



PREV

11. Stream Processing



NEXT

B. Additional Kafka Tools



Appendix A. Installing Kafka on Other Operating Systems

Apache Kafka is primarily a Java application, and therefore should be able to run on any system where you are able to install a JRE. It has, however, been optimized for Linux-based operating systems and so that is where it will perform best. Running on other operating systems you may find bugs that are specific to your OS. For this reason, when using Kafka for development or test purposes on a common desktop operating system, it's a good idea to consider running in a virtual machine that matches your eventual production environment.

Installing on Windows

As of Microsoft Windows 10, there are now two ways that you can run Kafka. The traditional way is using a native Java installation. While you can also use a package manager such as Chocolatey, the package has not been maintained properly and does not download correctly (at the time of publication). Windows 10 users have the option to use the Windows Subsystem for Linux, however. The latter method is highly preferred, as it provides a much simpler setup that more closely matches the typical production environment, and so we will review that first.

Using Windows Subsystem for Linux

If you are running Windows 10, you have the ability to install native Ubuntu support under Windows using Windows Subsystem for Linux (WSL). At the time of publication, Microsoft still considers WSL to be an experimental feature. While it acts similar to a virtual machine, it does not require the resources of a full VM and provides richer integration with the Windows operating system.

In order to install WSL, you should follow the instructions available from the Microsoft Developer Network at the [Bash on Ubuntu on Windows](#) page. Once that is done, you will need to install a Java Development Kit using `apt-get`:

```
$ sudo apt-get install openjdk-7-jre-headless
[sudo] password for username:
Reading package lists... Done
Building dependency tree
Reading state information... Done
[...]
done.
$
```

Once you have completed installing the JDK, you can proceed to install Apache Kafka using the instructions in [Chapter 2](#).

Using Native Java

For older versions of Windows, or if you prefer not to use the WSL environment, you can run Kafka natively with a Java environment for Windows. Be aware, however, that this can introduce you to bugs that are specific to the Windows environment.

These bugs may not get the attention in the Apache Kafka development community that similar problems with Linux might run into because of the dearth of developers running Kafka on Windows.

Before installing Zookeeper and Kafka, you must have a Java environment set up. The recommendation is that you install the latest version of Oracle Java 8, which can be found on the [Oracle Java SE Download page](http://www.oracle.com/technetwork/java/javase/downloads/index.html)

(<http://www.oracle.com/technetwork/java/javase/downloads/index.html>). Download a full JDK package, so that you have all the Java tools available, and follow the instructions for installation.

BE CAREFUL WITH PATHS

When installing Java and Kafka, it is highly recommended that you stick to installation paths that do not contain spaces. While Windows allows spaces in paths, applications that are designed to run in Unix environments are not set up this way and specifying paths will be difficult. When installing Java, make sure to set the installation path with this in mind. For example, if installing JDK 1.8 update 121, a good choice would be to use the path `C:\Java\jdk1.8.0_121`.

Once Java is installed, you should set up the environment variables so that it can be used. This is done in the Control Panel for Windows, though the exact location will depend on your version of the OS. In Windows 10, you must select *System and Security*, then *System*, and then click *Change settings* under the *Computer name, domain, and workgroup settings* section. This will open the System Properties window, where you can select the *Advanced* tab and then finally click the *Environment Variables...* button. Use this section to add a new user variable that is named `JAVA_HOME` and is set to the path where you installed Java. Then edit the system variable named `Path` and add a new entry which is `%JAVA_HOME%\bin`. Save these settings and exit out of the Control Panel.



Figure A-1. Adding the `JAVA_HOME` variable

Now you can proceed to install Apache Kafka. This includes Zookeeper in the installation, so that you do not have to install it separately. The current release of Kafka can be downloaded at <http://kafka.apache.org/downloads.html> (<http://kafka.apache.org/downloads.html>). At publication time, that version is 0.10.1.0 running under Scala version 2.11.0. The downloaded file will be GZip compressed and packaged with the `tar` utility, so you will need to use a Windows application such as 8Zip to uncompress it. Similar to installing on Linux, you must choose a directory to extract Kafka into. For this example, we will assume Kafka is extracted into `C:\kafka_2.11-0.10.1.0`.

