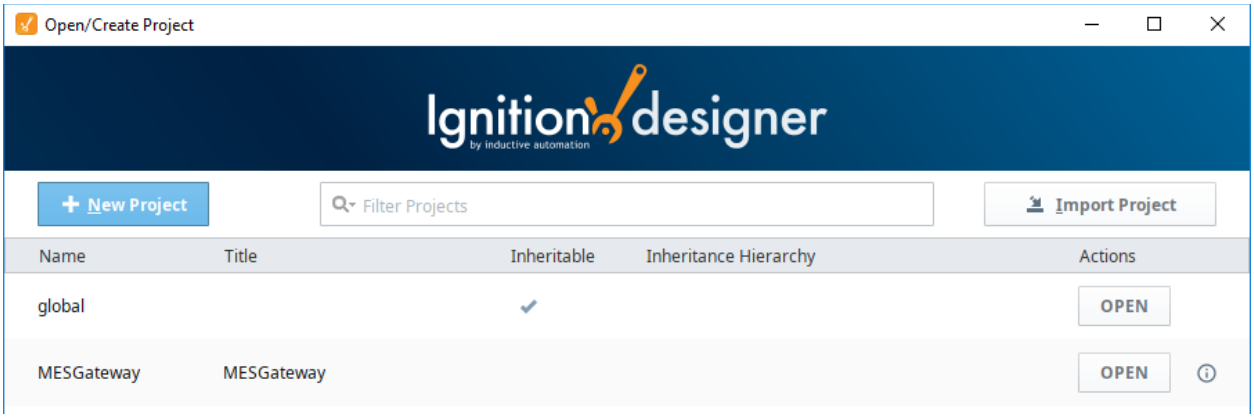


12. Assign the Database and Tags to the MESGateway project and then select Save.

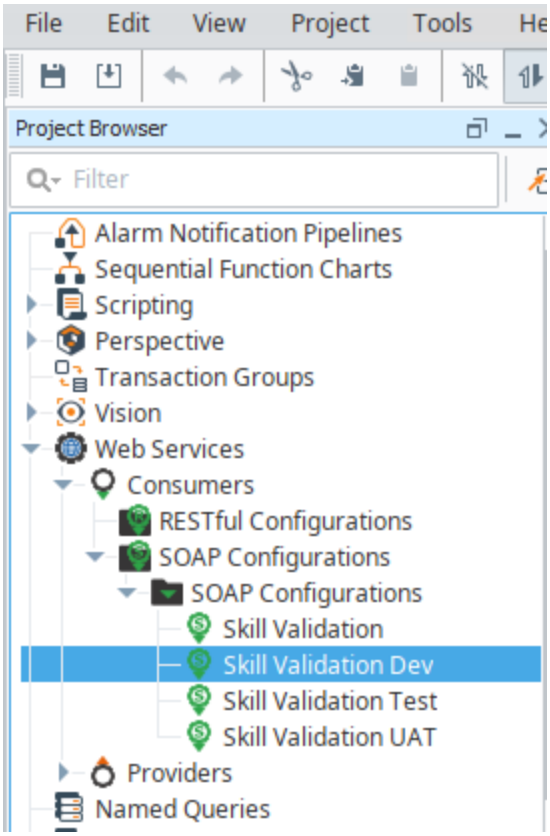
Field	Setting
User Source	default
Default Database	dev
Default Tag Provider	dev



13. Next, configure the Web Services module in Ignition Designer to point to the EQMS host. Open the Ignition Designer and select the MESGateway Project.

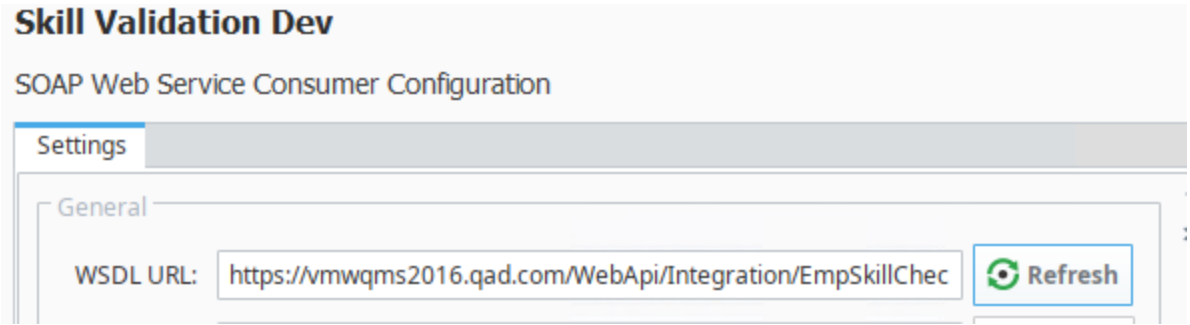


14. From the Project Browser, select Web Services > Consumers > SOAP Configurations > SOAP Configurations > Skill Validation Dev.

