

NiDB Installation

Greg Book – June 2020

1 System Preparation

NiDB is designed to run on CentOS 8. Running on other Linux distributions may work, but is not guaranteed. Separating the database/website/data servers are possible. NiDB has not been tested to work alongside other websites or database systems. Installing NiDB may alter existing programs and prevent their function. It is recommended to install NiDB on a dedicated server.

1.1 Hardware

More RAM, faster hard drive(s), and more CPU cores are all beneficial when querying data or performing analysis. There are no minimum specifications. If the hardware can run CentOS 8, then it should be able to run NiDB.

1.2 Operating System

Install CentOS 8 using the default options, with GUI. Running firefox *locally* is necessary to complete the setup. This can be run through VNC if direct access to the system is not possible. Root access is required to install NiDB. Several system packages will be installed to support NiDB, including MariaDB, Apache, and PHP.

1.3 Install FSL

FSL is required by NiDB, but must be installed separately due to licensing requirements. Install FSL from <https://fsl.fmrib.ox.ac.uk/fsl/fslwiki/FslInstallation> and note the location of the binary installation. The installation location might be something like `/usr/local/fsl/bin`.

1.3.1 Installing FSL

1. Visit <https://fsl.fmrib.ox.ac.uk/fsl/fslwiki/FslInstallation> and download the CentOS 8 version of FSL. Registration and agreeing to the license will be required.
2. Install python2.7 (as root)

```
wget https://fsl.fmrib.ox.ac.uk/fsl/downloads/fslinstaller.py # this may work
yum install python2
```

3. Install FSL (as root)

```
python2 fslinstaller.py
```

1.4 Install firejail

NiDB relies on firejail to do sandboxing for the mini-pipeline system. The mini-pipeline system runs arbitrary user-defined scripts, and must be sandboxed to prevent accidental or intentional system errors. Download firejail from <https://firejail.wordpress.com/> and follow the installation instructions.

1. Download the firejail current .rpm
2. Install (as root) `rpm -i firejail-x.y.z.rpm`

2 Install NiDB

2.1 Install Prerequisite Packages

2.1.1 CentOS 7

PHP7+ is required by NiDB, but is not provided by default in CentOS7. The following commands allow for the installation, via yum, of PHP 7.3.

```
yum install epel-release yum-utils
yum install http://rpms.remirepo.net/enterprise/remi-release-7.rpm
yum-config-manager --enable remi-php73
```

MariaDB 10+ is required by NiDB, but is not provided by default in CentOS 7. Install MariaDB using the following commands.

```
wget https://downloads.mariadb.com/MariaDB/mariadb\_repo\_setup
chmod +x mariadb_repo_setup
sudo ./mariadb_repo_setup
yum install MariaDB-server
```

2.1.2 CentOS 8

ImageMagick is required by NiDB, but is not included in the default yum repositories. The following command will enable the repository using yum.

```
yum install epel-release
```

2.2 Install NiDB RPM

Download the latest RPM from github releases. Install using the following command(s). PowerTools repository is required to allow installation of Perl modules. EPEL release repository is required to allow installation of ImageMagick.

```
yum --nogpgcheck localinstall nldb-xxxx.x.xx-1.el8.x86_64.rpm
```

Run MariaDB secure installation script using following command (for new installations only), and follow the instructions to setup MariaDB, using the suggested settings. By this point in the install the password should be 'password'. Due to a bug in the script below, don't attempt to change the root password here. This can be done later after the installation is complete.

```
bash$ sudo mysql_secure_installation
Enter current password for root (enter for none):
Change the root password? [Y/n] n
Remove anonymous users? [Y/n] Y
Disallow root login remotely? [Y/n] Y
Remove test database and access to it? [Y/n] Y
Reload privilege tables now? [Y/n] Y
```

2.3 Finish Setup

Complete the setup by launching Firefox and going to <http://localhost/setup.php> **Note: setup.php will only be accessible from the same server that the webserver is running on.** The list of IP addresses allowed to access setup.php can be changed by manually editing the nldb.cfg file and changing the setupips variable.

