README

Below is a preview of packaging improvements for the NVIDIA driver on Red Hat Enterprise Linux (RHEL).

1. For supported RHEL kernel releases, driver packages are provided that implement an alternative to DKMS. The new approach does not require `gcc` to be installed anymore, nor does the EPEL repository need to be enabled. The source files for the driver kmod packages are compiled in advance and then linked at installation time, hence these are called "precompiled drivers".

2. Multiple driver branches are now installable from a single package repository. The user can choose a specific driver branch or a virtual branch. Only updates on the selected branch will be considered, where the "latest" branch simply always tracks the latest driver release.

3. Simplified switching between driver branches via a `yum` plugin that complements installation and uninstallation of driver packages.

   Note: Currently these package improvements are only supported for RHEL 7.6 on x86_64 architecture. Support for RHEL 8 on x86_64 and other platforms will be added in the future. Other known issues are listed at the bottom of this document.

Pre-installation Actions:

- On RHEL 7 Linux (x86_64 workstation), execute the following:

  ```
  $ subscription-manager repos --enable=rhel-7-workstation-optional-rpms
  ```

- Add the tech preview repository:

  ```
  $ sudo yum-config-manager --add-repo=http://developer.download.nvidia.com/compute/cuda/preview/repos/rhel7/x86_64/techpreview_nvidia_rh_drv.repo
  ```

- Install the NVIDIA yum plugin.

  ```
  $ sudo yum install yum-plugin-nvidia
  ```

  Note: this command must be separate from those below that are used to install the drivers, and must be executed before those.

Precompiled (default and recommended)

- Verify that you have a supported kernel installed.

  ```
  $ uname -r
  3.10.0-957.12.2.el7.x86_64
  ```

  Note: For the tech preview, kmod packages are only provided for the 3.10.0-957.12.2 kernel.

DKMS (non-supported kernels)

Note: This is only needed for special circumstances such as custom kernels.

- Make sure the EPEL repository is enabled.

  ```
  $ sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
  ```

Package Manager Installation:

Select an installation branch

To select an installation branch, choose only one from the three options below:

1. Always update to the highest versioned driver (precompiled).

   ```
   $ sudo yum install nvidia-driver-latest
   ```

   Note: The following commands are equivalent.

   ```
   $ sudo yum install cuda-drivers
   $ sudo yum install nvidia-driver
   ```

2. Lock the driver updates to the specified driver branch (precompiled).

   ```
   $ sudo yum install nvidia-driver-branch-<id>
   ```

   Note: replace `<id>` with the appropriate driver branch. For the tech preview, valid values are `418` or `430`.

3. Always update to the highest versioned driver (*non-precompiled*).

   ```
   $ sudo yum install nvidia-driver-latest-dkms
   ```

   Note: This is only recommended for unsupported kernels.

Switch branches

- Swap from `latest` to `branch-418` branch.

```
$ sudo yum swap nvidia-driver nvidia-driver-branch-418
```

- Swap from `branch-430` to `latest` branch.

  ```
  $ sudo yum swap nvidia-driver nvidia-driver-latest
  ```

- Swap from `latest` to `latest-dkms` branch.

  ```
  $ sudo yum swap nvidia-driver nvidia-driver-latest-dkms
  ```

Known issues

- Currently unsupported: swap from `latest-dkms` to `latest` or `branch-<id>` branch.

  First remove driver packages prior to switching from *dkms* to a *precompiled* branch, as below:

  ```
  $ sudo yum remove nvidia-driver
  ```

- POWER (ppc64le) is not supported.

- Diagnostic packages can be optionally installed:

  - **latest** `$ sudo yum install nvidia-driver-latest-diagnostic`
  - **branch-<id>** `$ sudo yum install nvidia-driver-branch-<id>-diagnostic`
  - **latest-dkms** `$ sudo yum install nvidia-driver-latest-dkms-diagnostic`

  However, any such diagnostic packages must be removed prior to switching to another branch.

  Note: Diagnostic packages are deprecated.
```