

Lab 3

On-Premises Deployments (Optional)

Overview

This Lab is considered optional to the completion of the API-Led Connectivity Workshop.

Using Runtime Manager, you can register and set up the properties of servers, server groups, and clusters that reside on-premises. Once these are set up, you can select them as the deployment target for your applications, as an alternative to deploying to CloudHub.

Runtime Manager provides a single management interface where you can manage applications running in the cloud, on local servers, on server groups that are on-premises within your data center, or on clusters of servers. Applications that are on-premises run on Mule runtime instances on a single server or on a group of servers.

In this Lab you will install and run the Mule Enterprise Edition Runtime locally, to simulate a standalone on-premises deployment. You'll then be able to use Runtime Manager to deploy and manage the `api-omni-channel-complete-v4.0` application you used in the rest of [Module 7](#).

NOTE

Directions for completing this Lab are not as specific as others, due to differences between Operating Systems and environments. If you run into a problem, please ask the MuleSoft team leading the Workshop for help.

The lab will consist of several steps

- [Step 1: Download the Runtime](#)
- [Step 2: Register your Runtime](#)
- [Step 3: Start your Runtime](#)
- [Step 4: Deploy to your Runtime](#)
- [Step 5: Manage your Application](#)

Step 1: Download the Runtime

In this step, you will download and extract the standalone version of the Mule Enterprise Edition Runtime.

TIP

If you are using a MuleSoft-provided Workshop RDP instance, Step 1 has been done for you. The Mule Runtime is already downloaded. Unzip the file into the directory `C:\Users\workshop\mulesoft`. Once the folder is unzipped go directly to [Step 2](#).

1. First, [Download](#) the Mule Enterprise Edition Runtime.

Download Mule runtime

30 Day Trial

Download

SHA1: 0a97dfe2087bee1fc6c9337dd1a018862be2b570

By downloading Mule Runtime, I agree to [these terms](#)

[Installation instructions](#)

IMPORTANT

MuleSoft makes available trial versions of its Enterprise Edition runtime for the purpose of evaluation. Though perfect for exploring the myriad uses of Mule and Anypoint Studio, the trial license for Enterprise limits usage of the runtime to 30 days.

Before using a Mule Enterprise Edition Runtime in a production environment, you will need to [obtain](#) and [install](#) an Enterprise license.

2. Now extract the archive you just downloaded. It already contains a folder (e.g. mule-enterprise-standalone-4.1.4/) so just move that folder to the path where you want your Mule Runtime to reside. For example:

- **Windows:** `C:\Users\\mulesoft\mule-enterprise-standalone-4.1.4`
- **Mac or Linux** `~/mulesoft/mule-enterprise-standalone-4.1.4`

TIP

- For the rest of this Lab, instructions will refer to the path to the Mule runtime as simply `<MULE_HOME>`. In production installs you would define `MULE_HOME` as a variable.
- For Workshop RDP instance users, `<MULE_HOME>` is `C:\Users\workshop\mule-enterprise-standalone-4.1.4`

You have installed the Mule EE Runtime. That's all there is to it.

Step 2: Register your Runtime

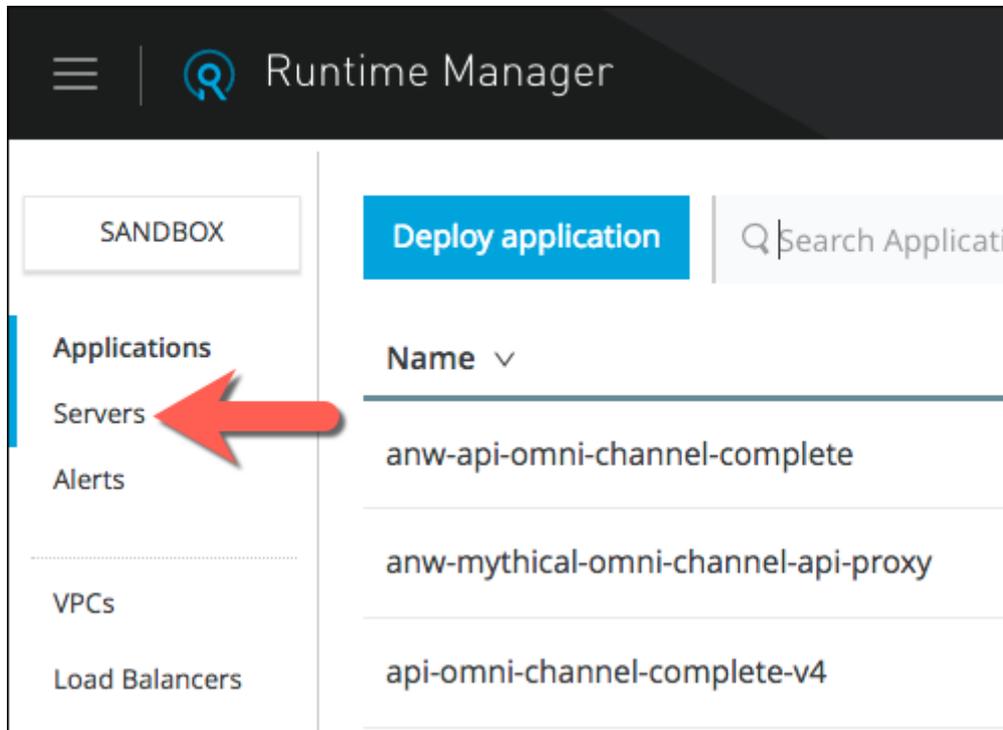
In this step you will register your new Mule instance with Anypoint Runtime Manager via the Runtime Manager Agent. This is done by executing a command, which will be provided by Runtime Manager.