Running Zookeeper and Kafka under windows is a little different, as you must use the batch files designed for Windows rather than the shell scripts for other platforms. These batch files also do not support backgrounding the application, so you will need a separate shell for each application. First, start Zookeeper:

```
PS C:\> cd kafka_2.11-0.10.2.0
PS C:\kafka_2.11-0.10.2.0> bin/windows/zookeeper-server-start.bat C:\kafka_2.11-0.10.2.0\config\zookeeper.properties
[2017-04-26 16:41:51,529] INFO Reading configuration from: C:\kafka_2.11-0.10.2.0\config\zookeeper.properties (org.apa
[...]
```

Once Zookeeper is running, you can open another window to start Kafka:

```
PS C:\> cd kafka_2.11-0.10.2.0
PS C:\kafka_2.11-0.10.2.0> .\bin\windows\kafka-server-start.bat C:\kafka_2.11-0.10.2.0\config\server.properties
[2017-04-26 16:45:19,804] INFO KafkaConfig values:
[...]
[2017-04-26 16:45:20,697] INFO Kafka version : 0.10.2.0 (org.apache.kafka.common.utils.AppInfoParser)
[2017-04-26 16:45:20,706] INFO Kafka commitId : 576d93a8dc0cf421 (org.apache.kafka.common.utils.AppInfoParser)
[2017-04-26 16:45:20,717] INFO [Kafka Server 0], started (kafka.server.KafkaServer)
```

Installing on OS X

The Apple OS X operating system runs on Darwin, a Unix operating system that is derived, in part, from FreeBSD. This means that many of the expectations of running on a Unix OS hold true, and installing applications designed for Unix, like Apache Kafka, is not too difficult. You can either keep the installation simple by using a package manager (like Homebrew), or you can install Java and Kafka manually for greater control over versions.

Using Homebrew

If you have already installed [Homebrew](#) for OS X, you can use it to install Kafka in one step. This will assure that you have Java installed first, and it will then install Apache Kafka 0.10.2.0 (as of publication).

If you have not yet installed Homebrew, do that first by following the directions on the [installation page \(http://docs.brew.sh/Installation.html\)](http://docs.brew.sh/Installation.html). Then you will be able to install Kafka itself. The Homebrew package manager will assure that you have all the dependencies installed first, including Java.

```
$ brew install kafka
==> Installing kafka dependency: zookeeper
[...]
==> Summary
/usr/local/Cellar/kafka/0.10.2.0: 132 files, 37.2MB
$
```

Homebrew will install Kafka under /usr/local/Cellar, however the files will be linked into other directories:

- Binaries and scripts will be in /usr/local/bin
- Kafka configurations will be in /usr/local/etc/kafka
- Zookeeper configuration will be in /usr/local/etc/zookeeper
- The log.dirs config (the location for Kafka data) will be set to /usr/local/var/lib/kafka-logs

After installation is complete, you can start Zookeeper and Kafka (this example starts Kafka in the foreground):

```
$ /usr/local/bin/zkServer start
JMX enabled by default
Using config: /usr/local/etc/zookeeper/zoo.cfg
Starting zookeeper ... STARTED
$ kafka-server-start.sh /usr/local/etc/kafka/server.properties
[...]
[2017-02-09 20:48:22,485] INFO [Kafka Server 0], started (kafka.server.KafkaServer)
```

Installing Manually

Similar to a manual installation for the Windows operating system, when installing Kafka on OS X you must first install a Java Development Kit. The same [Oracle Java SE Download page \(http://www.oracle.com/technetwork/java/javase/downloads/index.html\)](http://www.oracle.com/technetwork/java/javase/downloads/index.html) can be used to get the proper version for OS X. You can then download Apache Kafka, similar to Windows again. For this example, we will assume that the Kafka download is expanded into the /usr/local/kafka_2.11-0.10.2.0 directory.

Starting Zookeeper and Kafka looks just like starting them when using Linux, though you will need to make sure your JAVA_HOME directory is set first:

```
$ export JAVA_HOME="/usr/libexec/java_home"
$ echo $JAVA_HOME
/Library/Java/JavaVirtualMachines/jdk1.8.0._131.jdk/Contents/Home
$ /usr/local/kafka_2.11-0.10.2.0/bin/zookeeper-server-start.sh -daemon /usr/local/kafka_2.11-0.10.2.0/config/zookeeper
$ /usr/local/kafka_2.11-0.10.2.0/bin/kafka-server-start.sh /usr/local/etc/kafka/server.properties
[2017-04-26 16:45:19,804] INFO KafkaConfig values:
[...]
[2017-04-26 16:45:20,697] INFO Kafka version : 0.10.2.0 (org.apache.kafka.common.utils.AppInfoParser)
[2017-04-26 16:45:20,706] INFO Kafka commitId : 576d93a8dc0cf421 (org.apache.kafka.common.utils.AppInfoParser)
[2017-04-26 16:45:20,717] INFO [Kafka Server 0], started (kafka.server.KafkaServer)
```



◀ PREV
11. Stream Processing

NEXT ▶
B. Additional Kafka Tools