Installation and Configuration Guide for JIRA Software 7.10.2 in CentOS 7

1.0 GENERAL INFORMATION

1.0 GENERAL INFORMATION

1.1 System Overview

The JIRA Software is one of the most effective project management and collaboration-equipment systems of today. It brings a multitude of unique features and functions that can be used for internal and external project management, tracking and software development.

1.2 Organization of the Manual

The user's manual consists of four sections: General Information, System Summary, Getting Started, Data Backup and Data Restore.

General Information section explains in general terms the system and the purpose for which it is intended.

System Summary section provides a general overview of the system. The summary outlines the uses of the system's hardware and software requirements, system's configuration, user access levels and system's behaviour in case of any contingencies.

Getting Started section explains how to get JIRA and install it on the device. The section presents JIRA and MySQL Installation and Setup.

Data Backup section provides a detailed explanation on backing up JIRA Software Data and Data Directory.

Data Restore section explains the restoring procedure after all existing data is deleted.

2.0 SYSTEM SUMMARY

2.0 SYSTEM SUMMARY

This section provides a general overview of the system. The summary outlines the uses of the system's hardware and software requirements, system's configuration, user access levels and system's behavior in case of any contingencies.

2.1 System Configuration

JIRA can operate in two modes:

Public: Anyone can sign themselves up with self-registration and create issues (within the bounds of your JIRA system's permissions).

Private: Useful for internal issue-tracking systems where you do not want public users to login. Self-signup is disabled; only Administrators can create new users.

2.2 User Access Levels

In the Jira Software documentation, most configuration options are described as being restricted to either Jira administrators, project administrators, or board administrators.

- A Jira administrator is a user with the **Administer Jira** global permission.
- A project administrator is a user with the **Administer projects** project permission for a project.
  By default, the 'Administer projects' permission is assigned to the 'administrators' group (via the Administrators role) for projects.
  Additionally, to perform sprint-related actions, users need the 'Manage sprints' permission for all projects in the origin board — the origin board being the board in which the sprint was originally created.
- A board administrator is a user that has been added to the **Administrators** for a board.
  By default, the administrator of a board includes the person who created it.

2.3 Contingencies

Copy MySQL Connector JBDC before running JIRA Software Setup configuration

The MySQL server is set to block any request from other others IP address except localhost

The extraction of attachment backup file after restoration process the file will not readable to the JIRA Software

3.0 GETTING STARTED

3.0 GETTING STARTED

Getting Started section explains how to get JIRA 7.10.2 software and install it on the device.

3.1 JIRA Software 7.10.2 Installation

3.1.1 Download JIRA 7.10.2

The latest installation version can be downloaded from:

```
$ wget
https://www.atlassian.com/software/jira/downloads/binar
y/atlassian-jira-software-7.10.2-x64.bin
```

3.1.2 Run the Installer

1. Change to the directory where the file is downloaded and execute this command to change the permission to execute the installer;

```
$ chmod a+x atlassian-jira-xxxx-x.x.x.x-x64.bin
```

2. Run this command to execute the JIRA installer;

```
$ ./atlassian-jira-xxxx-x.x.x.x-x64.bin
```

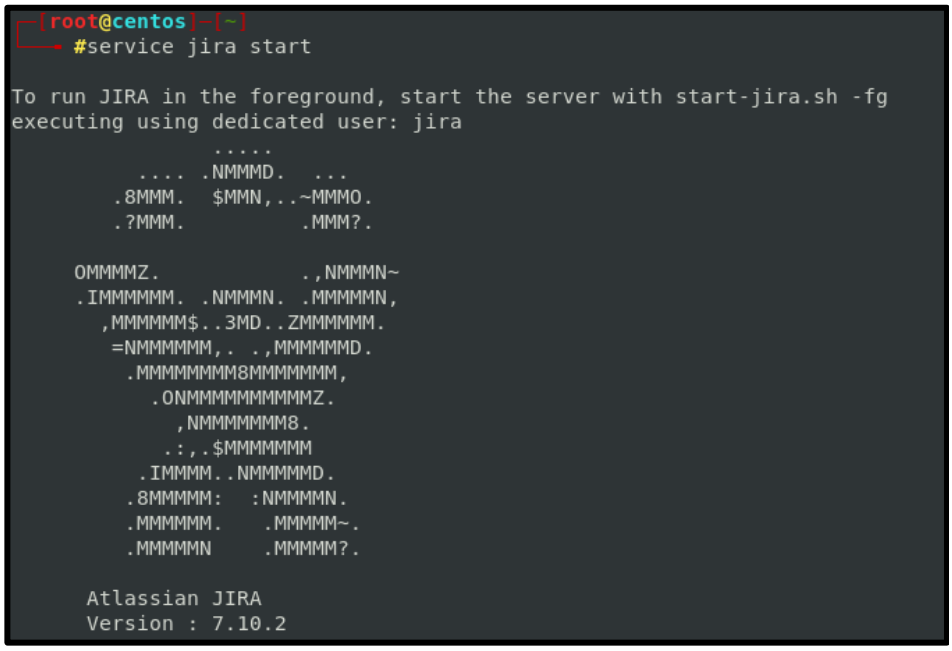
3. The installer will require the following prompts to install JIRA 7.10.2:
    - **Installer Type**
        - Choose Option 2 (custom) for the most control
    - **Destination Directory**
        - This is where JIRA will be installed
    - **Home Directory**
        - This is where JIRA's data - logs, search indexes and files – will be stored.
    - **TCP Ports**