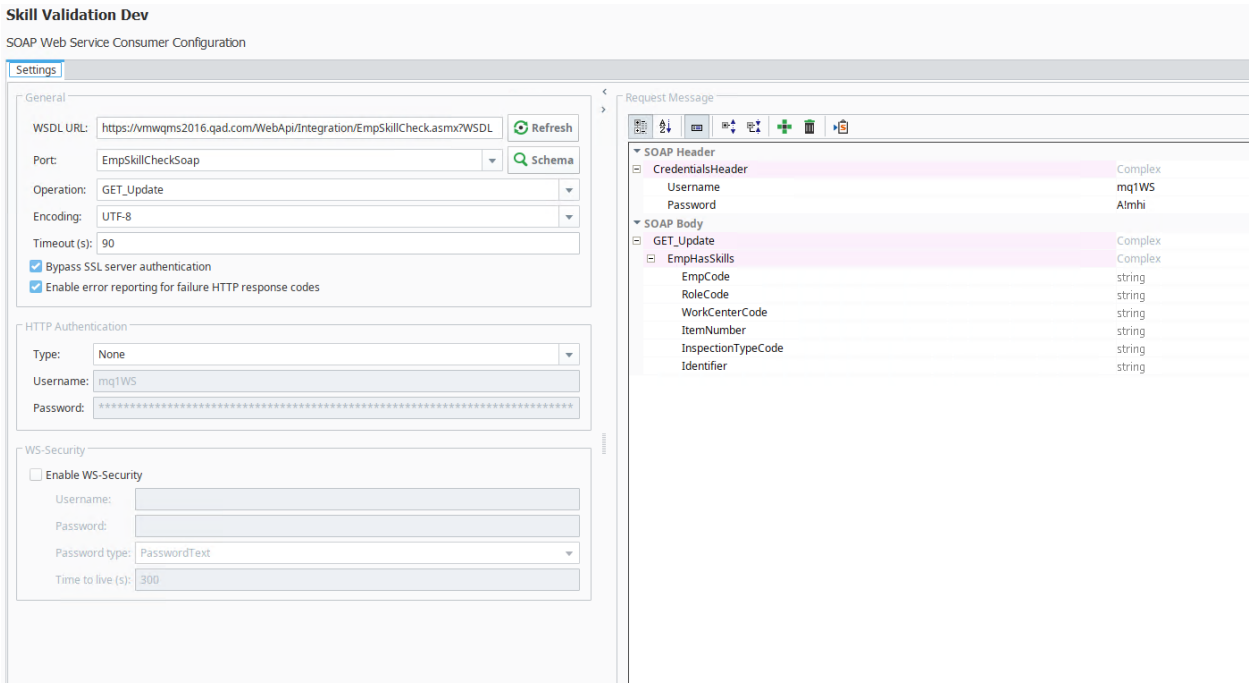


15. In the Skill Validation Dev window, enter the following WSDL link for the QMS host that needs to be integrated:

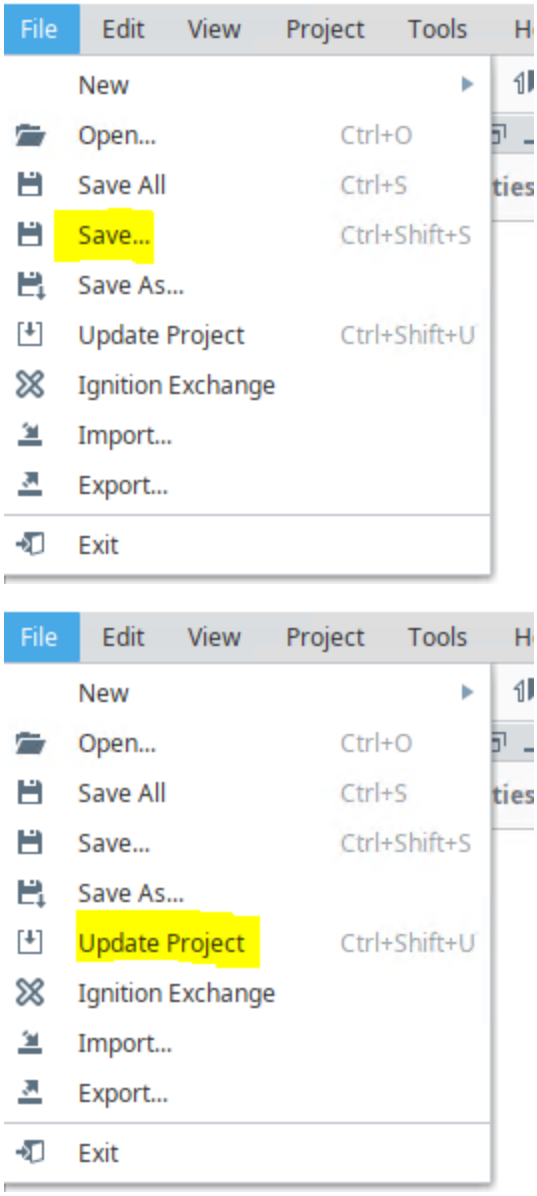
```
https://<EQMS hostname>/WebApi/Integration/EmpSkillCheck.asmx?WSDL
```



16. Select Refresh to refresh all the values on the screen.
17. Enter the values as shown. The WSDL URL must contain the EQMS host name.



18. Save the Project changes and then select Update Project.



Installing New Fonts

1. Take a Gateway backup before making any changes. From Ignition Gateway, you can take a Backup by navigating to Config → Backup/Restore → DownLoad Backup.
2. Download a compatible font. The font should be an otf, ttf, woff or any other Linux approved font type. One suggested site, which provides a lot of fonts is https://fonts.google.com/. However, you can download fonts from other sites as well.
3. Place the downloaded font/fonts on Ignition Server.

4. Make note of the directory where the font files are stored and then locate the install directory for Ignition, which is usually the location where the `ignition.sh` script is placed.
5. Execute the script `peInstallFont.sh` from the PE Ignition package. This script will prompt for Ignition install the directory and font directory. Specify the directory noted in the previous step.
6. Once the script is executed, the font will be installed for Ignition. Clear the Unix server cache for changes to take effect. The cache can be cleared using https://linuxhint.com/clear_cache_linux/ or rebooting the server.
7. On a Windows machine, clear the cache for the Vision client. Delete folder `<user directory>\.ignition\cache\resources`. For example, `c:\users\<userid>\.ignition/cache/resources`.
8. Reinstall the Vision Client by downloading the Vision client from the Ignition Gateway link.

Updating Nifi Flow Components

Follow these steps if you are currently using QAD Production Execution and are updating the Nifi Flow Components:

1. Deactivate or remove the previous version of the Nifi Flow Components. It is recommended that you deactivate the previous version of the package rather than deleting it because it allows you to easily reactivate the previous version if necessary. See Deactivate Previous Version of Nifi Flow.
2. Install the new version of the Nifi Flow Components using yab. See Installing a New Version of the Nifi Flow Components.

Deactivate Previous Version of Nifi Flow

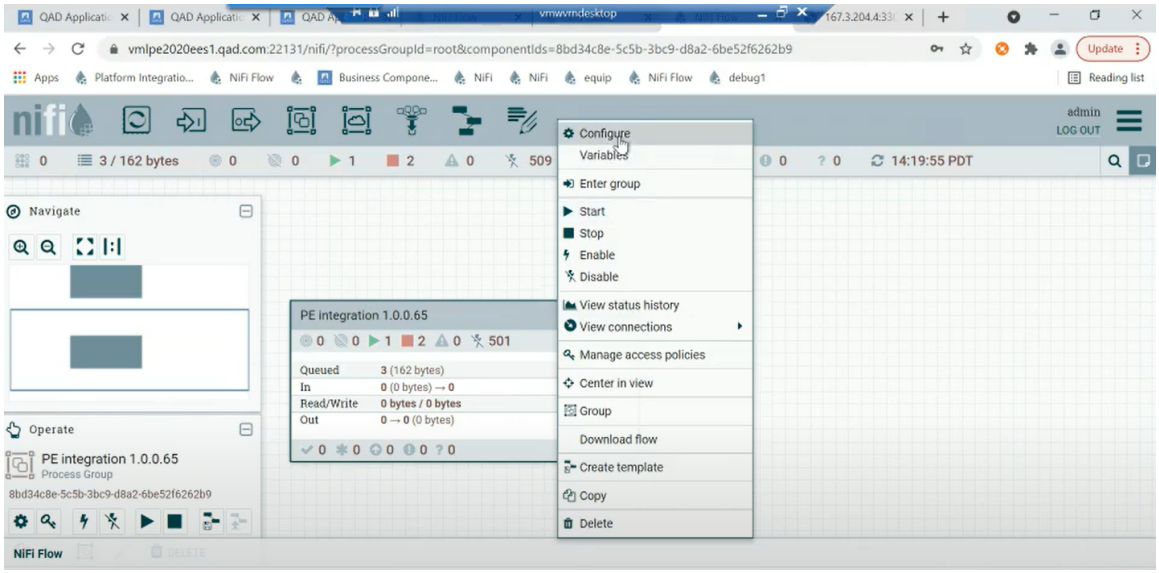
Note: This task only applies if you are upgrading the Nifi flow and there is a prior version of the `penifi` package deployed.

If there is a prior version of the `penifi` package deployed, then the Nifi flow needs to be deactivated or removed before installing the new version. This is required because the updated Nifi flow uses the same port as the previous flow.

