



VIRTUAL GPU SOFTWARE R384 FOR HUAWEI UVP

RN-07939-001 _v5.0 and 5.1 | May 2018

Release Notes



TABLE OF CONTENTS

Chapter 1. Release Notes	1
1.1. Updates in Release 5.0.....	1
1.2. Updates in Release 5.1.....	2
Chapter 2. Validated Platforms	3
2.1. Supported NVIDIA GPUs and Validated Server Platforms.....	3
2.2. Hypervisor Software Releases.....	3
2.3. Guest OS Support.....	4
2.3.1. Windows Guest OS Support.....	4
2.3.2. Linux Guest OS Support.....	5
Chapter 3. Known Product Limitations	6
3.1. VM running older NVIDIA vGPU drivers fails to initialize vGPU when booted.....	6
3.2. Virtual GPU fails to start if ECC is enabled.....	7
3.3. Single vGPU benchmark scores are lower than passthrough GPU.....	9
3.4. nvidia-smi fails to operate when all GPUs are assigned to GPU passthrough mode.....	9
3.5. VMs configured with large memory fail to initialize vGPU when booted.....	10
3.6. vGPU host driver RPM upgrade fails.....	11
Chapter 4. Resolved Issues	12
Chapter 5. Known Issues	13
5.1. Since 5.1: Issues in remote desktop sessions if a license is acquired after a session is started.....	13
5.2. License settings configured through a GPO are ignored.....	14
5.3. Licensing pop-up windows contain the text microsoft.explorer.notification.....	14
5.4. Since 5.2: The license expires prematurely in Linux guest VMs.....	15
5.5. Multiple display heads are not detected by Ubuntu 14.04 guest VMs.....	15
5.6. Since 5.1: On GPUs based on the Pascal architecture, Ubuntu 16.04 VMs run slowly after acquiring a license.....	16
5.7. Resolution is not updated after a VM acquires a license and is restarted.....	16
5.8. NVIDIA vGPU encoder and process utilization counters don't work with Windows Performance Counters.....	17
5.9. A segmentation fault in DBus code causes nvidia-gridd to exit on Red Hat Enterprise Linux and CentOS.....	18
5.10. Since 5.1: No Manage License option available in NVIDIA X Server Settings by default....	19
5.11. Since 5.1: The nvidia-gridd service fails because the required configuration is not provided.....	20
5.12. Since 5.1: The Apply button is disabled after change to unlicensed mode.....	21
5.13. Licenses remain checked out when VMs are forcibly powered off.....	21
5.14. Multiple WebGL tabs in Microsoft Internet Explorer may trigger TDR on Windows VMs...	22

Chapter 1.

RELEASE NOTES

These *Release Notes* summarize current status, information on validated platforms, and known issues with NVIDIA vGPU software and associated hardware on Huawei UVP.

The releases in this release family of NVIDIA vGPU software include the software listed in the following table:

Software	5.0	5.1	5.2	5.3
NVIDIA Virtual GPU Manager for the Huawei UVP releases listed in Hypervisor Software Releases	384.73	384.99	Not supported	Not supported
NVIDIA Windows driver	385.41	385.90	Not supported	Not supported
NVIDIA Linux driver version	384.73	384.99	Not supported	Not supported



Caution

If you install the wrong package for the version of Huawei UVP you are using, NVIDIA Virtual GPU Manager will fail to load.

The releases of the vGPU Manager and guest VM drivers that you install must be compatible. Different versions of the vGPU Manager and guest VM driver from within the same main release branch can be used together. For example, you can use the vGPU Manager from release 5.1 with guest VM drivers from release 5.0. However, versions of the vGPU Manager and guest VM driver from different main release branches cannot be used together. For example, you cannot use the vGPU Manager from release 5.1 with guest VM drivers from release 4.4. See [VM running older NVIDIA vGPU drivers fails to initialize vGPU when booted](#).

1.1. Updates in Release 5.0

New Features in Release 5.0

- ▶ New NVIDIA vGPU schedulers for GPUs based on the NVIDIA Pascal architecture
- ▶ Support for NVML and `nvidia-smi` on 32-bit Windows VMs

- ▶ Application-level monitoring of NVIDIA vGPU engine utilization
- ▶ Encoder session monitoring
- ▶ Support for NVENC on Linux NVIDIA vGPUs
- ▶ Software enforcement of licensing requirements
- ▶ Miscellaneous bug fixes

Feature Support Withdrawn in Release 5.0

- ▶ GRID K1 and GRID K2 GPUs are no longer supported.