- These are the HTTP connector and control ports JIRA will run on. Stick with the default unless you are running another application on the same port.

- **Install as Service**
  - This option is only available if you run the installer as sudo.

3.1.3 Starting the Software

Upon installation completion, run this command to start JIRA Socan:



3.1.4 JIRA 7.10.2 Web Setup

Once installation is completed, proceed to http://localhost:8080 in your browser to begin the setup process. (Replace 8080 if a different port is chosen during installation)

3.2 MySQL 5.7 Installation

3.2.1 Default Repository in CentOS 7

CentOS 7 prefers MariaDB, a fork of MySQL managed by the original MySQL developers and is designed as a replacement for MySQL. If `$ yum install MySQL` is opened on CentOS7, it is MariaDB that is installed rather than MySQL.

3.2.2 Download MySQL 5.7 Repository

To download the MySQL version for this installation, use:

```
$ wget https://dev.mysql.com/get/mysql57-community-release-el7-9.noarch.rpm
```

3.2.3 Verify Integrity

Once the RPM is saved, verify the integrity of the downloaded file:

```
$ md5sum mysql57-community-release-el7-9.noarch.rpm
```

The output of the MD5 value should be similar with the MD5 value shown below;

```
1a29601dc380ef2c7bc25e2a0e25d31 mysql57-community-release-el7-9.noarch.rpm
```

Differences in MD5 value might indicate if file is corrupted while transferring.

3.2.4 Install MySQL Package

To add new yum repositories:

```
$ sudo rpm -ivh mysql57-community-release-el7-9.noarch.rpm
```

Install MySQL 5.7:

```
$ sudo yum install mysql-server
```

3.2.5 Start MySQL

Start MySQL daemon:

```
$ sudo systemctl start mysqld
```

The output should contain - Active: active (running)- once the status of MySQL is successfully opened.

3.2.6 Get MySQL Temporary Password

Temporary password for MySQL root user is generated with every installation. Password is located in mysqld.log. To retrieve the password:

```
$ sudo grep 'temporary password' /var/log/mysqld.log
```

3.2.7 Change MySQL Password Policy

To change the temporary MySQL password, the user is required to create a strong password:

```
ERROR 1819 (HY000): Your password does not satisfy the
current policy requirements
```

To prevent that, a new policy needs to be set;

1. Enter MySQL

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

2. Change the password policy

   ```
   mysql> SET GLOBAL validate_password_policy = low;
   ```

3. Successful change of policy will result in the prompt below:

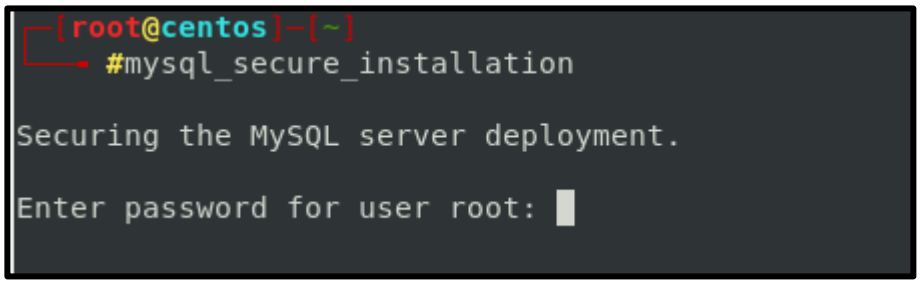
```
Query OK, 0 rows affected (0.00 sec)
```

3.2.8 Configure MySQL Password

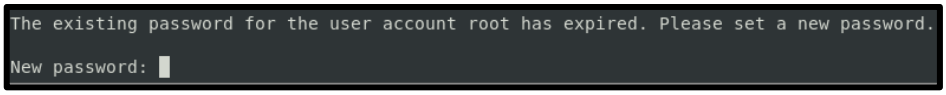
Run MySQL security script:

```
$ sudo mysql_secure_installation
```

User will be prompted to enter the current password (the temporary password):



After entering the current password, user will be prompted to enter a new password for the root user account. User will be prompted to confirm the password change with re-entering the new password:



Following the prompt to change the password along with other configurations:

- New password for root account
- The strength of the password
- Remove anonymous users
- Disallow root login remotely
- Remove default database “test”
- Reload privilege database

3.2.9 Test MySQL

The installation above can be verified by connecting with the mysqladmin tools; a client that executes administrative commands. The following commands connect to MySQL as **root** (-u root), prompt for password (-p), and return.

```
$ mysqladmin -u root -p version
```

Below is the output of the installed version MySQL:

```
[root@centos]-[~]
  #mysqladmin -u root -p version
Enter password:
mysqladmin  Ver 8.42 Distrib 5.7.25, for Linux on x86_64
Copyright (c) 2000, 2019, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Server version          5.7.25
Protocol version        10
Connection              Localhost via UNIX socket
UNIX socket             /var/lib/mysql/mysql.sock
Uptime:                 1 hour 34 min 37 sec

Threads: 1  Questions: 19  Slow queries: 0  Opens: 114  Flush tables: 1  Open tables: 1
07  Queries per second avg: 0.003
```

3.3 Creating and Configuring MySQL 5.7 Database

1. Create a database user in which JIRA will connect as, (e.g **jiradbuser**):

   ```
   GRANT ALL PRIVILEGES ON *.* TO 'jiradbuser@'localhost'
   IDENTIFIED BY Centos1234';
   ```

   *please note the ***Centos1234*** is just an example of a password

2. Create a database for JIRA to store issues, (e.g **jiradb**):

   The database must have a character set of UTF8. To set it, enter the following within the MySQL command client:

   ```
   CREATE DATABASE jiradb CHARACTER SET utf8mb4 COLLATE
   utf8mb4_bin;
   ```

3. Ensure the user has the permission to connect to the database and permission to create and populate tables. To provide these permissions, enter the following commands:

   For MySQL 5.7.0 to 5.7.5:

   ```
   GRANT
   SELECT,INSERT,UPDATE,DELETE,CREATE,DROP,ALTER,INDEX
   on <JIRADB>.* TO 'jiradbuser'@' localhost' IDENTIFIED BY
   'Centos1234';
   flush privileges;
   ```

   For MySQL 5.6.7 and later (must also include the REFERENCES permission):

   ```
   GRANT
   SELECT,INSERT,UPDATE,DELETE,CREATE,DROP,REFERENCES,A
   LTER,INDEX on <JIRADB>.* TO 'jiradbuser'@'localhost' IDENTIFIED
   BY ' Centos1234';
   flush privileges;
   ```

   ***please note that username, server IP and password is just an example**

4. Edit the my.cnf file in MySQL Server Directory in /etc/my.cnf (for detailed instructions on editing these files, check here: https://dev.mysql.com/doc/refman/5.6/en/option-files.html).

5. In my.cnf file, locate the [myssqld] section in the file, and add/modify the following parameters:

   a. Set the default storage engine to InnoDB:

      ```
      default-storage-engine=INNODB
      ```

b. Specify the character set used by the database server:

```
character_set_server=utf8mb4
```

c. Set the default row format to DYNAMIC:

```
innodb_default_row_format=DYNAMIC
```

d. Enable the larger prefix:

```
innodb_large_prefix=ON
```

e. Set the InnoDB file format to Barracuda:

```
innodb_file_format=Barracuda
```

f. Specify the value of noodb_log_file_size to be at least 2G:

```
innodb_log_file_size=2G
```

g. Ensure the sql_mode parameter does not specify NO_AUTO_VALUE_ON_ZERO:
Remove (//) if it exists

```
sql_mode = NO_AUTO_VALUE_ON_ZERO
```

6. Restart the MySQL server for the changes to take effect.
Run the following command to restart the MySQL server:

```
service mysqld stop
```

***then, run the same command, replacing 'stop' with 'start'**

3.4 Adding JDBC Driver for MySQL 5.7

1. Copy MySQL JDBC driver to JIRA Installation Directory

   ```
   $ wget https://dev.mysql.com/get/Downloads/Connector-J/mysql-connector-java-5.1.47.zip
   ```

