

2017/11/10

Mxnet2 for Linux(64bit)

Contents

1 .	Setting up udev rule file	2
	udev reload command	3
2 .	Matrix API Shared Library Setup.....	3
3 .	Linux version of mxnet2 server	4
4 .	Connection Status Display	5

Mxnet2 for Linux (64bit) is a version of mxnet2 server that runs on 64bit Linux. It offers similar functionalities to Windows version of mxnet2 and can be used in place of Windows mxnet2 server. The remote client API can connect to mxnet2 servers running either on Windows and Linux. No 32 version of mxnet2 for Linux is available.

On Linux box, you have to set up udev rule file to deploy Matrix dongles for mxnet2; otherwise, mxnet2 will not be able to detect Matrix dongles.

1 . Setting up udev rule file

Linux will automatically detect Matrix dongles and create device nodes for them. The default permission setting of the device node gives the access only to root user. Non-root user cannot open the node. We will install the rule file that changes the access mode of the auto-created node to "0666". The rule also runs a script when a dongle is plugged in and out and the plug-in and plug-out events to mxnet2 server are fired.

Open "99-mxnet.rules" under udev directory in the distribution ZIP file. The file contains two rules to be applied when a dongle is added and removed.

```
SUBSYSTEM=="usb", DRIVER=="usb", ACTION=="add", ATTRS{idVendor}=="0e50",  
ATTRS{idProduct}=="000[1-9]", MODE="0666",RUN+="/path to/mxevent -a"  
SUBSYSTEM=="usb",ACTION=="remove",ENV{ID_VENDOR_ID}=="0e50",ENV{ID_  
MODEL_ID}=="000[0-9]", RUN+="/path to /mxevent -r"
```

You must edit the path to "mxevent" program to point to the correct position.

- 1 . Copy “mxevent” under udev directory to a fixed disk directory
- 2 . Set the path to the destination to RUN+=””

- ACTION==”add” rule requires “-a” parameter to “mxevent”.
RUN+=“/../../mxevetn -a”
- ACTION==”remove” rule calls for “-r” parameter to “mxevent2”.
RUN+=“/../../mxevetn -r”

After editing 99-mxevent.rules file, copy it to /etc/udev/rules.d directory. You must have root privileges for the copy.

The new rule will be active after udev is reloaded. You can re-boot the system or issue the udev reload command.

udev reload command

```
>udevadm control --reload
```

Now plug Matrix dongle and see if “mxevent” is run. “mxevent” will write the plug-in and plug-out to the system log. udev also write the plug-in/out event to the system log. You should find an entry by “mxevent” with the following message.

```
dongle plug event is raised  
dongle unplug event is raised
```

When mxnet2 is not running, mxevent terminates with the exit code -1.

If you do not find any entry by mxevent, it is likely that udev rule is not correctly applied. The udev file must be active for mxnet2 to function properly.

2 . Matrix API Shared Library Setup

mxnet2 server has dependency on Matrix API shared library. Install the shared library by running install.sh under API directory. install.sh will ask which version X86 or X64 to install. Be sure to select 2 (x64 version)

3 . Linux version of mxnet2 server

Now you are ready to run mxnet2 server. Before running the server, review the content of the configuration file, mxnet2.cfg, in the same directory. The client must have the same Port setting.

```
[option]
Port=12300
LoginTimeOut=300
SessionTimeOut=3600
LogLevel=0
SessionBaseLicense=1
CheckCmdIssueTime=1
CmdIssueTimeInterval=30
```

```
[dongle1]
Sernr=1234567890
Name=MyDongle
5=MyApplication
```

* The section and key names in mxnet2.cfg is case-sensitive. Do not change the cases of characters in Mxnet2.cfg.

The default values under [option] section are the same as those in Windows's mxnet2.ini

The command line parameter given to mxnet2 can override the values set in the configuration file. For the detail on the command line arguments, run "mxnet2 -help"

[dongleX] sections are the same as those in Windows's mxnet2.ini. They are used when displaying the connection status by "listlogin" program. You can find the program in mxnet2 directory.

Stopping MxNet2 Server

When run, Mxnet2 displays the optional values. To exit the program, press CTRL+C(SIGINT). Once mxnet2 runs without an error, plug in and out a dongle and see if mxnet2 detects the dongle event correctly. After making sure that a dongle is connected, run a client with Remote API to connect to the server.

Running MxNet2 server as a daemon

Mxnet2 can run as a daemon by giving --daemonize or -d command argument. It will keep running after the terminal in which the daemonized mxnet2 is run is closed or you log out. Mxnet2 daemon can be terminated by SIGTERM signal.

Dropping root privileges

When Mxnet2 is running with the root privileges, it will warn to set User/Group options to drop the privilege. You can specify User/Group under with whose privileges mxnet2 is to run in the configuration file or in the command line

```
[option]
```

```
User=xxxx
```

```
Group=xxxx
```

```
>mxnet2 --user xxxx --group yyy
```

4 . Connection Status Display

Using the command line program, you can display mxnet2's connection status. It shows the same message as "Connection Status Display" of Windows version of mxnet2.

Without an argument, listlogin displays the currently active and zombie sessions and Appslot login status of each session.

```
>listlogin
```

The list contains sessionID and DngNr/AppSlot

To close a session, run with `-release (-r)` command,

```
>listlogin -release (sessionID)
```

To log off a session from AppSlot, use `-logout`, `--dngnr` and `-appslot` commands.

```
>listlogin -logout sessionid -dngnr dngnr -appslot appslot
```