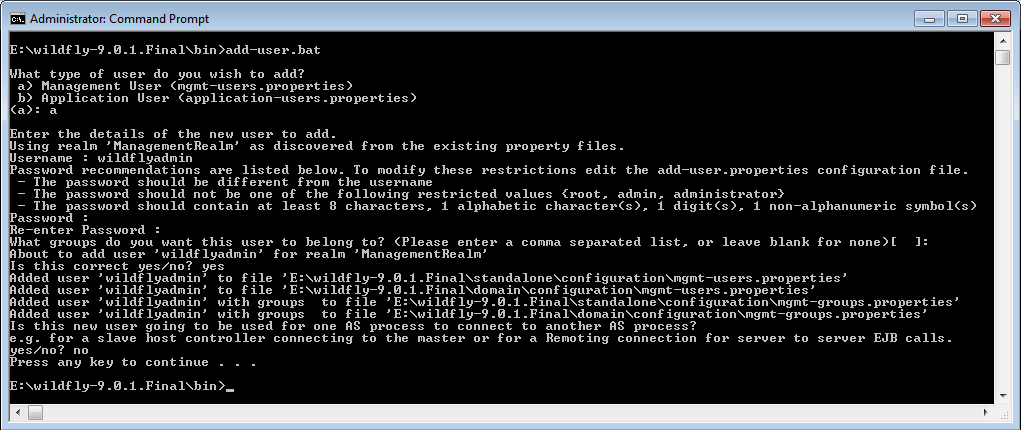
WildFly installation:

Requirements:

* Java SE 7 or later (version that was tested and also recommended is: 1.7.0\_71, 64-bit)

The following items are considered as prerequisites before attempting to install WildFly:

1. Download WildFly (formerly known as JBoss AS) version 10.1.0.Final available on <http://wildfly.org/downloads/>
2. Unzip the distribution in a location of your choice, here-after called ${jboss\_path}
3. Run the add-user command from ${jboss\_path}\bin and create a management user running add-user command, like in the image below:



**NOTE**: The MySQL scripts included to set up the DB include configuration for username 'admin'.

The user for webservices exposed to the Node has to be created as well. In current configuration is **ETX-NODE-SERVICE-USER / 5Uf6.O!TQ~~j;IF** and group **NODE-SERVICE-ROLE**

Sample files located in the source dir: **docs/open\_source\_version/wildfly\_sample\_conf\_files**

1. From ${jboss\_path}\bin, start the server by running the standalone.bat command
2. Access <http://localhost:9990/> and click on the Administration Console link. On the new window opened, authenticate your-self as the user created at step 3. A successful authentication means a successful JBoss installation. For any other problems related with the installation please consult documentation section at: <https://docs.jboss.org/author/display/WFLY9/Documentation>

WildFly setup:

1. MySQL Datasource configuration
2. Navigate to ${jboss\_path}\modules and create the folder: system\layers\base\com\mysql\main
3. Create a file called module.xml and add the following inside:

<module xmlns="urn:jboss:module:1.3" name="com.mysql">

<resources>

<resource-root path="mysql-connector-java-5.1.46-bin.jar"/>

</resources>

<dependencies>

<module name="javax.api"/>

<module name="javax.transaction.api"/>

<module name="javax.servlet.api" optional="true"/>

</dependencies>

</module>

1. Download the last JDBC Driver from MySQL site (version 5.1. 46 at this moment) - go to <http://dev.mysql.com/downloads/connector/j/> and choose “Platform Independent” in the dropdown for selecting the platform.
2. Unzip it and copy from inside the file **mysql-connector-java-5.1.46-bin.jar** to ${jboss\_path}\modules\ system\layers\base\com\mysql\main
3. Navigate to ${jboss\_path}\standalone\configuration\standalone.xml and inside this file search for <subsystem xmlns="urn:jboss:domain:datasources:4.0">
4. In the datasource subsystem, in the datasources-> drivers section, add the following:

<driver name="mySql" module="com.mysql">

<xa-datasource-class>com.mysql.jdbc.jdbc2.optional.MysqlXADataSource</xa-datasource-class>

</driver>

1. In the datasource subsystem, in the datasources section, define a new datasource linked to the newly added driver like below where you have to replace with your data:

<xa-datasource jndi-name="java:/jdbc/EtxWebDb" pool-name="EtxWebDb" enabled="true" use-java-context="true">

<xa-datasource-property name="URL">

jdbc:mysql://localhost:3306/etxweb

</xa-datasource-property>

<driver>mySql</driver>

<xa-pool>

<min-pool-size>0</min-pool-size>

<max-pool-size>20</max-pool-size>

</xa-pool>

<security>

<user-name>etxweb</user-name>

<password>etxweb</password>

</security>

<validation>

<valid-connection-checker class-name="org.jboss.jca.adapters.jdbc.extensions.mysql.MySQLValidConnectionChecker"/>

<background-validation>true</background-validation>

<exception-sorter class-name="org.jboss.jca.adapters.jdbc.extensions.mysql.MySQLExceptionSorter"/>

</validation>

</xa-datasource>

1. Install the JMS queues
   1. Navigate to ${jboss\_path}\standalone\configuration and open for edition the file standalone.xml
   2. Add the following extension to the server->extensions section:

<extension module="org.jboss.as.messaging"/>

* 1. In the server->profile, add the configuration for the messaging subsystem that defines the hornet server:

<subsystem xmlns="urn:jboss:domain:messaging:3.0">

<hornetq-server>

<journal-file-size>102400</journal-file-size>

<connectors>

<in-vm-connector name="in-vm" server-id="0"/>

</connectors>

<acceptors>

<in-vm-acceptor name="in-vm" server-id="0"/>

</acceptors>

<security-settings>

<security-setting match="#">

<permission type="send" roles="guest"/>

<permission type="consume" roles="guest"/>

<permission type="createDurableQueue" roles="guest"/>

<permission type="deleteDurableQueue" roles="guest"/>

<permission type="createNonDurableQueue" roles="guest"/>

<permission type="deleteNonDurableQueue" roles="guest"/>

</security-setting>

</security-settings>

<address-settings>

<address-setting match="#">

<dead-letter-address>jms.queue.DeadLetterJMSQueue</dead-letter-address>

<expiry-address>jms.queue.ExpiryQueue</expiry-address>

<max-size-bytes>10485760</max-size-bytes>

<page-size-bytes>2097152</page-size-bytes>

<message-counter-history-day-limit>10</message-counter-history-day-limit>

</address-setting>

</address-settings>

<jms-connection-factories>

<connection-factory name="InVmConnectionFactory">

<connectors>

<connector-ref connector-name="in-vm"/>

</connectors>

<entries>

<entry name="java:/ConnectionFactory"/>

</entries>

</connection-factory>

<pooled-connection-factory name="hornetq-ra">

<transaction mode="xa"/>

<connectors>

<connector-ref connector-name="in-vm"/>

</connectors>

<entries>

<entry name="java:/JmsXA"/>

<entry name="java:jboss/DefaultJMSConnectionFactory"/>

<entry name="jms/EtxWebConnectionFactory"/>

</entries>

<consumer-window-size>0</consumer-window-size>

</pooled-connection-factory>

</jms-connection-factories>

</hornetq-server>

</subsystem>

1. Install the JMS and mail queues:

You can create them manually taking as as an example, or just directly use the file:

**docs/open\_source\_version/standalone-etx-web.xml**

for your wildfly server configuration