Progress: This in this document

1. **JIRA Software 7.10.2 Installation**
2. **MySQL 5.7**
3. **Creating and configuration MySQL database**
4. **Driver JDBC**
5.

JIRA Software Server (CentOS 7)

JIRA Software (7.10.2) Installation

1. Download Jira

To download Jira Software Server (7.10.2) can follow this command:

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

2. Run the installer

1. Change to the directory where the file is downloaded Jira then 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. Follow the prompts to install Jira. Installer will ask the following

info:

- **Install type** – choose option 2 (custom) for the most control.
- **Destination directory** – this is where Jira will be installed.
- **Home directory** – this is where Jira data like logs, search indexes and files will be stored.
- **TCP ports** – these are the HTTP connector port and control port Jira will run on. Stick with the default unless you're running another application on the same port.
- **Install as service** – this option is only available if you ran the installer as sudo.

3. Starting JIRA Software

After installation finish, JIRA Socan be start by running this command:

4. **Jira Web Setup**

Once installation is complete head to http://localhost:8080 in your browser to begin the setup process. (Replace 8080 if you chose a different port during installation).

Before Installing JIRA

1. Operating System: **CentOS 7 Minimal**
2. Database: **MySQL 5.7**
3. Driver: **MySQL Connector/J (JDBC)**
4. License: **Valid JIRA Software Server**

MySQL (5.7) Installation

1. **Default repository in CentOS 7**

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

2. **Download MySQL 5.7 repository**

   Download desired MySQL version for this installation using this command.

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

3. **Verify the integrity**

   Once the rpm is saved, we need to verify the integrity of downloaded file:

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

   The output of the MD5 values should same with MD5 value show below, if not the file might be corrupted while transferring:

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

4. Install the MySQL package

This will add new yum repositories.

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

To install MySQL 5.7 into machine, run this command.

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

5. Starting MySQL

Run this command to start MySQL daemon.

```
$ sudo systemctl start mysqld
```

Check the status of MySQL has successfully start by run this command and the output should contain Active: active (running).

6. Get MySQL Temporary Password

Every new installation of MySQL it generated temporary password for MySQL root user. Located in mysqld.log.

To retrieve the password run this command

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

7. Change MySQL Password Policy

Every time user wants to change the MySQL temporary password, it will ask the user to change with super powerful password as user create.

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

To prevent that we need to set a new policy in MySQL.

1. Enter to MySQL

    `$ mysql -u root -p`

2. Change the password policy

    `mysql> SET GLOBAL validate_password_policy = low;`

3. Successful changes the policy will prompt as below

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

8. **Configuration MySQL Password**

    Run MySQL security script by this command:

    `$ sudo mysql_secure_installation`

    Prompt will be asking user to enter password of the current user which temporary password before.

    ```
    ┌─[root@centos]─[~]
    └──➤ #mysql_secure_installation

    Securing the MySQL server deployment.

    Enter password for user root: █
    ```

    After entering a temporary password, it will prompt user to enter a new password for the root user account. Enter a new password and it will ask to confirm the password with asking to enter the password again.

    ```
    The existing password for the user account root has expired. Please set a new password.
    New password: █
    ```

    Following the prompt to change the password along with other configuration during running this script:

    - New password for root account
    - The strength of the password entered

- Remove anonymous users
- Disallow root login remotely
- Remove default database "test"
- Reload privileges database

9. Testing MySQL

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

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

Below the output of the entered commands:

```
┌─[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
```

Creating and configure the MySQL database

1. Create a database user which Jira will connect as, for example **jiradbuser**.

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

   *please note that **Centos1234** it just an example of password

2. Create a database for Jira to store issues in, for example **jiradb**.

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

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

3. Make sure the user has permission to connect to the database, and permission to create and populate tables. You can provide these permissions with the following commands.

   MySQL 5.7.0 - 5.7.5:

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

   MySQL 5.7.6 and later (must also include the REFERENCES permission):

```
GRANT
SELECT,INSERT,UPDATE,DELETE,CREATE,DROP,REFERENCES,ALTER,INDEX
on <JIRADB>.* TO '<USERNAME>'@'<JIRA_SERVER_HOSTNAME>' IDENTIFIED
BY '<PASSWORD>';
flush privileges;
```

4. Edit the my.cnf file in the 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 [mysqld] section in the file, and add or modify the following parameters:

- Set the default storage engine to InnoDB:

  `default-storage-engine=INNODB`

- Specify the character set used by the database server:

  `character_set_server=utf8mb4`

- Set the default row format to DYNAMIC:

  `innodb_default_row_format=DYNAMIC`

- Enable the large prefix:

  `innodb_large_prefix=ON`

- Set the InnoDB file format to Barracuda:

  `innodb_file_format=Barracuda`

- Specify the value of innodb_log_file_size to be at least 2G:

  `innodb_log_file_size=2G`

- Ensure the sql_mode parameter does not specify NO_AUTO_VALUE_ON_ZERO

// remove this if it exists

```
sql_mode = NO_AUTO_VALUE_ON_ZERO
```

6. Restart the MySQL server for the changes to take effect.

   Run following command to restart MySQL server:

   ```
   service mysqld stop
   ```

   *Then, run the same command, replacing stop with start.

JDBC Driver for MySQL 5.7

1. Copy the MySQL JDBC driver to the Jira installation directory.

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

2. Extract the 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
   ```

JIRA Software Setup

1.