NOTE

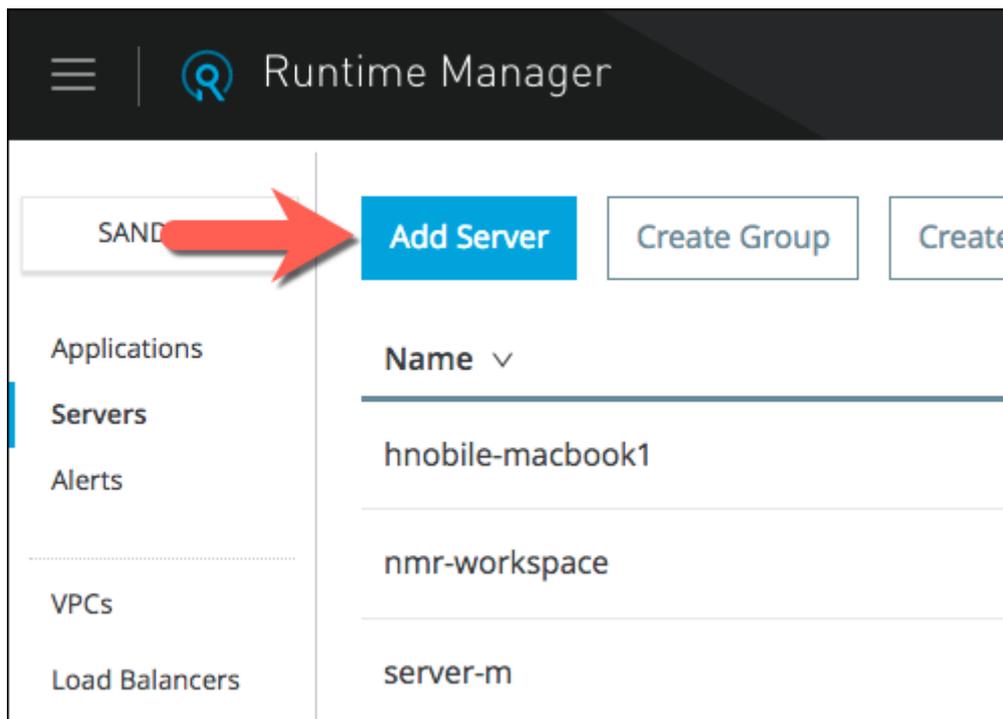
The Runtime Manager Agent is a Mule plugin that exposes the Mule Agent API as a service, allowing users to securely manipulate and monitor Mule Runtime instances from external systems such as Anypoint Runtime Manager.

In Runtime Manager, the term "Server" is used to mean a Mule runtime instance in a location other than CloudHub.

1. Open [Anypoint Runtime Manager](#) and **click** on the Servers section on the left.



2. If your Environment already has at least one Server deployed, you will first need to **click** on the "Add Server" button.



3. After clicking "Add Server", or if there are no other Servers in the Environment, you will see a command with a registration token (redacted in the example below) and a placeholder server-name. **Copy** this line to your clipboard.

```
./amc_setup -H <REDACTED>---54910 server-name
```

The `amc_setup -H` command configures the Runtime Manager Agent to create a hybrid management connection with Runtime Manager. The connection is to a specific environment for a specific business group in Anypoint Platform. This Server registration command will need to be run from the machine where your new Mule runtime resides.

4. From the command-prompt, use the `cd` command to navigate to the directory where you unzipped the Mule software distribution and then to the `bin` directory within.
 - **Windows:** `cd <MULE_HOME>\bin`
 - **Mac or Linux:** `cd <MULE_HOME>/bin`
5. **Paste** the server `amc_setup` command string into your shell window (or a text editor,) but **DO NOT** execute it yet. You'll want to make one or two changes before executing this command.

Change `server-name` to a meaningful name for your new Server. For example, you might use your machine's short hostname. On Windows machines, remove the `./` and append `.bat` so that the command will run correctly for your platform.