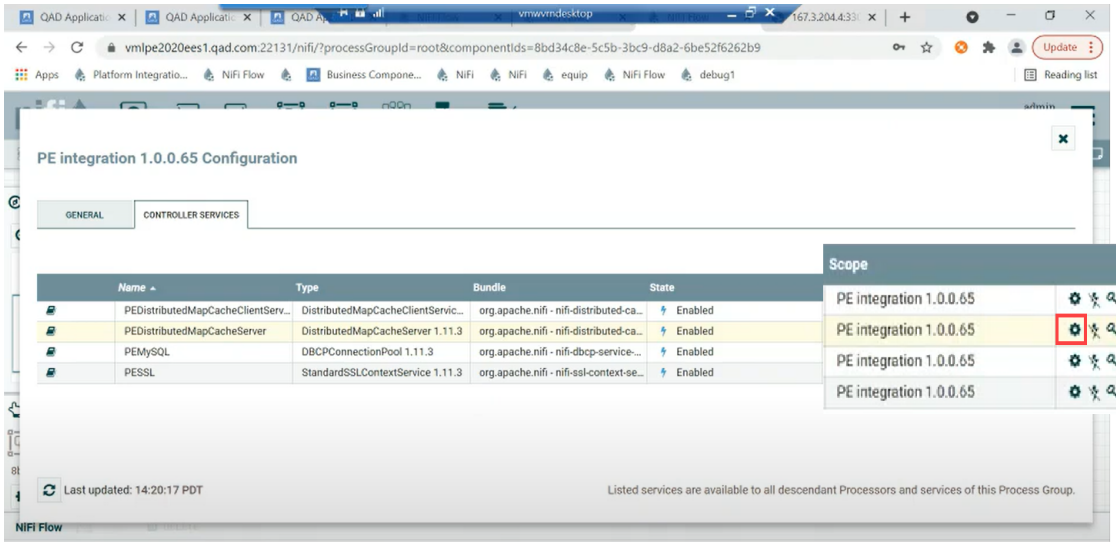
Note: It is recommended that you deactivate the previous version of the `penifi` package rather than deleting it because it allows you to easily reactivate the previous version if necessary.

Follow these steps to deactivate the previous version of the `penifi` package:

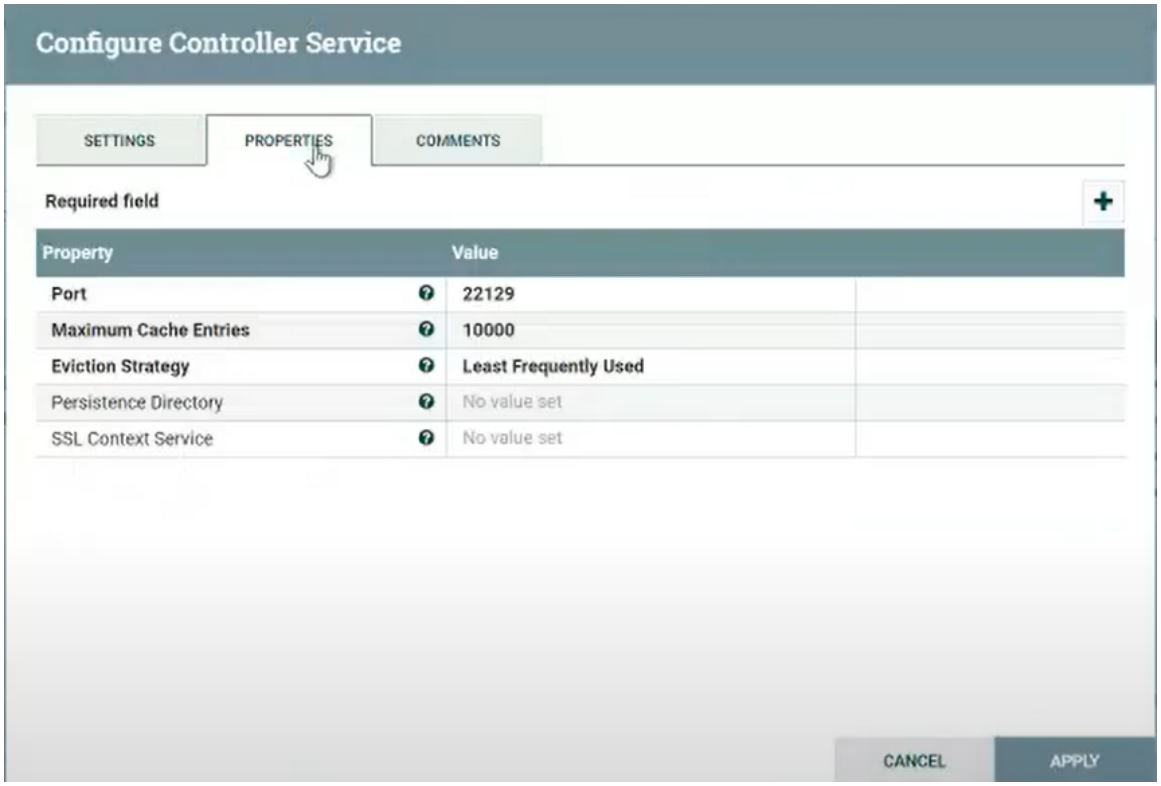
1. In Nifi, right-click on the flow and select Configure.



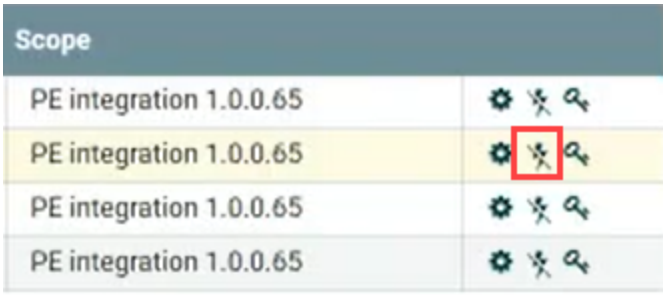
2. In the Configuration window > Controller Services tab, select the Settings button for the `PEDistributedMapCacheServer` piece of the flow.