2. Extract file using Unzip

   ```
   $ unzip mysql-connector-java-5.1.47.zip
   ```

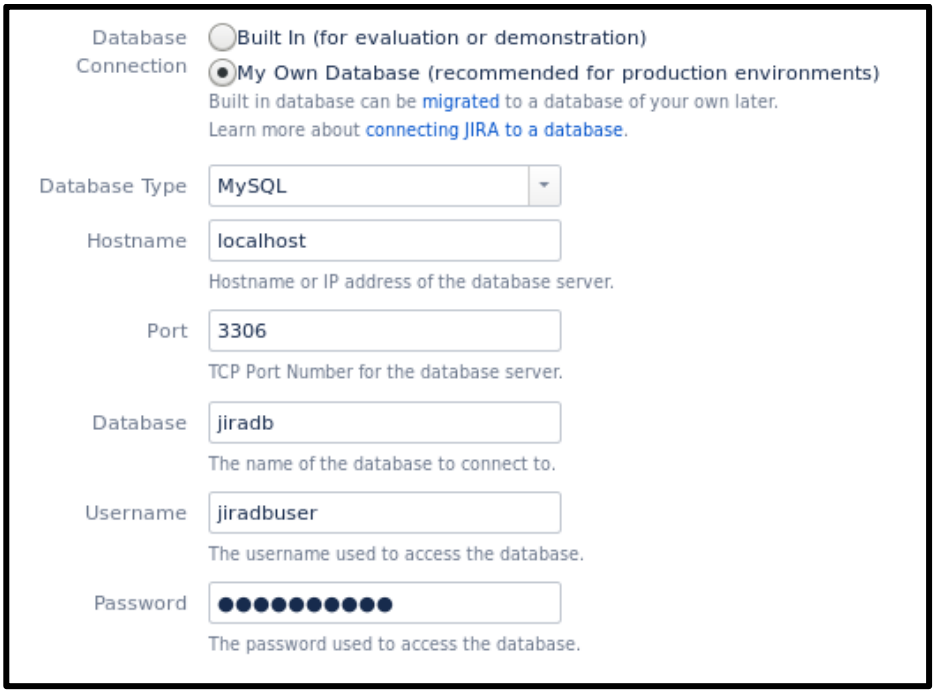
3. Copy the driver to the following directory:

   ```
   $ cp -r /mysql-connector-java-5.1.47/* /opt/atlassian/jira/lib
   ```

3.5 JIRA 7.10.2 Software Setup

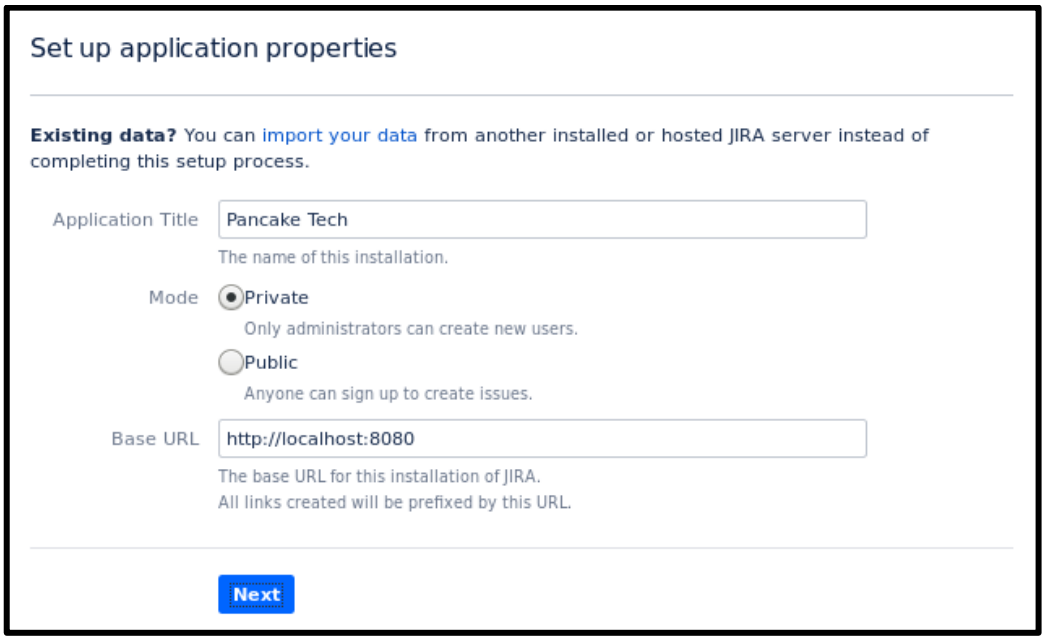
1. Open any installed browser in the machine and enter JIRA address - http://localhsot:8080- in the search bar.

2. Once the browser loads JIRA Software setup page, select "**I will set it up myself**" to deploy a production or testing purpose.
3. Select "My Own Database (recommended for production environments)" and the setup will further prompt for more information that is needed for the setup:
    a. Database Type used (**MySQL**)
    b. Hostname of the MySQL server (**localhost**)
    c. Port used by MySQL (**3306**)
    d. Database Name in MySQL (**jiradb**)
    e. Database Username (**jiradbuser**)
    f. Database Password (**Centos1234**)