- **Windows:** `.\amc_setup.bat -H <access_token> <your-hostname>`
 - **Mac or Linux:** `./amc_setup -H <access_token> <your-hostname>`
6. Once you have modified the registration command, press **Enter/return** to execute it. The command will unpack the Mule Agent and establish a secure communication channel with ARM. If successful, the last few lines of the output should look like this:

```
Mule Agent configured successfully
Connecting to Core Services to extract client_id and client_secret
Credentials extracted correctly, updating wrapper conf file
```

Step 3: Start your Runtime

Your local Runtime will now show up in the [Servers](#) area of Anypoint Runtime Manager and, once started, will be available as a deployment target for your Mule applications. So let's do that now.

TIP

If you are using a MuleSoft-provided Workshop RDP instance, there is already a shortcut on the Desktop to start the Mule Runtime. Just double-click it and skip to [Step 4](#).

1. To start the Mule runtime, use the command appropriate for your platform within the `<MULE_HOME>/bin` directory

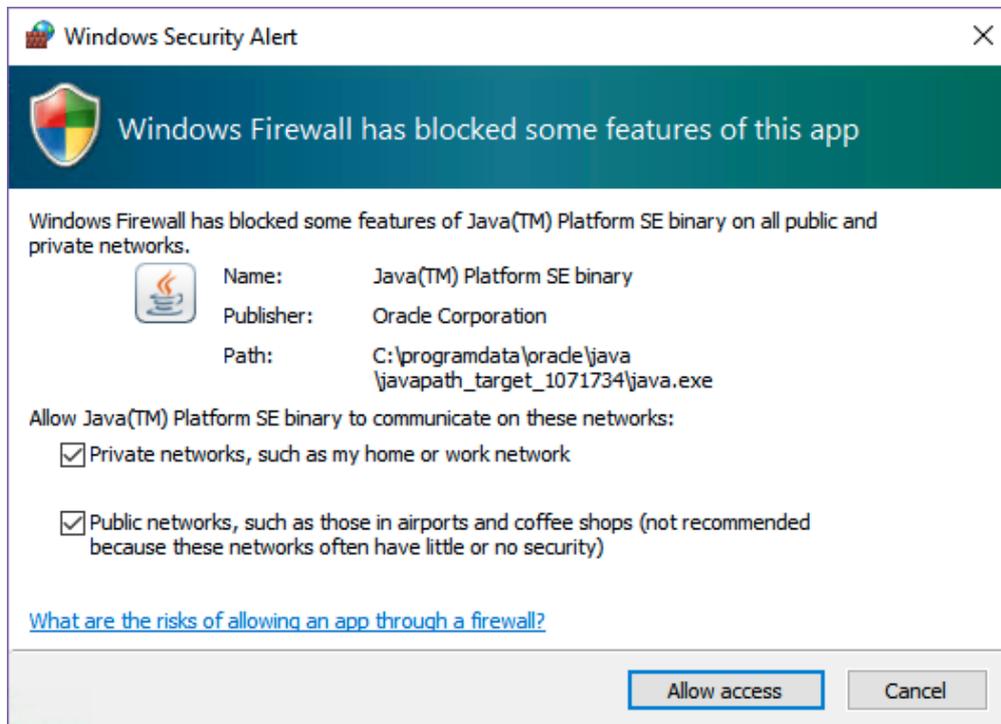
- **Windows:** `mule.bat`
- **Mac or Linux:** `./mule`

Mule Runtime Command-line Options

NOTE

Parameter	Description
<code>start</code>	Starts the Mule server in the terminal background.
<code>stop</code>	Stops the Mule server. The stopping process stops inbound endpoints to process in-flight messages. However due to a timeout duration not all messages are ensured to be processed before final shutdown occurs.
<code>restart</code>	Restarts the Mule server.
<code>status</code>	Displays the status of the Mule server (<code>running</code> or <code>not running</code>) and the process PID if running.
<code>dump</code>	(Unix only.) Dumps the Mule wrapper's core to <code>\$MULE_HOME/log/mule_ee.log</code> .
<code>console</code>	Start Mule in the terminal foreground (console mode). Same as running <code>mule</code> with no parameters.
<code>install</code>	(Windows only.) Install Mule as a Windows service.
<code>remove</code>	(Windows only.) Remove Mule from your Windows services.