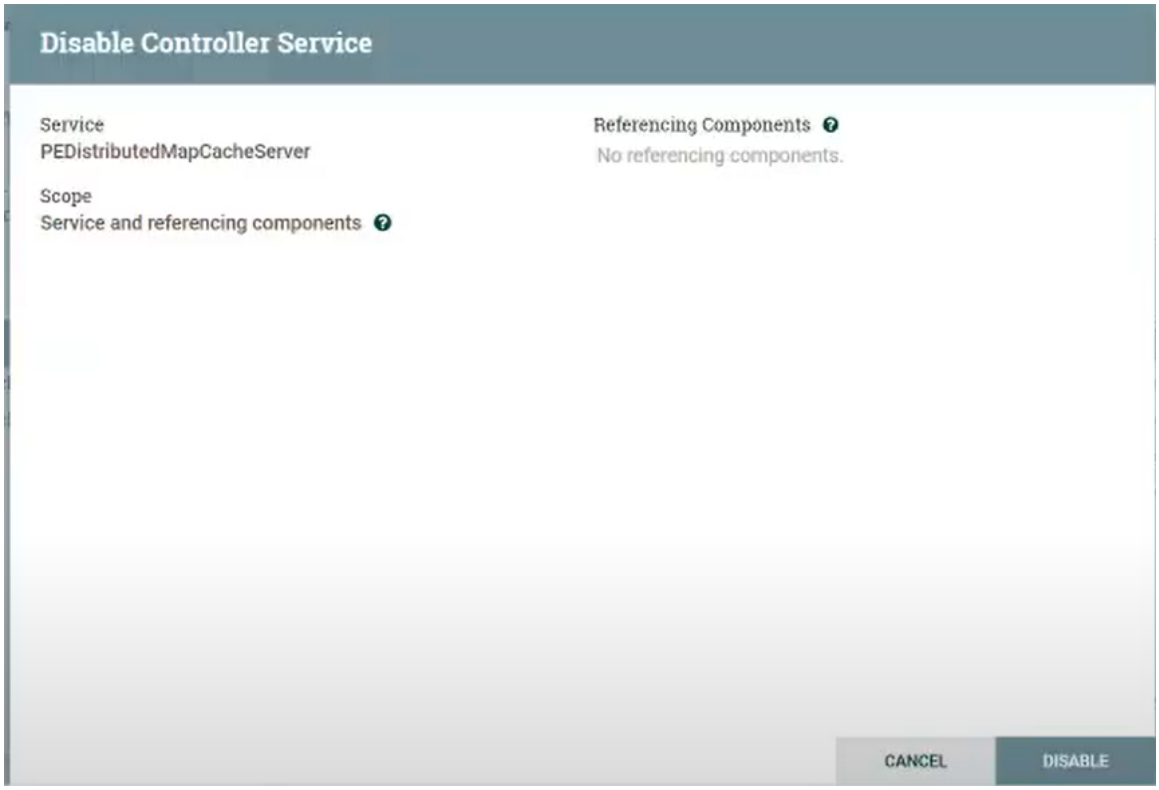
3. A Configure Controller Service window appears. Write down the Port number. If you needed to re-create this piece of the flow, you could create a new piece with the same name (PEDistributedMapCacheServer) and the port number you just recorded. Close the Configure Controller Service window.



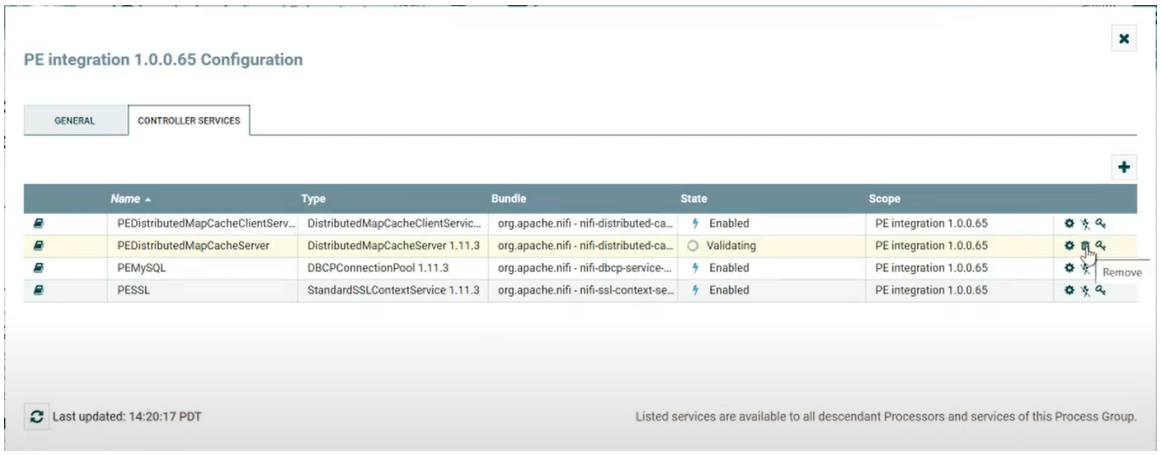
4. Click the Disable button for PEDistributedMapCacheServer.



5. A Disable Controller Service window opens. Select Disable and then close the window.



6. Once you disable PEDistributedMapCacheServer, a Delete button appears, allowing you to delete PEDistributedMapCacheServer. Select the Delete/Remove button.



7. A Delete Controller Service window opens. Select Yes to delete.

8. Because you deleted this piece of the `PE integration x.x.x.xx` flow, this version of the flow is now disabled and not functional. You can now import the new version of the `PE integration x.x.x.xx` flow.

   Note: If the new version of the `PE integration x.x.x.xx` flow does not function properly, you can re-enable the old version by re-creating the piece you just deleted. To re-create the piece, you need the information you recorded in the Configure Controller Service settings window.

Deleting a Nifi Flow Using YAB

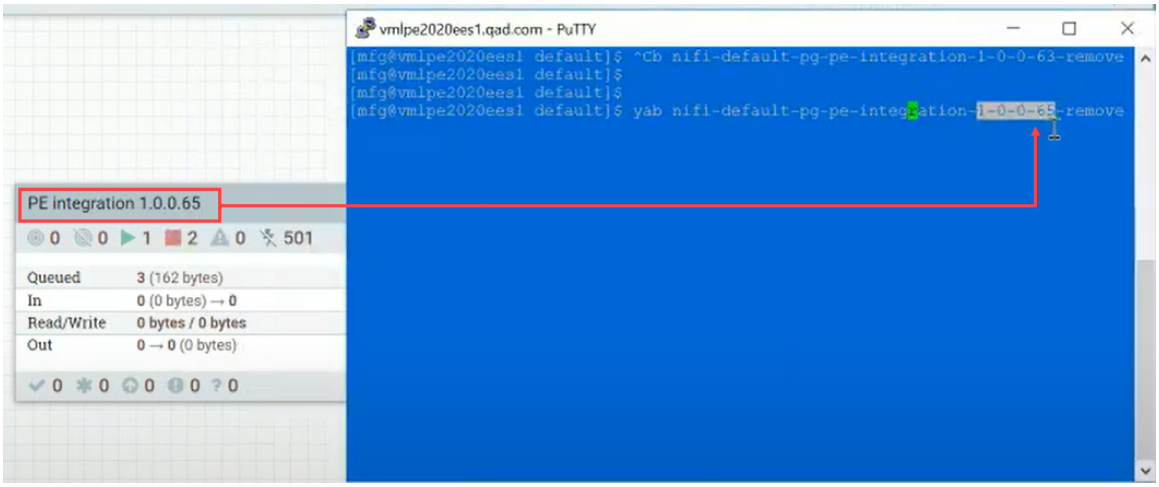
Follow these steps to delete a Nifi flow using yab:

1. Enter the following yab command to view a list of Nifi processes:

   ```
   yab nifi-status
   ```

2. Then enter the following command to remove the version of the Nifi flow you want to remove:

   ```
   yab nifi-default-pg-pe-<remote/local>-x-x-x-x-remove
   ```



Deleting a Nifi Flow Using Nifi

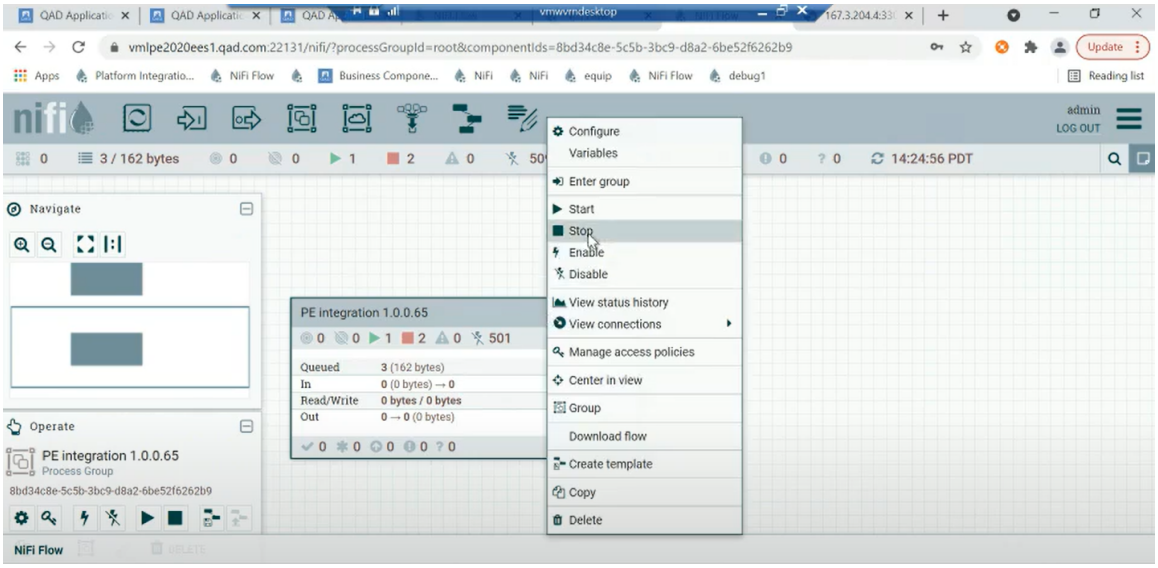
Note: It is recommended that you delete Nifi flows using yab as the process is easier and quicker.

Before deleting a Nifi flow you must do the following; otherwise, you will receive an error:

- Disable all associated Controller Services.
- Clear any associated Queues.