1.2. Updates in Release 5.1

New Features in Release 5.1

- ▶ Miscellaneous bug fixes

Chapter 2.

VALIDATED PLATFORMS

This release of NVIDIA vGPU software provides support for several NVIDIA GPUs on validated server hardware platforms, Huawei UVP hypervisor software versions, and guest operating systems.

2.1. Supported NVIDIA GPUs and Validated Server Platforms

This release of NVIDIA vGPU software provides support for the following NVIDIA GPUs on Huawei UVP, running on validated server hardware platforms:

- ▶ Tesla M60

For a list of validated server platforms, refer to [NVIDIA GRID Certified Servers](#).



Tesla M60 and M6 GPUs support compute mode and graphics mode. NVIDIA vGPU requires GPUs that support both modes to operate in graphics mode.

Recent Tesla M60 GPUs and M6 GPUs are supplied in graphics mode. However, your GPU might be in compute mode if it is an older Tesla M60 GPU or M6 GPU, or if its mode has previously been changed.

To configure the mode of Tesla M60 and M6 GPUs, use the `gpumodeswitch` tool provided with NVIDIA vGPU software releases.

2.2. Hypervisor Software Releases

This release supports **only** the hypervisor software releases listed in the table.



If a specific release, even an update release, is not listed, it's **not** supported.

Software	Release Supported
Huawei UVP	Version RC520

2.3. Guest OS Support

NVIDIA vGPU software supports several Windows releases and Linux distributions as a guest OS. The supported guest operating systems depend on the hypervisor software version.



Use only a guest OS release that is listed as supported by NVIDIA vGPU software with your virtualization software. To be listed as supported, a guest OS release must be supported not only by NVIDIA vGPU software, but also by your virtualization software. NVIDIA cannot support guest OS releases that your virtualization software does not support.

2.3.1. Windows Guest OS Support

NVIDIA vGPU software supports **only** the Windows releases listed in the table as a guest OS on Huawei UVP. The releases of Huawei UVP for which a Windows release is supported depend on whether NVIDIA vGPU or pass-through GPU is used.



If a specific release, even an update release, is not listed, it's **not** supported.

Guest OS	NVIDIA vGPU - Huawei UVP Releases	Pass-Through GPU - Huawei UVP Releases
Windows Server 2016 1607, 1709	RC520	RC520
Windows Server 2012 R2	RC520	RC520
Windows Server 2008 R2	RC520	RC520
Windows 10 RTM (1507), November Update (1511), Anniversary Update (1607), Creators Update (1703) (64-bit)	RC520	RC520
Windows 10 RTM (1507), November Update (1511), Anniversary Update (1607), Creators Update (1703) (32-bit)	RC520	RC520
Windows 8.1 Update (64-bit)	RC520	RC520
Windows 8.1 Update (32-bit)	RC520	RC520
Windows 8.1 (64-bit)	RC520	-
Windows 8.1 (32-bit)	RC520	-
Windows 8 (32/64-bit)	RC520	-
Windows 7 (32/64-bit)	RC520	RC520

2.3.2. Linux Guest OS Support

NVIDIA vGPU software supports **only** the Linux distributions listed in the table as a guest OS on Huawei UVP. The releases of Huawei UVP for which a Linux release is supported depend on whether NVIDIA vGPU or pass-through GPU is used.



If a specific release, even an update release, is not listed, it's **not** supported.

Guest OS	NVIDIA vGPU - Huawei UVP Releases	Pass-Through GPU - Huawei UVP Releases
Red Hat Enterprise Linux 6.6	RC520	RC520
CentOS 6.6	RC520	RC520
Ubuntu 14.04 LTS	RC520	RC520

Chapter 3.

KNOWN PRODUCT LIMITATIONS

Known product limitations for this release of NVIDIA vGPU software are described in the following sections.

3.1. VM running older NVIDIA vGPU drivers fails to initialize vGPU when booted

Description

A VM running a version of the NVIDIA guest VM drivers from a previous main release branch, for example release 4.4, will fail to initialize vGPU when booted on a Huawei UVP platform running the current release of Virtual GPU Manager.

In this scenario, the VM boots in standard VGA mode with reduced resolution and color depth. The NVIDIA virtual GPU is present in **Windows Device Manager** but displays a warning sign, and the following device status:

```
Windows has stopped this device because it has reported problems. (Code 43)
```

Depending on the versions of drivers in use, the Huawei UVP VM's `/var/log/messages` log file reports one of the following errors:

- ▶ An error message:

```
vmiop_log: error: Unable to fetch Guest NVIDIA driver information
```

- ▶ A version mismatch between guest and host drivers:

```
vmiop_log: error: Guest VGX version(1.1) and Host VGX version(1.2) do not match
```

- ▶ A signature mismatch:

```
vmiop_log: error: vGPU message signature mismatch.
```


Resolution

Install the current NVIDIA guest VM driver in the VM.