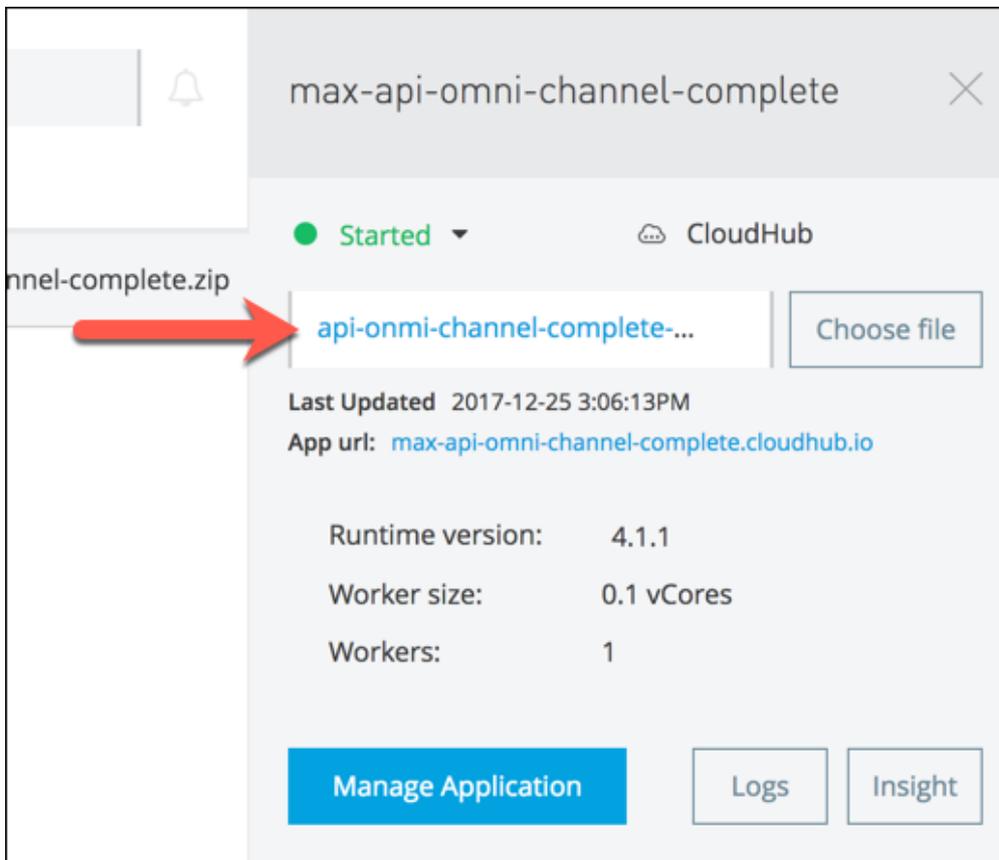
2. Once the runtime starts, on Windows machines, you may get a Windows Security Alert, saying you need to set a firewall exception. You will need to choose which networks and "Allow access".



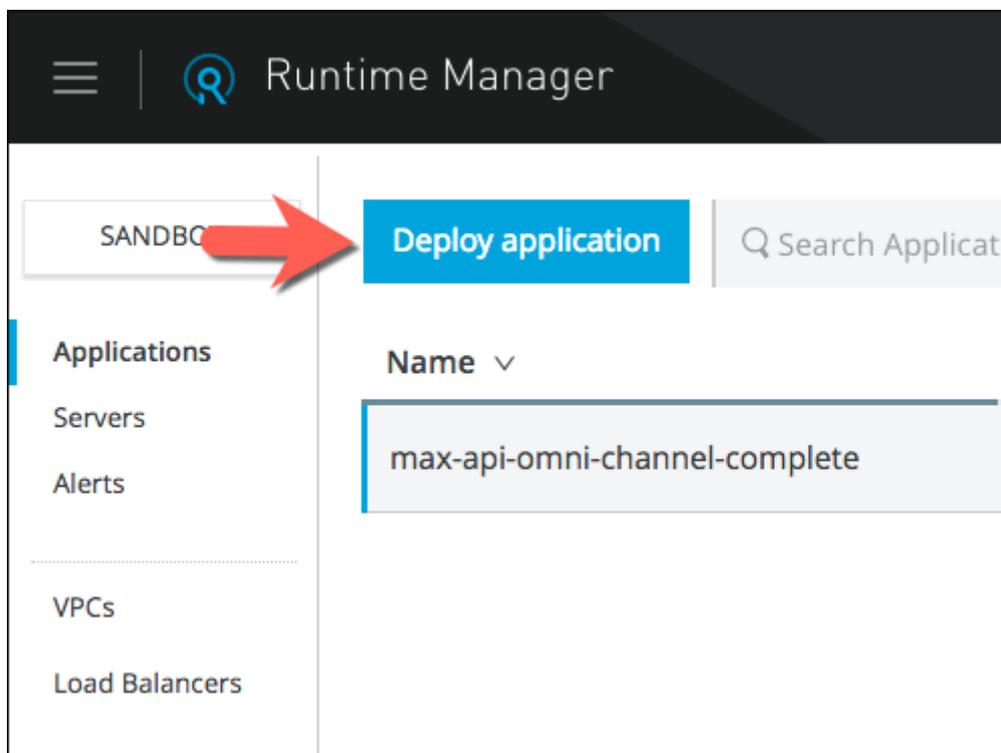
Step 4: Deploy to your Runtime

Now your new on-premises Mule is probably feeling lonely. Let's give it something to run.