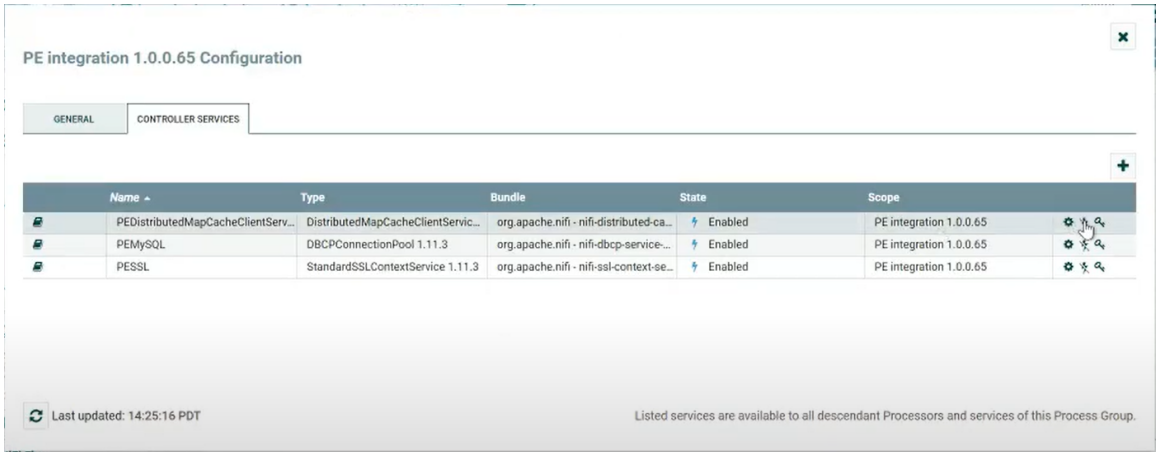
Disable Controller Services

1. Right-click on the Nifi flow and select Stop.

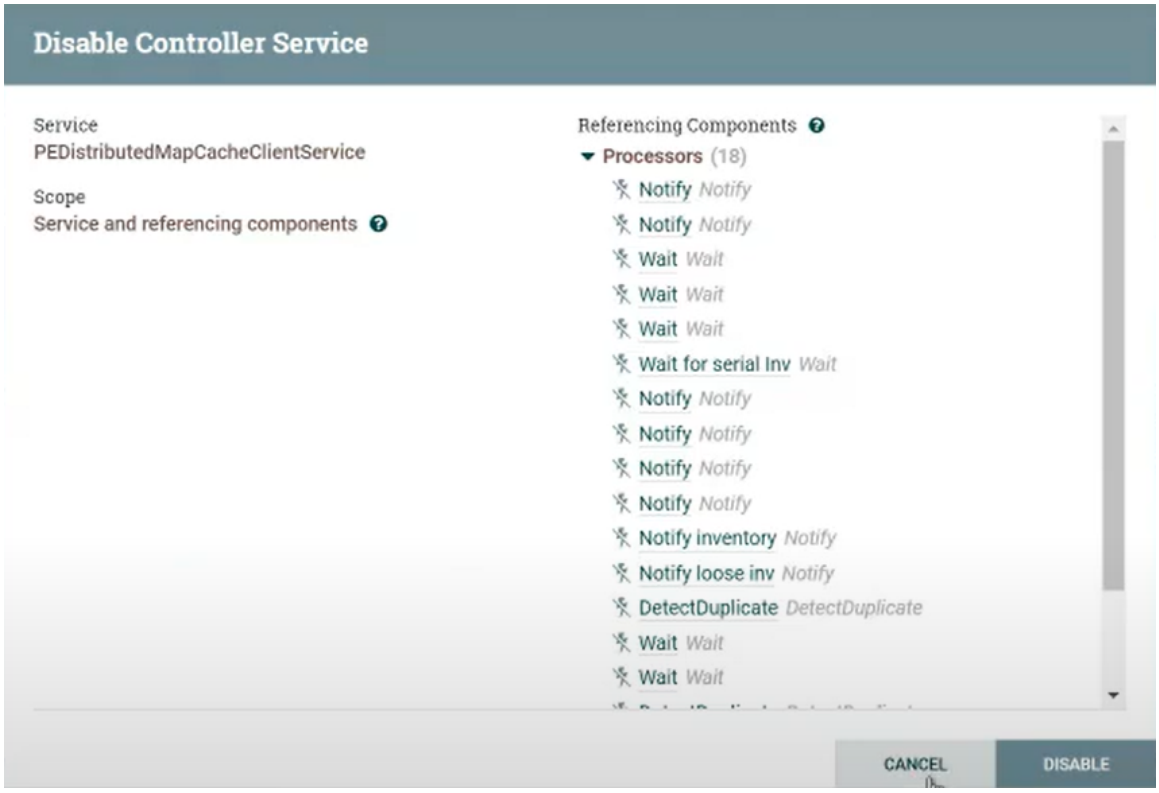


2. Right-click on the Nifi flow again and select Disable.

3. Right-click on the Nifi flow again and select Configure. The Nifi flow Configuration window opens.



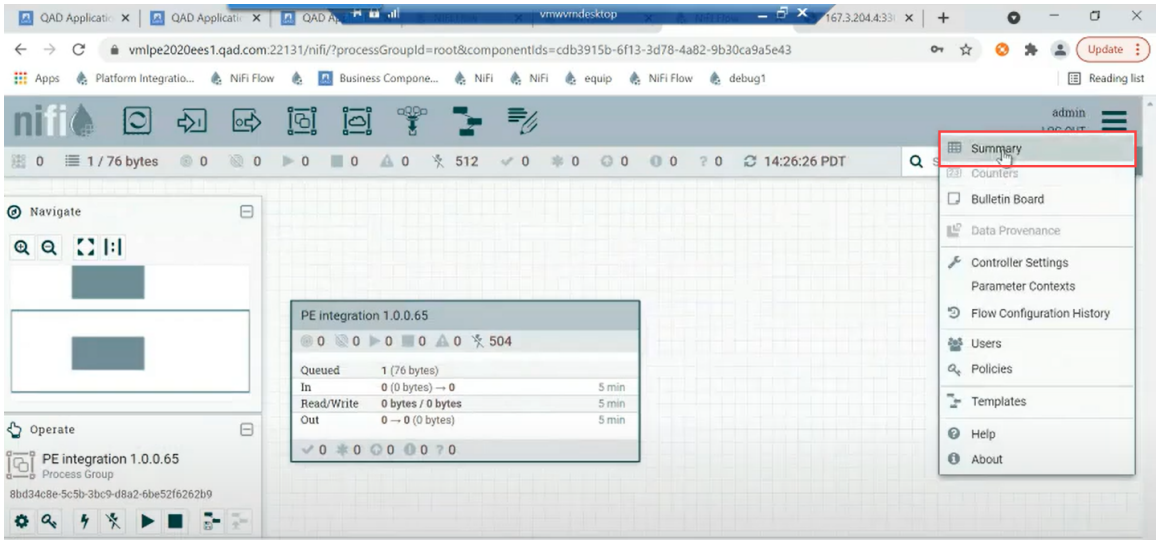
4. Select the Disable button for the first Controller Service. A Disable Controller Service window opens. Select Disable and then close the window.



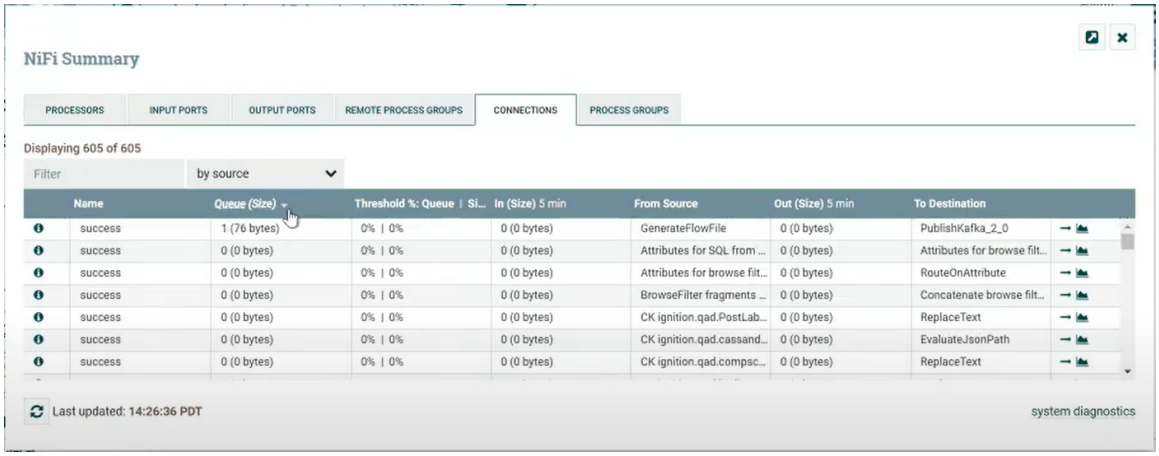
5. Repeat this step for the other Controller Services until all are disabled.

Clear Nifi Flow Queues

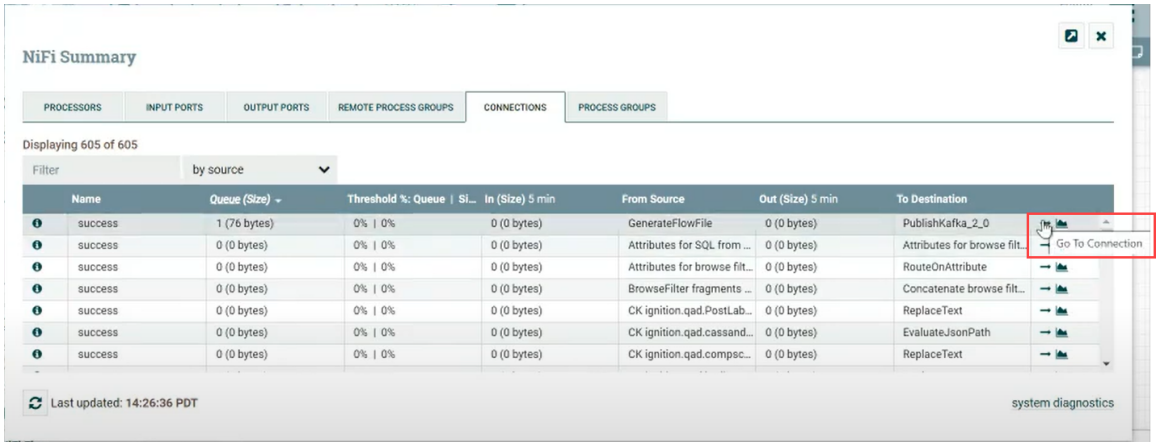
1. Select Summary from the Hamburger menu in the top right of the screen. The Nifi Summary window opens.



