

# Installing SwiftRiver on Ubuntu or Debian [old guide]

## On this page

- [Install the required packages](#)
- [Install the API](#)
- [Create the Database](#)
- [Configure Solr](#)

This guide applies on the following platforms:

- Ubuntu 10.04 LTS +
- Debian 6.0 (Squeeze)

The SwiftRiver install process is:

- Install the required packages
- Set up a MySQL database
- Set up Solr
- Set up Tomcat
- Create the application directories
- Build and deploy
- Create the database tables
- Set up the UI Client

## Install the required packages

The following software packages must be installed on your system:

- Java 1.5 or greater: preferably Java 1.6 (also known as Java 6)
- [MySQL Server](#) version 5.1 or greater
- [Apache Solr](#)
- [Apache Tomcat](#) (version 6.0 or greater) or other [servlet container](#)
- [Apache HTTP server](#) with `mod_rewrite` enabled
- [PHP](#) version 5.3 or greater
- [Python](#) version 2.6 or version 2.7
- [RabbitMQ](#) version 3.0 or greater
- [Sendmail](#) or other [Mail Transfer Agent](#)

## Install the API

- Download and extract the API distribution
- Stop your servlet container
- Copy the API war to your webapps directory
- Copy the sample context configuration file - `config/swiftriver-api.xml` - to `<TOMCAT_HOME>/conf/Catalina/localhost` and modify the default configuration parameters. The following is an example of the modified file:

```

<Context docBase="" path="/swiftriver-api" >
  <!-- SwiftRiver Database configuration -->
  <Resource auth="Container" driverClassName="com.mysql.jdbc.Driver"
    maxActive="8" maxIdle="4"
    name="jdbc/SwiftRiverDB"
    type="javax.sql.DataSource"
    url="jdbc:mysql://localhost/swiftriver?zeroDateTimeBehavior=convertToNull"
    username="swiftriver"
    password="swiftriver"/>

  <!-- Encryption Key -->
  <Environment name="encryptionKey" type="java.lang.String" value="2344228477#97{7&6&gt;82}/>

  <!-- MQ Properties -->
  <Environment name="mqHost" type="java.lang.String" value="localhost"/>
  <Environment name="mqUser" type="java.lang.String" value="guest"/>
  <Environment name="mqPass" type="java.lang.String" value="guest"/>

  <!-- HTTP Solr Server -->
  <Environment name="solr/serverURL" type="java.lang.String" value="http://localhost:8080/solr"/>

  <!-- Location of Solr indexing properties file -->
  <Environment name="solr/indexerProperties" type="java.lang.String"
value="/Users/ekala/indexer.properties" />

  <!-- Keys for the indexer properties file -->
  <Environment name="indexer/lastDropIdPropKey" type="java.lang.String" value="indexer.lastDropId" />
  <Environment name="indexer/batchSizePropKey" type="java.lang.String" value="indexer.batchSize" />
  <Environment name="indexer/runInterval" type="java.lang.String" value="30000"/>

  <!-- Default authentication scheme. Possible values are:
    database
    crowdmapid

    'database' is the default
  -->
  <Environment name="authSchemeName" type="java.lang.String" value="database"/>

  <!-- CrowdmapID API URL e.g. https://example.com/ -->
  <Environment name="crowdmapid/serverURL" type="java.lang.String"
value="https://crowdmapid.com/api"/>
  <Environment name="crowdmapid/apiKey" type="java.lang.String" value="" />
  <Environment name="crowdmapid/apiKeyParamName" type="java.lang.String" value="api_secret"/>

  <!-- Mail configuration -->
  <Environment name="mail/host" type="java.lang.String" value="localhost" />
  <Environment name="mail/senderAddress" type="java.lang.String" value="no-reply@swiftriver.dev"/>
  <Environment name="mail/resetPasswordUrl" type="java.lang.String"
value="http://swiftriver.dev/login/reset_password"/>
  <Environment name="mail/activateAccountUrl" type="java.lang.String"
value="http://swiftriver.dev/login/activate"/>

</Context>

```

An explanation for each of the configuration parameters is provided below:

Parameter	Description
mqHost	The host running the RabbitMQ server
mqUser	User to connect to RabbitMQ
mqPassword	Password for the user used to connect to RabbitMQ
solr/serverURL	URL of your Solr server
solr/indexerProperties	Location of the properties file for the indexer - a background process that periodically updates Solr with the new drops
indexer/lastDropIDPropKey	The property key that specifies the ID of the last drop to be posted to Solr. This value serves as the reference point for fetching new drops
indexer/batchSizePropKey	The property key that specifies the maximum no. of drops to post to Solr during each run
indexer/runInterval	The property key that specifies how often (in ms) the indexer should check for new drops and update Solr
authSchemeName	Name of the authentication scheme. The possible values are <code>database</code> and <code>crowdmapid</code>
crowdmapid/serverURL	URL of the CrowdmapID deployment
crowdmapid/apiKey	API key for authenticating requests to the CrowdmapID deployment specified in <code>crowdmapid/serverURL</code>
crowdmapid/apiKeyParamName	Name of the request parameter used to specify the api key when submitting a request to the CrowdmapID deployment
mail/host	Name/IP address of the mail server
mail/senderAddress	Email address to be used when sending out emails
mail/resetPasswordUrl	URL to be used when sending the password reset link.
mail/activateAccountUrl	URL to be used for sending the account activation link; when a new account is created.

## Create the Database

Create a new database called `swiftriver`

```
CREATE DATABASE swiftriver CHARACTER SET utf8 COLLATE utf8_unicode_ci;
```

Create a database user called `swiftriver` and grant them all the privileges on the `swiftriver` database.

```
GRANT ALL PRIVILEGES ON swiftriver.* TO swiftriver@'localhost' IDENTIFIED BY 'swiftriver';
```

Create the database tables by running the schema setup script - `install/schema.sql`

```
mysql swiftriver -u swiftriver -p < install/schema.sql
```

## Configure Solr

Copy `solr/solr.xml` (from the API distribution) to your `SOLR_HOME`. The default `solr.xml` file looks like this:

```
<solr persistent="true">  
  <cores adminPath="/admin/cores" defaultCoreName="swiftriver" host="${host:}" hostPort="${jetty.port:}">  
    <core name="swiftriver" instanceDir="swiftriver" dataDir="data"/>  
  </cores>  
</solr>
```

Create a directory for the `swiftriver` core in `SOLR_HOME`

```
mkdir <SOLR_HOME>/swiftriver
```

Copy the existing `conf` directory to the core director we have created above

```
cp -rf <SOLR_HOME>/conf <SOLR_HOME>/swiftriver
```

Copy `solr/schema.xml` and `solr/solrconfig.xml` to the `conf` directory of the `swiftriver` core:

```
cp solr/schema.xml <SOLR_HOME>/swiftriver/conf  
cp solr/solrconfig.xml <SOLR_HOME>/swiftriver/conf
```

Configure Background Processes

Start your servlet container