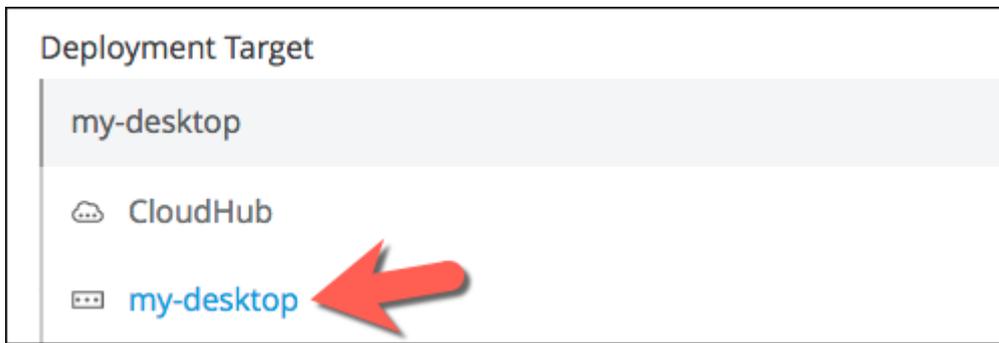
1. If you don't already have it open, re-open [Anypoint Runtime Manager](#) in your browser
2. Click the column containing your <name>-api-omni-channel-complete application, to bring up the management options
 - a. Then click your application bundle (<project name>.zip), which appears in blue



- b. This will let you download a zip file containing your Mule application as a deployable archive to your local machine.
3. Once that's done you're going to deploy that application to your local runtime. To do this, back on the [Applications](#) page, **click** "Deploy Application"



- a. **Type** a new application name for your local application instance
- b. From the box labeled "Deployment Target", **choose** your new on-premises Mule Runtime

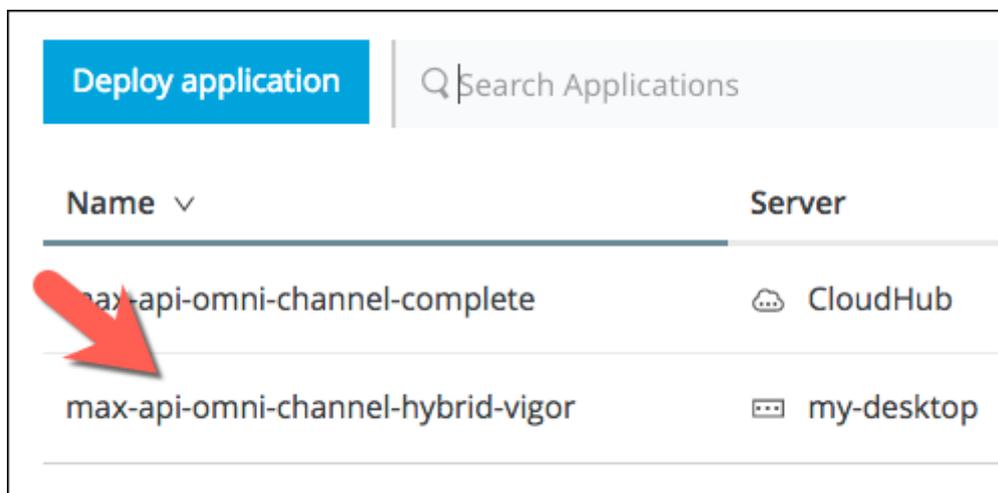


- c. Then on the right, **click** "Choose File" and browse to the location where you just saved your application bundle (e.g., api-omni-channel-complete-v4.0.2.zip)
 - d. Remember to **click** the **Insight** tab and enable **Metadata**. Notice the option "Metadata and Replay" is not available, as on-premises runtimes do not have the ability to store message data for Replay.
4. Now you are ready to **click** the box for "**Deploy Application**"