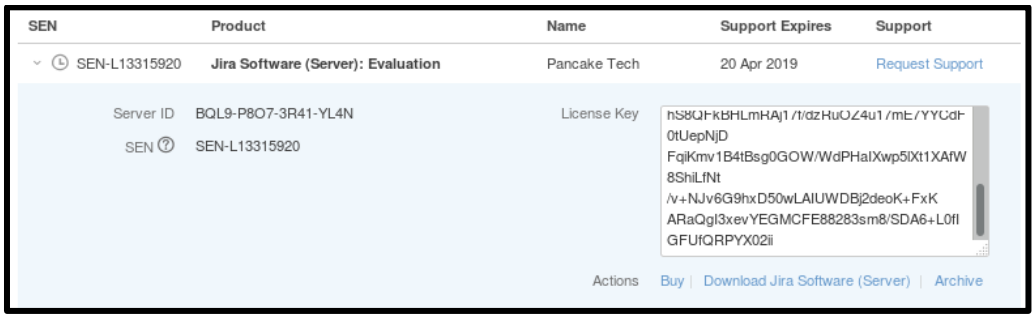


4. Select "Test Connection" to check if the database is connected with the JIRA Software. A prompt will appear if the connection is successful:

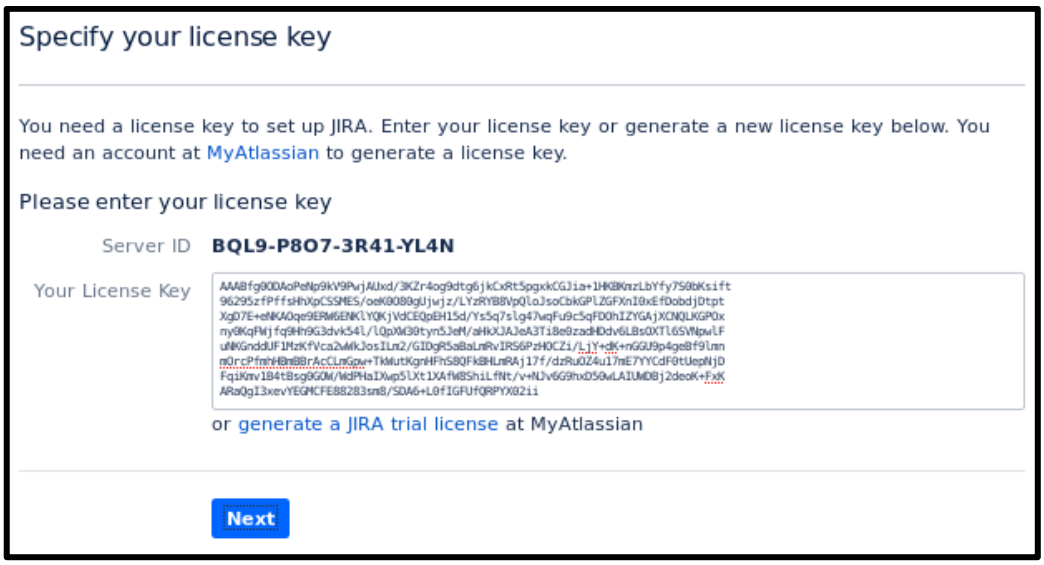
5. Click ‘next’ to proceed and select “Set up application properties”. Please fill the company application name, title and base URL accordingly.



6. Enter the license key for JIRA Software Server. Every JIRA Software installation will create a specific Server ID which is required to generate the license for each server. The license can be created via https://my.atlassian.com/product where the user can create a **New Evaluation License** in the portal. User will be prompted to select Product, Type of Server, Name of Company, Server ID and Server State. The portal will then generate the new license;

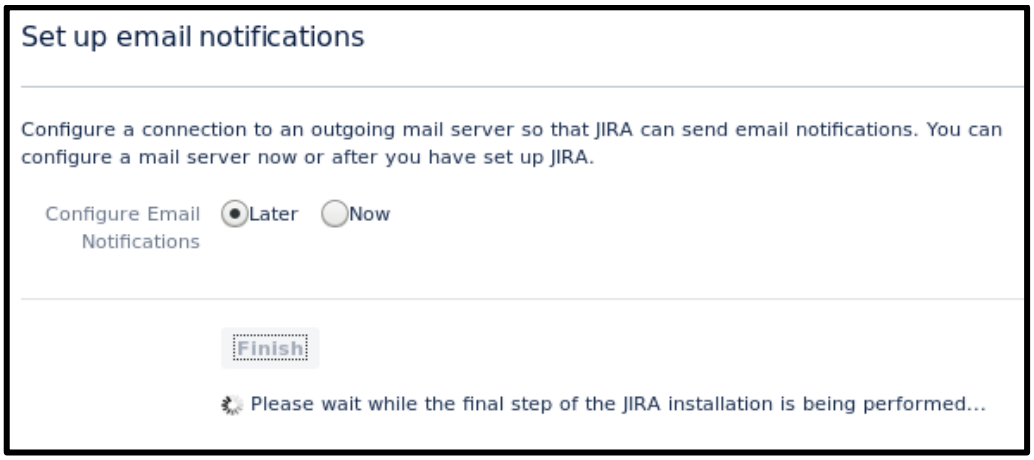


In JIRA's setup, copy the license created before in the license space to proceed to the JIRA Software setup shown below:



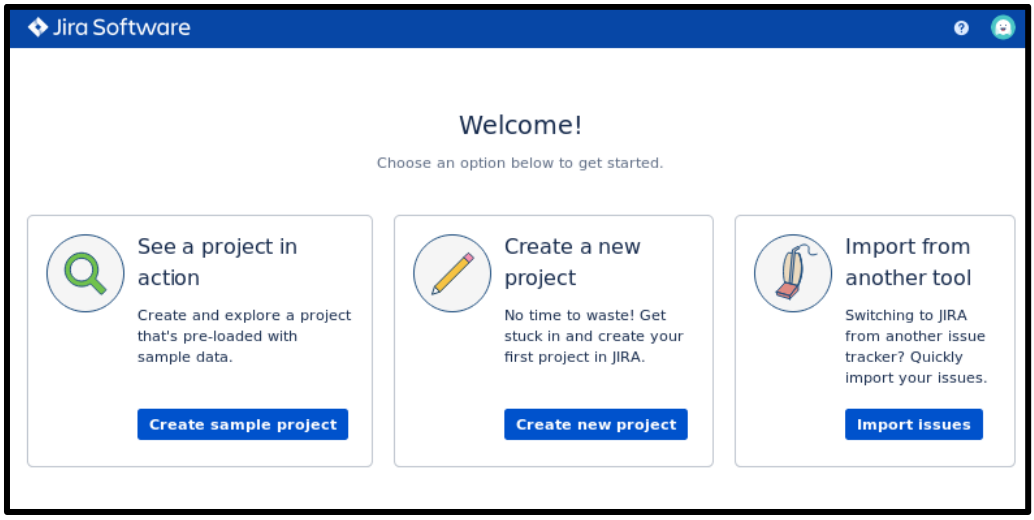
7. Create an administrator account, by filling in the required information;
    a. Name
    b. Email
    c. Username
    d. Password
    e. Confirm Password

8. Email notifications can be set up to be configured during or after the installation of JIRA software.



9. A prompt will appear to set up the preferred language that will be used for the application and the administrator avatar.

10. JIRA Software is successfully installed and is ready to be used.



3.6 Enabling Port Services in CentOS 7

In CentOS 7, all port services are blocked by the default FirewallD where full control over what traffic is allowed/disallowed to and from the system. Port services needs to be added manually via JIRA Software to enable it to be accessed from outside the network and other PCs.

To enable port used for JIRA Software:

```
$ sudo firewall-cmd --permanent --add--port=8080/tcp
$ sudo firewall-cmd --permanent --add--port=8005/tcp
```

To enable port used for MySQL:

```
$ sudo firewall-cmd --permanent --add--port=3306/tcp
```

To reload FirewallD to enable configuration:

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

4.0 DATA BACKUP

4.0 DATA BACKUP

4.1 Backup Data JIRA Software 7.10.2

For production use, regular backups are strongly recommended. When JIRA Software is in use, XML backups are not guaranteed to be consisted as the database may be updated during the backup process.

It is recommended to stop the use of JIRA Software once its' backup data migrates to another server. There are two things that requires backup in JIRA Software; XML Backup and Data Directory

4.1.1 XML Backup

Performing one off backup where it will create a backup with all the data in the JIRA Software are compiling in one backup file.

4.1.2 Attachments Directory

The data directory contains an application data for your JIRA instance. e.g. Issue attachments are stored in <jira/home>\data\attachments directory.

4.2 Backup Database Content

4.2.1 Using JIRA XML Backup Utility

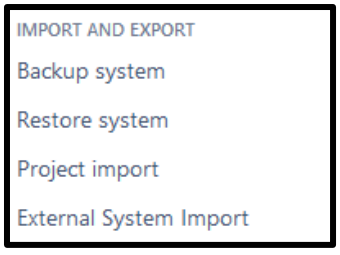
When JIRA is in use, XML backups are not guaranteed to be consistent as the database may be updated during the backup process. JIRA does not report any warnings or error messages when an XML backup is generated with inconsistencies and the XML backups will fail during the restoring process.