Firefox can also be launched by typing the command

```
firefox http://localhost/setup.php
```

After setup is complete, you should be able to navigate to <http://localhost> and login as 'admin' with password of 'password'

2.4 Extra Software

2.4.1 phpMyAdmin

PhpMyAdmin is an extremely useful tool to manage the SQL database which NiDB uses. Download it from <http://phpmyadmin.net>. Basically unzip the downloaded file into `/var/www/html/` and rename the unzipped directory to `phpMyAdmin`. It will then be available from <http://localhost/phpMyAdmin>.

2.5 Usernames & Passwords

Default usernames and passwords created by the install process are as follows:

	Username	Password
(Linux)	nldb	password
MariaDB	root	password
MariaDB	nldb	password
NiDB	admin	password

It is important to change passwords after the installation. Change passwords using the following methods

2.5.1 Linux

From the command line, perform the following, and follow the prompts

As root user:

```
passwd nldb
```

As nldb user

```
passwd
```

2.5.2 MariaDB

The easiest method to change MariaDB passwords is from within phpMyAdmin. Login to <http://localhost/phpMyAdmin> using the root MySQL account and password. Go to the **User Accounts** menu option. Then click *Edit privileges* for the root account that has a '%' as the hostname. Then click *Change password* button at the top of the page. Enter a new password and click **Go**.

2.5.3 NiDB

To change the admin password with NiDB, go to **My Account**, and enter a new password in the password field(s). Click **Save** to change the password.

3 Upgrade Existing Installation

If you have an existing installation of NiDB, you can upgrade using the following methods. However, upgrading to a newer version is a potentially dangerous task as data loss may occur. Be careful when upgrading NiDB, and if you are unsure of the safety of any steps, consult the NiDB support forum.

3.1 Preparation Steps

3.1.1 Notify Users

Inform your users of the upcoming downtime and upgrade. Inform your users via email, in person, etc. Or use the Admin → Settings page to set a message that is displayed at the top of every page.

3.1.2 Turn off Website

Manually edit the `nldb.cfg` file to set `offline` to 1. This will prevent users from accessing any of the website features.

3.1.3 Stop all Modules

On the Admin → Modules page, set all modules to **disabled**.

3.2 Upgrade

3.2.1 Backup Database

Make a backup copy of the database by performing the following command from the command line. Note the location of the `.sql` file. This is the only backup of the database that is created.

```
cd /nldb
mysqldump -uroot -ppassword nldb > nldb-backup-2020-06-30.sql
```

3.2.2 Upgrade nldb using yum

Download the current version of the `nldb.rpm` from github releases. Install the `.rpm` using the following command

```
yum --nogpgcheck localinstall nldb-xxxx.x.xx-1.el8.x86_64.rpm
```

After installing the `.rpm`, launch firefox to complete the upgrade via the website

```
Firefox http://localhost/setup.php
```

3.3 Final Steps

3.3.1 Turn on Website

Manually edit the `nldb.cfg` file to set `offline` to 0.

3.3.2 Enable Modules

Under Admin → Modules, set modules to **enabled**.

4 Migrate Existing Installation to New Server

Migrating an NiDB installation from one server to another is very similar to upgrading an existing installation. The only changes are the need to copy the exported old database to the new server and import it prior to the upgrade.

4.1 Preparation Steps (Both Old and New Servers)

4.1.1 Notify Users

Inform your users of the upcoming downtime and upgrade. Inform your users via email, in person, etc. Or use the Admin → Settings page to set a message that is displayed at the top of every page.

4.1.2 Turn off Website

Manually edit the `nldb.cfg` file to set `offline` to 1. This will prevent users from accessing any of the website features.

4.1.3 Stop all Modules

On the Admin → Modules page, set all modules to **disabled**.

4.2 Export Old Database/Transfer to New Server

Perform the following command on the old server to dump the NiDB SQL database

```
mysqldump -uroot -ppassword nldb > nldb-backup.sql
```

Copy the exported .sql file to the new server. Many possibilities for how to do this.

On the new server, perform the following command to import the new database

```
mysql -uroot -ppassword nldb < nldb-backup.sql
```

4.3 Finish Upgrade on New Server

Launch firefox using the following command

```
firefox http://localhost/setup.php
```

Follow the prompts on the webpages to complete the upgrade process.