Step 5: Manage your Application

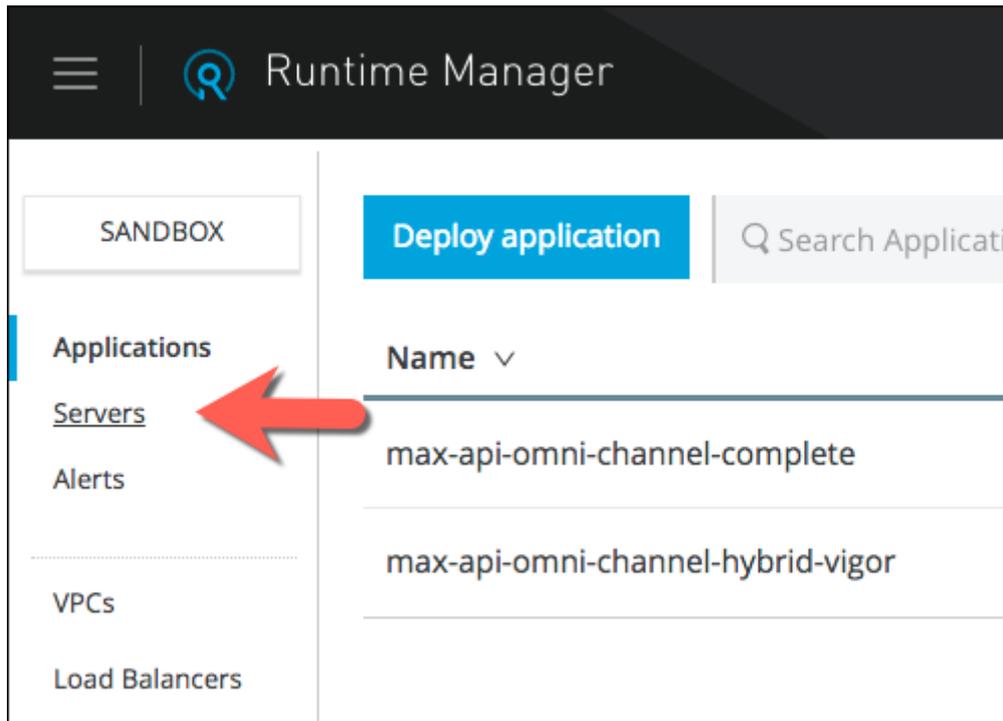
Once your application has started, you can go view and manage it from Runtime Manager, as with the CloudHub example. With on-premises runtimes, you can deploy more than one application to a runtime, or you can allocate more than one runtime per application by using Server Groups and Clusters.

1. Your new hybrid instance of your will show up under "Applications" within [Runtime Manager](#), and can be managed in much the same way as a CloudHub application

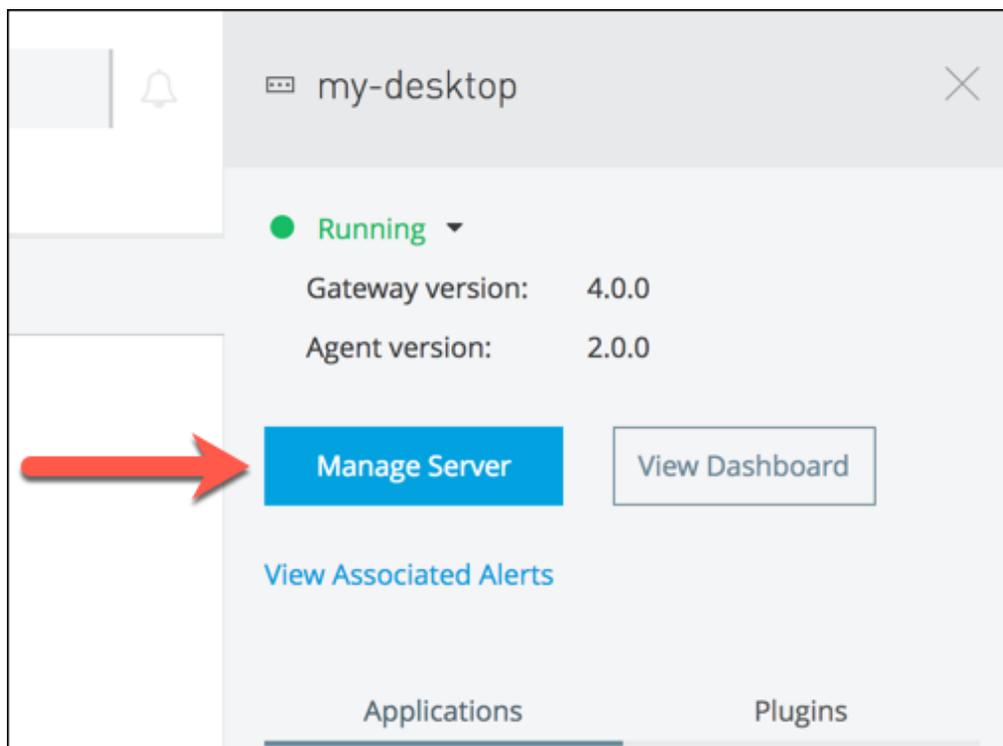


- a. This interface should already be somewhat familiar. Take a moment to note similarities and differences in how you can manage your new on-premises Application compared to one on CloudHub Workers.
- b. Visit the [local API console](#) of your application and submit some results to your `get:/products/search` flow. It should look and behave identically to the API console running on CloudHub.
- c. Go back to [Runtime Manager](#) and look at your new Transactions on the Insights page.