1. Login as an Administrator account in JIRA Software.

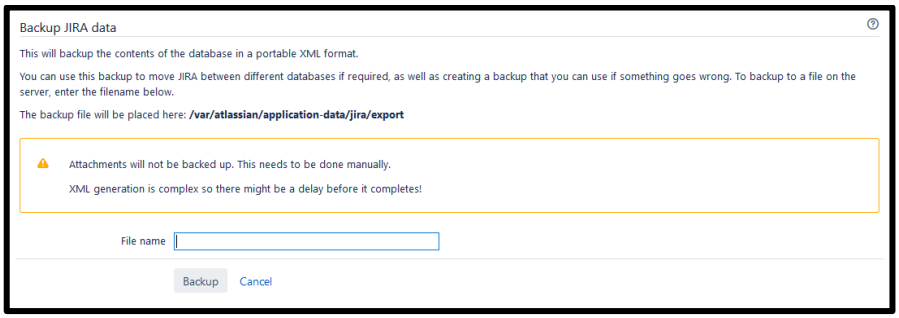
2. Click "**Settings**" on the top right corner and select "**System**".

3. JIRA Software will prompt for an Administrator Access. Enter the Administrator's password to validate unauthorized users.

4. Scroll down the dashboard to find the **Import and Export** section.



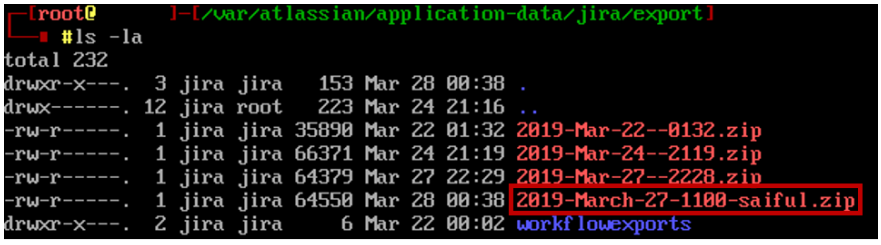
5. Select “**Backup System**” to backup JIRA Software data.



6. Create and enter the backup file name. It is recommended to be specific when naming a backup file; include date, time or username in which the file was created for future use.
   e.g. **2019-March-27-1100-Saiful**

7. Once the XML backup is successful, the software will appear to show the location of the file where it is exported to. (needs reviewing)

8. Login to CentOS 7 server and change the directory where the file is located; **/var/Atlassian/application-data/jira/export** to confirm the file creation in the server. Run commands **ls-la** to show all file information.



9. Transfer the backup file to another drive to restore the file in another server.

4.3 Backup Data Directory

4.3.1 Using JIRA XML Restore System

It is recommended to backup JIRA Software data directory, which is the sub-directory of the JIRA Software home directory. It contains application data of JIRA Software. e.g. Issue attachments are stored in the <jira-home>\data\attachments directory

1. Change to the attachment directory where the application data are stored:

```
$ cd /var/atlassian/application-data/jira/data
```

2. In attachments folder will contains all the project name created in the JIRA Software. For this current documentation there is two projects (LINUX and SEC) are created. Every issue created and attached with a file will in the issues name and the file will be rename as 10001 and will be increasing depending to the file in the project shown below using **tree** command.

```
┌─[root@saiful]─[/var/atlassian/application-data/jira/data]
└──╼ #tree
.
└── attachments
    ├── LINUX
    │   └── 10000
    │       ├── LINUX-1
    │       │   └── 10001
    │       └── LINUX-2
    │           └── 10000
    └── SEC
        └── 10000
            ├── SEC-1
            │   └── 10002
            └── SEC-2
                └── 10003

9 directories, 4 files
```

3. To backup this file we need to archive all the file in single file using ZIP command in Linux where **2019-March-27-saiful-attachment_backup.zip** is an example of file name.

```
$ zip -r 2019-March-27-saiful-attachment_backup.zip attachments/
```

4. A new file will be created after run the command and it now can be transfer to another instances.

5.0 DATA RESTORE

5.0 DATA RESTORE

Once the restoring procedure begins, all the existing data in the Jira application database is deleted**,** including all user accounts replaced data from XML backup file. Make sure that the valid password to a login in the backup file that has the Jira System Administrator global permission.

5.1 Disable email sending and receiving

If restoring the production data into a test Jira instance for experimentation purposes, System Administrator must disable all Jira application's email features before begins the restoration process:

i. Disable email notifications

ii. Disable POP/IMAP email polling

5.2 Restore the attachments

If you created a backup of the attachment's directory, you will need to restore the backup into a directory where Jira can access it.