3.2. Virtual GPU fails to start if ECC is enabled

Description

Tesla M60, Tesla M6, and GPUs based on the Pascal GPU architecture, for example Tesla P100 or Tesla P4, support error correcting code (ECC) memory for improved data integrity. Tesla M60 and M6 GPUs in graphics mode are supplied with ECC memory disabled by default, but it may subsequently be enabled using `nvidia-smi`. GPUs based on the Pascal GPU architecture are supplied with ECC memory enabled.

However, NVIDIA vGPU does not support ECC memory. If ECC memory is enabled, NVIDIA vGPU fails to start.

The following error is logged in the Huawei UVP VM's `/var/log/messages` log file:

```
vmiop_log: error: Initialization: VGX not supported with ECC Enabled.
```

Resolution

Ensure that ECC is disabled on all GPUs.

Before you begin, ensure that NVIDIA Virtual GPU Manager is installed on your hypervisor.

1. Use `nvidia-smi` to list the status of all GPUs, and check for ECC noted as enabled on GPUs.

```
# nvidia-smi -q
=====NVSMI LOG=====
Timestamp                : Tue Dec 19 18:36:45 2017
Driver Version           : 384.99
Attached GPUs            : 1
GPU 0000:02:00.0
[...]
  Ecc Mode
    Current                : Enabled
    Pending                : Enabled
[...]
```

2. Change the ECC status to off on each GPU for which ECC is enabled.
 - ▶ If you want to change the ECC status to off for all GPUs on your host machine, run this command:

```
# nvidia-smi -e 0
```

- ▶ If you want to change the ECC status to off for a specific GPU, run this command:

```
# nvidia-smi -i id -e 0
```

id is the index of the GPU as reported by `nvidia-smi`.

This example disables ECC for the GPU with index `0000:02:00.0`.

```
# nvidia-smi -i 0000:02:00.0 -e 0
```

3. Reboot the host.

```
# reboot
```

4. Confirm that ECC is now disabled for the GPU.

```
# nvidia-smi -q
```

```
=====NVSMI LOG=====
```

```
Timestamp                : Tue Dec 19 18:37:53 2017
```

```
Driver Version           : 384.99
```

```
Attached GPUs            : 1
```

```
GPU 0000:02:00.0
```

```
[...]
```

Ecc Mode

```
Current                   : Disabled
```

```
Pending                   : Disabled
```

```
[...]
```

If you later need to enable ECC on your GPUs, run one of the following commands:

- ▶ If you want to change the ECC status to on for all GPUs on your host machine, run this command:

```
# nvidia-smi -e 1
```

- ▶ If you want to change the ECC status to on for a specific GPU, run this command:

```
# nvidia-smi -i id -e 1
```

id is the index of the GPU as reported by `nvidia-smi`.

This example enables ECC for the GPU with index `0000:02:00.0`.

```
# nvidia-smi -i 0000:02:00.0 -e 1
```

After changing the ECC status to on, reboot the host.

3.3. Single vGPU benchmark scores are lower than passthrough GPU

Description

A single vGPU configured on a physical GPU produces lower benchmark scores than the physical GPU run in passthrough mode.

Aside from performance differences that may be attributed to a vGPU's smaller framebuffer size, vGPU incorporates a performance balancing feature known as Frame Rate Limiter (FRL), which is enabled on all vGPUs. FRL is used to ensure balanced performance across multiple vGPUs that are resident on the same physical GPU. The FRL setting is designed to give good interactive remote graphics experience but may reduce scores in benchmarks that depend on measuring frame rendering rates, as compared to the same benchmarks running on a passthrough GPU.

Resolution

FRL is controlled by an internal vGPU setting. NVIDIA does not validate vGPU with FRL disabled, but for validation of benchmark performance, FRL can be temporarily disabled by setting `plugin0.frame_rate_limiter=0` in the vGPU configuration file. vGPU configuration files are stored in `/usr/share/nvidia/vgx` and are named for the vGPU types they define, for example, `grid_k100.conf`.

The setting takes effect the next time any VM using the given vGPU type is started or rebooted.

With this setting in place, the VM's vGPU will run without any frame rate limit. The FRL can be reverted back to its default setting by setting `plugin0.frame_rate_limiter=1` in the vGPU configuration file.