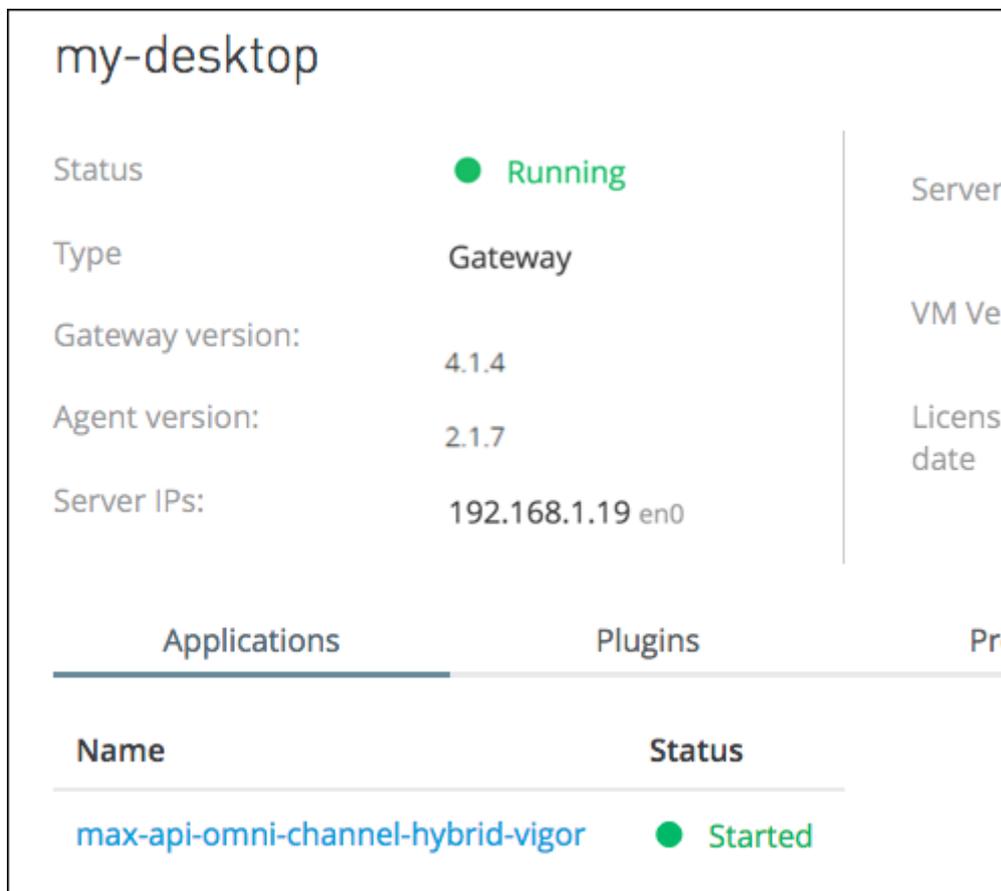
2. Once you've managed your application, you also have the ability to manage the local runtime instance itself. You can do this by visiting the Servers section of [Runtime Manager](#).



3. As with the Application management, you just click on the row containing your Server's name and select Manage Server



4. From this page, you can access information about your on-premises Runtime, as well as configure plugins, set server properties, and view alerts. Notice your deployed Application is also shown below.