1. Copy the zipped attachment file to JIRA Software **/attachments** directory.

```
$ cp <name attachments backup file>.zip /var/atlassian/application-data/jira/data/
```

2. Delete the current attachment file in the directory.

```
$ rm -rf attachments/
```

3. Unzip the file of attachment backup using UNZIP command, it will be extracting file attachments inside the directory.

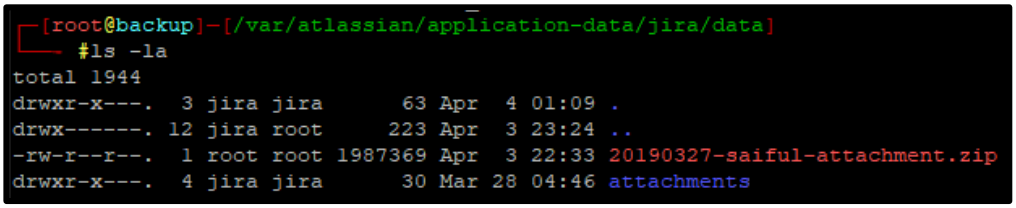
```
$ unzip <name attachments backup file>.zip
```

4. Change user ownership of the files to jira instead root or others. When transferring file in Linux it will save the ownership if the file with the current

user login. To avoid from JIRA Software from unable accessing the file, the file ownership must change to **jira.**

```
$ chown -R jira:jira attachments/
```

5. Check if the the ownership of the file has changed using **ls -la.**



5.3 Restore the XML file

1. Copy the zipped XML backup file to JIRA Software **/import** directory. To enable the JIRA Software to restore system into current system.

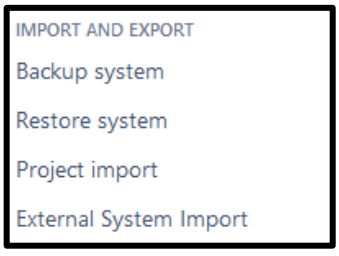
```
$ cp <nameXMLbackupfile>.zip/var/atlassian/applicationdata/jira/import/
```

2. Login as a System Administrator account in the JIRA Software on http://localhost:8080.

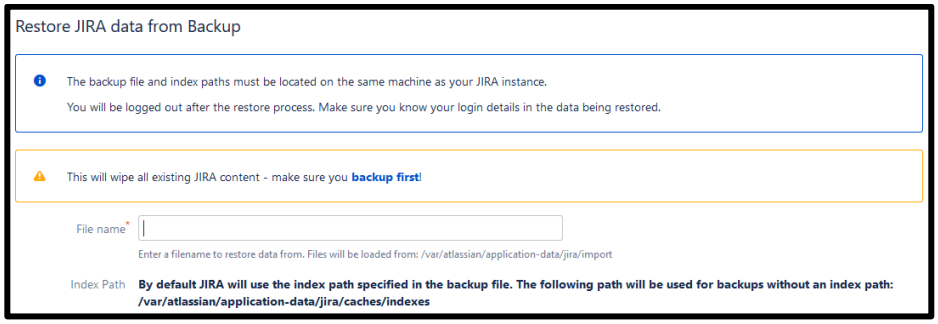
3. Click **Setting** on the right above corner and choose **System**.

4. The JIRA Software will prompt an Administrator Access which user need to enter the Administrator password to validate the user from unauthorized user.

5. Scroll down until found **Import & Export** Section in the dashboard.

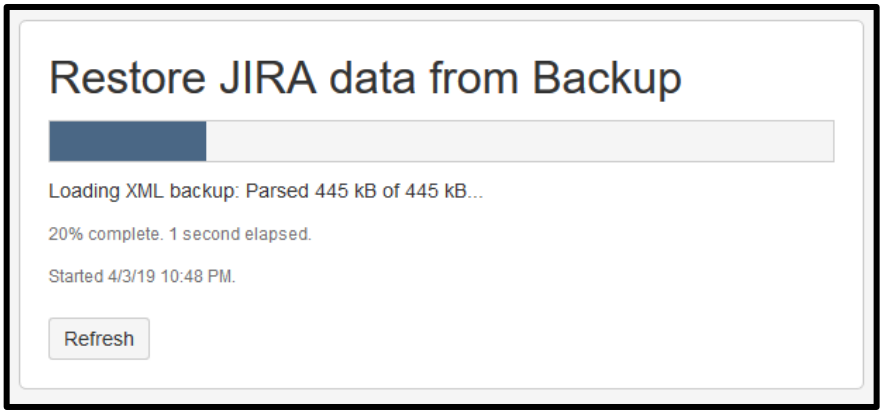


6. Choose **Restore system** to back up the JIRA Software data. In this board will be display two warning to the System Administrator:
    a. **File location:** The backup file and index paths must be located on the same machine as JIRA instance
    b. **Wipe Everything:** When running this restoration process all current data in JIRA Software will be deleted and replaced with a backup file data.



7. In the field of file name, enter a filename to restore data from. Files will be loaded from: **/var/atlassian/application-data/jira/import**

8. After entering the XML backup file, click on restore to start restoration process. The JIRA Software will start to restore JIRA data from the backup and this process might take some time depending to the size of the data.



9. Once the restoration process is finish, the restore wizard will prompt "**Your Import has been successful!".** Click on **Log In again** to used backup data in a new JIRA Software instance.