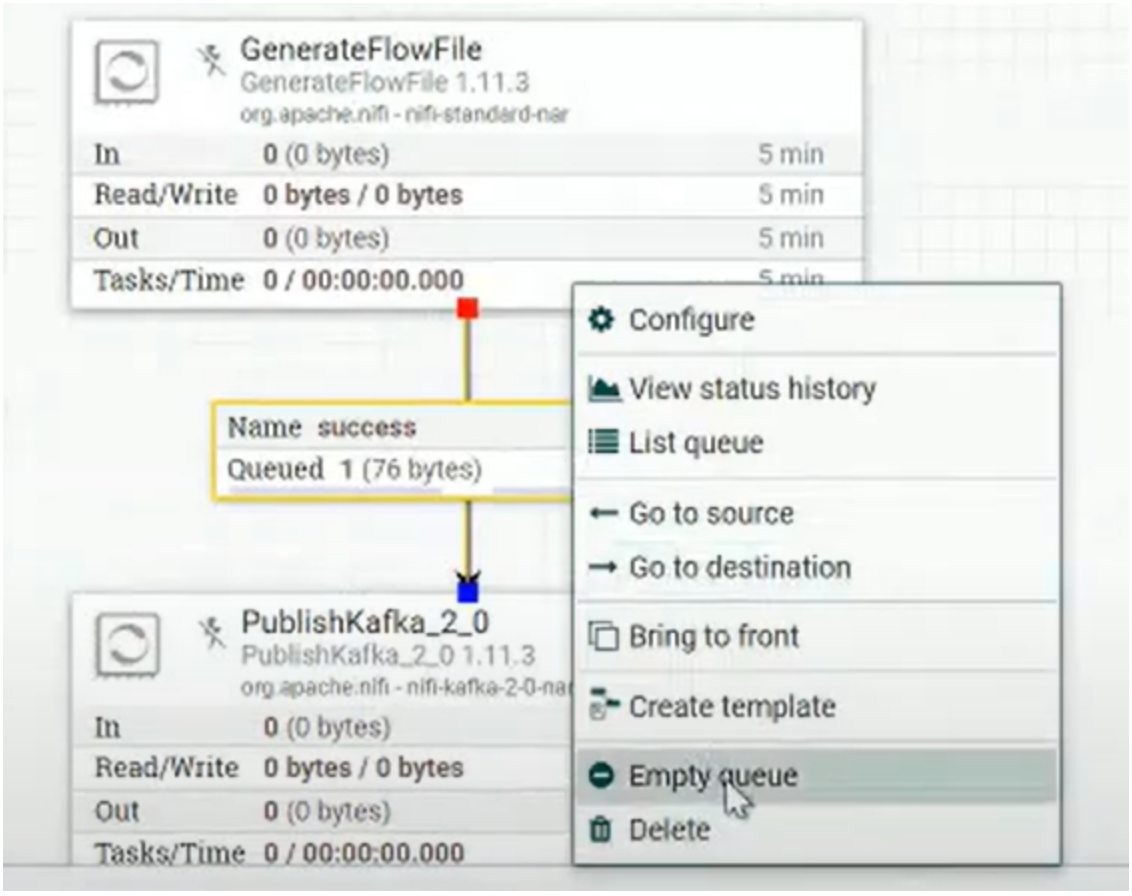
2. On the Connections tab, sort the Queue (Size) column in descending order so that any Queues that are non-zero are displayed first.



3. Select the Go To Connection button for the first Queue.



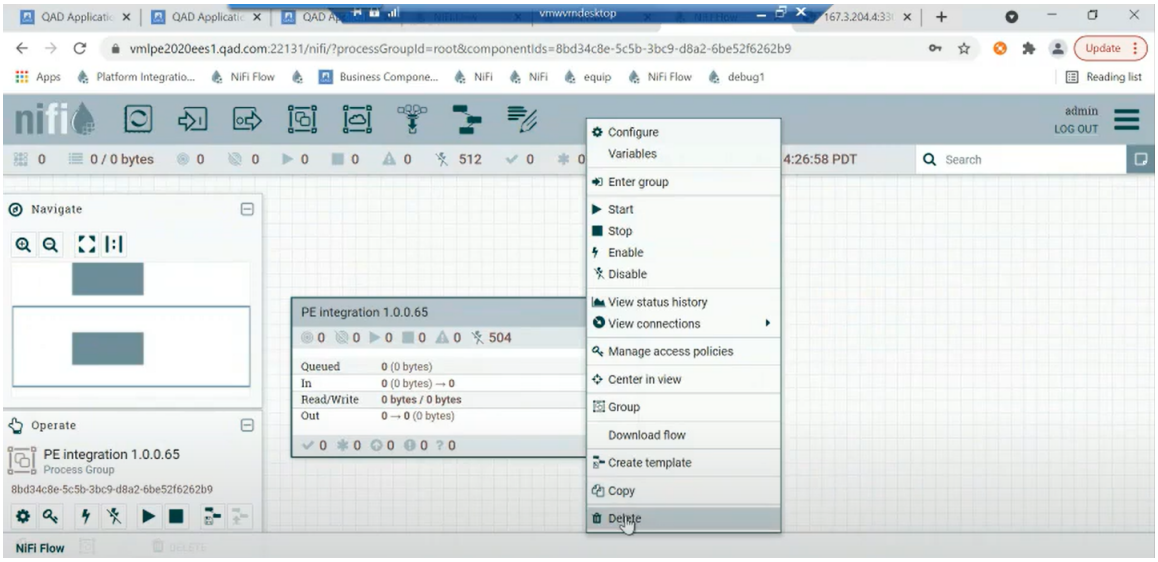
4. Right-click on the Queue and select Empty Queue.



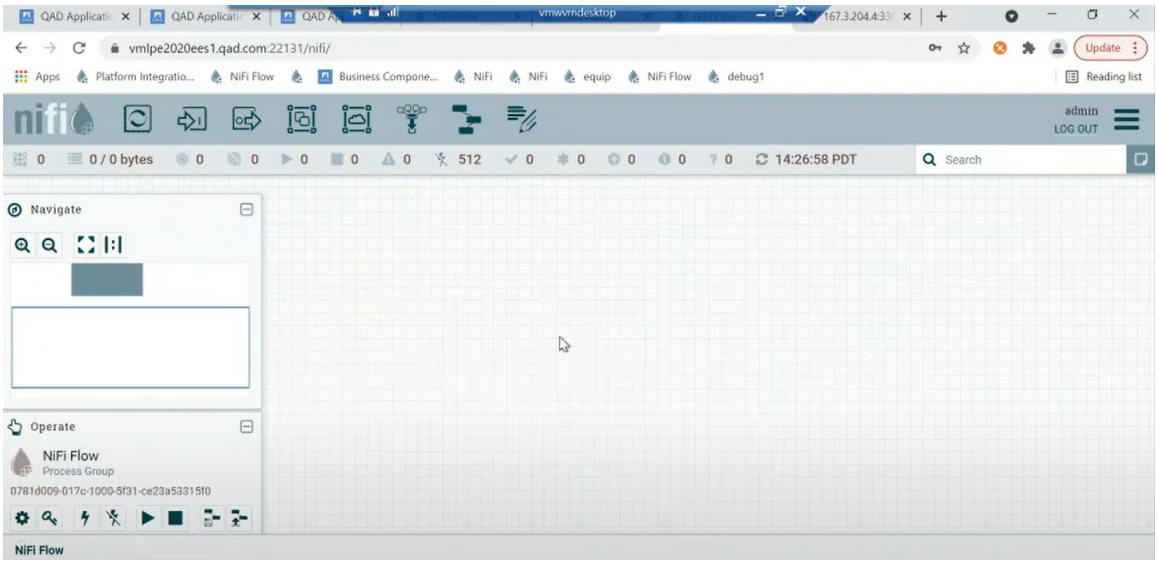
5. Navigate back to the Nifi Summary window and repeat these steps for any remaining Queues.