3.4. `nvidia-smi` fails to operate when all GPUs are assigned to GPU passthrough mode

Description

If all GPUs in the platform are assigned to VMs in passthrough mode, `nvidia-smi` will return an error:

```
[root@vgx-test ~]# nvidia-smi
Failed to initialize NVML: Unknown Error
```

This is because GPUs operating in passthrough mode are not visible to `nvidia-smi` and the NVIDIA kernel driver operating in the Huawei UVP dom0.

Resolution

N/A

3.5. VMs configured with large memory fail to initialize vGPU when booted

Description

When starting multiple VMs configured with large amounts of RAM (typically more than 32GB per VM), a VM may fail to initialize vGPU. In this scenario, the VM boots in standard VGA mode with reduced resolution and color depth. The NVIDIA vGPU software GPU is present in **Windows Device Manager** but displays a warning sign, and the following device status:

```
Windows has stopped this device because it has reported problems. (Code 43)
```

The Huawei UVP VM's `/var/log/messages` log file contains these error messages:

```
vmiop_log: error: NVOS status 0x29
vmiop_log: error: Assertion Failed at 0x7620fd4b:179
vmiop_log: error: 8 frames returned by backtrace
...
vmiop_log: error: vGPU message 12 failed, result code: 0x29
...
vmiop_log: error: NVOS status 0x8
vmiop_log: error: Assertion Failed at 0x7620c8df:280
vmiop_log: error: 8 frames returned by backtrace
...
vmiop_log: error: vGPU message 26 failed, result code: 0x8
```

Resolution

vGPU reserves a portion of the VM's framebuffer for use in GPU mapping of VM system memory. The reservation is sufficient to support up to 32GB of system memory, and may be increased to accommodate up to 64GB by specifying `plugin0.enable_large_sys_mem=1` in the vGPU configuration file.

vGPU configuration files are stored in `/usr/share/nvidia/vgx` and are named for the vGPU types they define, for example, `grid_k100.conf`.

The setting takes effect the next time any VM using the given vGPU type is started or rebooted.

With this setting in place, less GPU FB is available to applications running in the VM. To accommodate system memory larger than 64GB, the reservation can be further increased by specifying `plugin0.extra_fb_reservation` in the vGPU configuration file, setting its value to the desired reservation size in megabytes. The default value of 64M is sufficient to support 64GB of RAM. We recommend adding 2M of reservation for

each additional 1GB of system memory. For example, to support 96GB of RAM, set `extra_fb_reservation` to 128:

```
plugin0.extra_fb_reservation=128
```

The reservation can be reverted back to its default setting in one of the following ways:

- ▶ Removing `enable_large_sys_mem` from the vGPU configuration file
- ▶ Setting `enable_large_sys_mem=0`

3.6. vGPU host driver RPM upgrade fails

Description

Upgrading vGPU host driver RPM fails with an error message about failed dependencies on the console.

```
[root@uvp ~]# rpm -U NVIDIA-vGPU-kepler-uvp-210.0-352.70.x86_64
error: Failed dependencies: NVIDIA-vgx-uvp conflicts with NVIDIA-vGPU-kepler-
uvp-210.0-352.70.x86_64
[root@uvp ~]#
```

Resolution

Uninstall the older vGPU RPM before installing the latest driver.

Use the following command to uninstall the older vGPU RPM:

```
[root@uvp ~]# rpm -e NVIDIA-vgx-uvp
```

Chapter 4.

RESOLVED ISSUES

Issues Resolved in Release 5.0

No resolved issues are reported in this release for Huawei UVP.

Issues Resolved in Release 5.1

No resolved issues are reported in this release for Huawei UVP.

Chapter 5.

KNOWN ISSUES

5.1. Since 5.1: Issues in remote desktop sessions if a license is acquired after a session is started

Description

A VM might acquire a license for NVIDIA vGPU software after a remote desktop session has connected to the VM. In this situation, some licensed features and capabilities are not available to a properly licensed vGPU or pass-through GPU in the session. For example, the updated maximum resolution supported is not available.

Workaround

Before attempting this workaround, confirm that the VM has obtained the correct license for NVIDIA vGPU software.