my-desktop

Status ● Running

Type Gateway

Gateway version: 4.1.4

Agent version: 2.1.7

Server IPs: 192.168.1.19 en0

Server

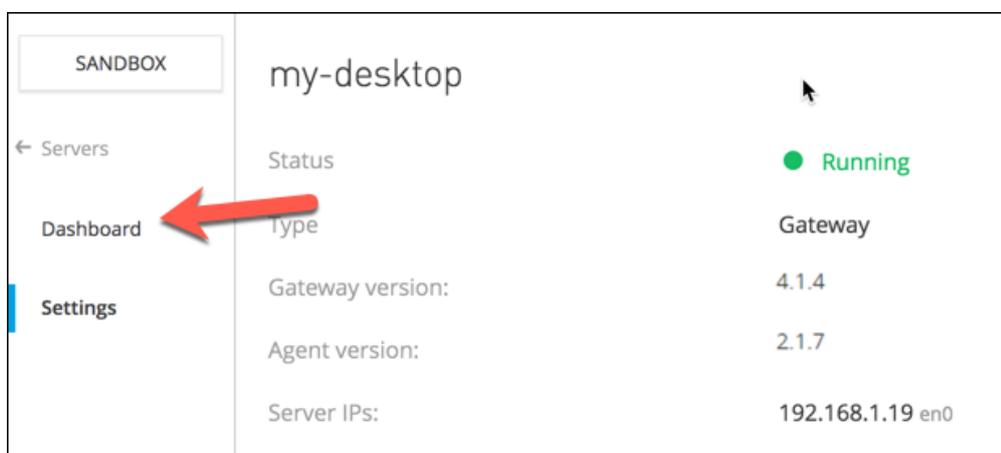
VM Ver

License date

Applications Plugins Pro

Name	Status
max-api-omni-channel-hybrid-vigor	● Started

5. You can view how your new Mule Runtime is performing by visiting the Dashboard on the left



SANDBOX

← Servers

Dashboard ←

Settings

my-desktop

Status ● Running

Type Gateway

Gateway version: 4.1.4

Agent version: 2.1.7

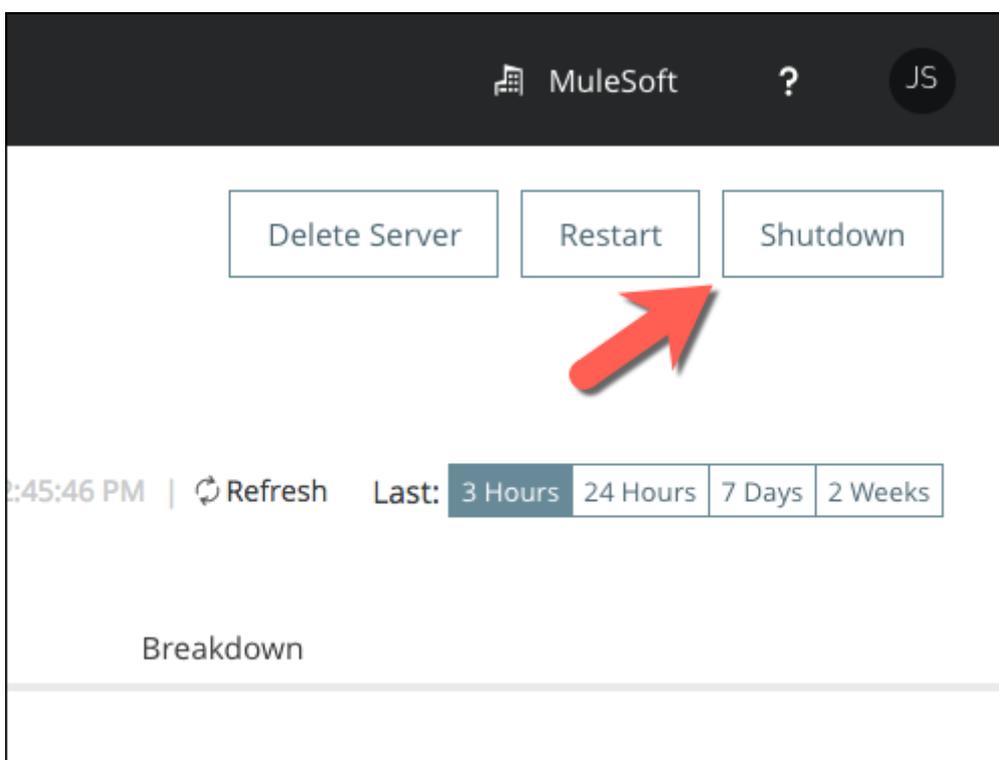
Server IPs: 192.168.1.19 en0

6. The Dashboard shows you metrics about how your on-premises Runtime is performing. Additional detail is available on the "Breakdown" tab.

7. At the bottom of this page, you can also view metrics associated with your Application instance on this Runtime by clicking the circle next to your Application name.

Applications running on my-desktop	
<input type="radio"/> Name	Status
<input type="radio"/> max-api-omni-channel-hybrid-vigor	● Started

8. When you're finished, remember to shutdown your local Mule runtime
 - a. You can take advantage of our hybrid management approach by simply clicking Shutdown on the top of the Server Dashboard



- b. You can also stop your Mule Runtime instance locally from `<MULE_HOME>/bin` by running `./mule stop` or `mule.bat stop`

Summary

In this Lab, within the space of a few minutes, you saw how easy it is to setup an on-premises Mule Runtime. You downloaded, registered, and managed it with minimal overhead. More importantly, you deployed and managed the same Application in CloudHub and on-premises with zero modification and both from the same interface.

Mule Runtime is the engine of Anypoint Platform. Anypoint is the industry's only unified platform that combines data and application integration across legacy systems, SaaS applications, and APIs with hybrid deployment options for maximum flexibility. Connect anything, change everything.

In this lab, we completed these steps

- [Step 1: Download the Runtime](#)
- [Step 2: Register your Runtime](#)
- [Step 3: Start your Runtime](#)
- [Step 4: Deploy to your Runtime](#)
- [Step 5: Manage your Application](#)

If you'd like to learn more

- See [Deploying to Your Own Servers](#) for more information
- See [Managing Servers](#) for more information [Configuring Mule for Different Deployment Scenarios](#) for more information

Congratulations! You have completed Lab 3.

Please proceed to [Lab 3](#)

[Take me to the TOP](#)