Deleting a Nifi Flow

1. To delete a Nifi flow, right-click on the flow and select Delete.



2. The flow is deleted.



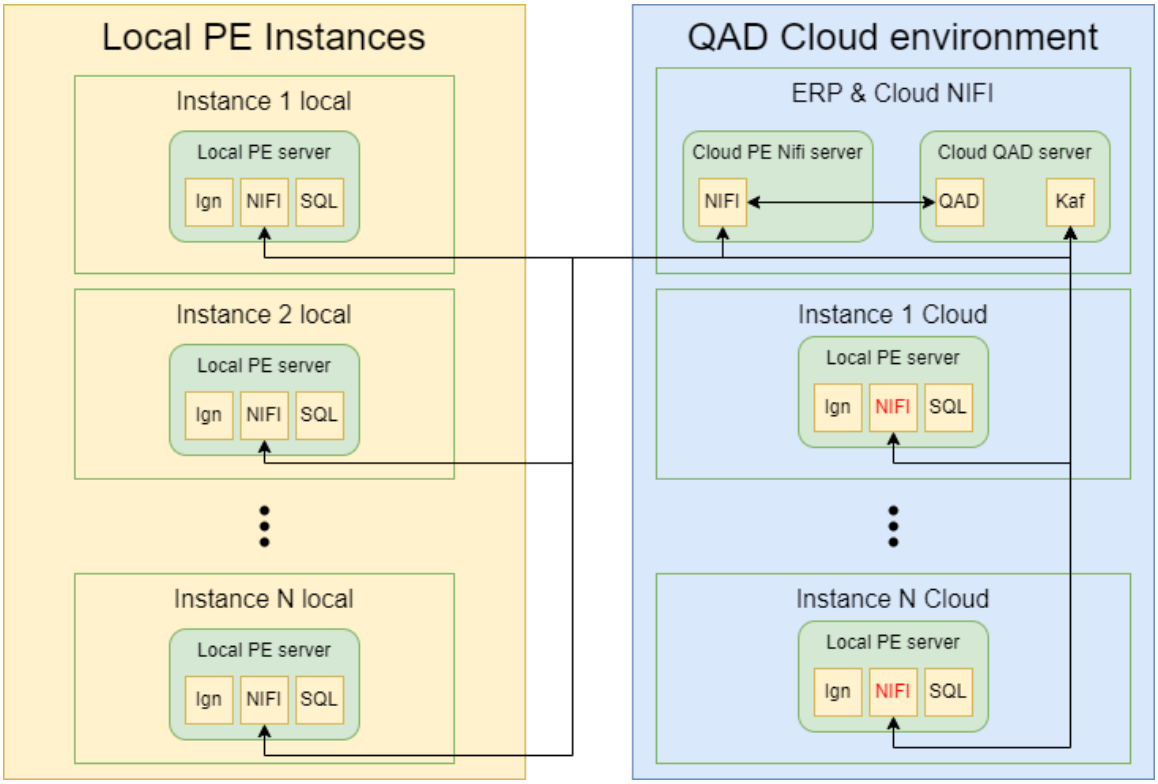
Chapter 3:

Post Installation Steps

This chapter provides detailed instructions on the post installation steps:

Setup Instances

QAD Production Execution can be run with one EE/Adaptive UX ERP installation and multiple instances. Each instance gets its own instance name. In the PE NiFi remote/cloud server, the data is identified by the domain and site information and it is sent to the right local NiFi instances. See the diagram below.



Naming the Instance

On the PE Ignition box, NiFi should be identified as an instance in the following file:

`<nifi install path>/conf/variable-registry-templates/pevariables.properties`.

For the implementation, the variable “instance” in the file should contain a unique value. The default is qad.

Domain and Site Setup for the Instance

On the NiFi remote box (for the default installed beside the ERP), an instance must be coupled with one or more domain and site combinations in the following file:

```
<nifi install
path>/common/productionexecution/conf/peinstance.csv
```

This file contains a comma-separated list for domain, site, instance, and a command. All values must be quoted. For example:

Instance: INST001, domain is CUST001 and site is CST01S01

Instance: INST002, domain is CUST002 and site is CST02S01

domain is CUST002 and site is CST02S02

Using the example above, the PE instance .csv will look like the following:

```
Peinst_domain,peinst_site,peinst_instance,peinst_remarks

"CUST001","CST01S01","INST001","Customer 1, domain CUST001 site
CST01S01"

"CUST002 ","CST02S01","INST002","Customer 2, domain CUST002
site CST02S01"

"CUST002 ","CST02S01","INST002","Customer 2, domain CUST002
site CST02S02"
```

Starting PE and Setting Up PE Admin Permissions

Before the system administrator can set up PE screens, terminals, Overview layouts, and so on, they must perform the following steps:

1. Log in to .NET UI and use the Role Membership screen to assign "webui" Role to the PE Apps userid.
2. Log in to QAD Adaptive ERP using the PE Apps userid and export PE Apps seed data (APISettings, Buttons, Icons, Colors) from QAD domain into the PE Domain.
3. Update the API Settings screen in QAD Adaptive ERP with the required information. The information entered in this screen can be viewed in the Admin > Integration Management screen in PE.

Field	Value
Host: Port	<Nifi Local Hostname>:<PE RestAPI port> **Note**: The <Nifi Local Hostname > variable needs to be replaced with the hostname where the Nifi local package is installed. The <PE RestAPI port> variable needs to be replaced with the PE REST API port value.

Images	<GatewayURL>/images **Note**: The <Gateway URL> variable needs to be replaced with the Ignition Gateway URL.
Docs	<GatewayURL>/docs
Videos	<GatewayURL>/videos
Sounds	<GatewayURL>/sounds
Web Directory	<Ignition Install Dir>/webserver/webapps/main/ **Note**: The <Ignition Install Dir> variable needs to be replaced with the directory where Ignition is installed on the server.

4. Perform a data-sync between PE and QAD Adaptive ERP for users and roles. This step can be performed after setting up the QAD Adaptive ERP base data for Users, Employees, and Roles or after all the ERP base data is set up. This step is required because until the data-sync is performed, there are no users in MariaDB.

Follow these steps to update the Integration Management screen with the required information and to perform a data sync for ERP user and role data:

1. Log in to the Vision Client using the Gateway Admin user ID and password that was created during the PE installation process.
2. Select the Admin > Integration tab and update the fields with the correct data syncing information.

Field	**Value**
Host: Port	<Nifi Local Hostname>:<PE RestAPI port> **Note**: The <Nifi Local Hostname > variable needs to be replaced with the hostname where the Nifi local package is installed. The <PE RestAPI port> variable needs to be replaced with the PE REST API port value.
Images	<GatewayURL>/images **Note**: The <Gateway URL> variable needs to be replaced with the Ignition Gateway URL.
Docs	<GatewayURL>/docs
Videos	<GatewayURL>/videos
Sounds	<GatewayURL>/sounds

Web Directory	<Ignition Install Dir>/webserver/webapps/main/ **Note**: The <Ignition Install Dir> variable needs to be replaced with the directory where Ignition is installed on the server.

3. Select the Admin > Data Management tab and sync the PE Database with QAD Adaptive ERP user data, role data and role mapping data. For more information, see Data Management.
4. Select the Admin > Ignition Roles tab and assign the correct permissions to each role. For more information, see Define Role Functions.
5. Log out of PE and then log back in as the PE Administrator (SFAdmin) and complete the setup for screens, terminals, and so on.