1. After installing the guest VM driver package and configuring required license settings on the VM (or on the master image used for VM deployment), add the following registry setting:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\NVIDIA Corporation\Global\GridLicensing]
"IgnoreSP"=dword:00000001
```

2. Restart the VM.

Status

Open

Ref.

200391532

5.2. License settings configured through a GPO are ignored

Description

License settings configured through a Windows Group Policy Object (GPO) are ignored. Windows Registry settings applied through a GPO are set after the NVIDIA vGPU software

display driver service is started. Therefore, NVIDIA vGPU software cannot be configured through a GPO.

Workaround

Use the Registry Editor to set the Windows Registry keys for license settings individually.

Status

Open

Ref.

2010398

5.3. Licensing pop-up windows contain the text `microsoft.explorer.notification`

Description

On Windows 10 Creators Update (1703), licensing pop-up windows contain the text `microsoft.explorer.notification`.

Version

Windows 10 Creators Update (1703)

Status

Open

Ref.

200346607

5.4. Since 5.2: The license expires prematurely in Linux guest VMs

Description

In Linux guest VMs, the license expires before the default borrow period has elapsed. In normal operation, the license is renewed periodically at an interval that depends on the license borrow period. As a result, a failure to renew the license may cause the license to expire before the default borrow period has elapsed.

Workaround

To reduce the possibility of license-renewal failures caused by transient network issues, increase the license borrow period to a value of about 7 days.

Status

Open

Ref.

200376678

5.5. Multiple display heads are not detected by Ubuntu 14.04 guest VMs

Description

After an Ubuntu 14.04 guest VM has acquired a license, multiple display heads connected to the VM are not detected.

Version

Ubuntu 14.04

Workaround

To see all the connected display heads after the VM has acquired a license, open the **Displays** settings window and click **Detect displays**.

Status

Open

Ref. #

200334648

5.6. Since 5.1: On GPUs based on the Pascal architecture, Ubuntu 16.04 VMs run slowly after acquiring a license

Description

On GPUs based on the Pascal architecture, Ubuntu VMs to which an NVIDIA vGPU or pass-through GPU is assigned run slowly after acquiring a license. Ubuntu VMs that have not been assigned an NVIDIA vGPU or pass-through GPU run noticeably faster.

Workaround

After the VM has acquired a license, restart the `lightdm` service.

Status

Open.

Ref. #

200359618

5.7. Resolution is not updated after a VM acquires a license and is restarted

Description

In a Red Enterprise Linux 7.3 guest VM, an increase in resolution from 1024×768 to 2560×1600 is not applied after a license is acquired and the `gridd` service is restarted. This issue occurs if the `multimonitor` parameter is added to the `xorg.conf` file.

Version

Red Enterprise Linux 7.3

Status

Open

Ref. #

200275925

5.8. NVIDIA vGPU encoder and process utilization counters don't work with Windows Performance Counters

Description

GPU encoder and process utilization counter groups are listed in Windows Performance Counters, but no instances of the counters are available. The counters are disabled by default and must be enabled.

Workaround

Enable the counters by running the following sequence of commands from a command shell:

```
wmic /namespace:nv path System call enableProcessUtilizationPerfCounter
```

```
wmic /namespace:nv path System call enableEncoderSessionsPerfCounter
```

If you need to disable the counters, run the following sequence of commands from a command shell:

```
wmic /namespace:nv path System call disableProcessUtilizationPerfCounter
```

```
wmic /namespace:nv path System call disableEncoderSessionsPerfCounter
```

Status

Open

Ref. #

1971698

5.9. A segmentation fault in DBus code causes `nvidia-gridd` to exit on Red Hat Enterprise Linux and CentOS

Description

On Red Hat Enterprise Linux 6.8 and 6.9, and CentOS 6.8 and 6.9, a segmentation fault in DBus code causes the `nvidia-gridd` service to exit.

The `nvidia-gridd` service uses DBus for communication with **NVIDIA X Server Settings** to display licensing information through the **Manage License** page. Disabling the GUI for licensing resolves this issue.

Since 5.1: The GUI for licensing is disabled by default.

Version

Red Hat Enterprise Linux 6.8 and 6.9

CentOS 6.8 and 6.9

NVIDIA vGPU software 5.0

5.0 Only: Workaround

This workaround requires `sudo` privileges.

1. As root, edit the `/etc/nvidia/gridd.conf` file to set the `EnableUI` option to `FALSE`.
2. Start the `nvidia-gridd` service.

```
# sudo service nvidia-gridd start
```

3. Confirm that the `nvidia-gridd` service has obtained a license by examining the log messages written to `/var/log/messages`.

```
# sudo grep gridd /var/log/messages
```

```
...
```

```
Aug 5 15:40:06 localhost nvidia-gridd: Started (4293)
```

```
Aug 5 15:40:24 localhost nvidia-gridd: License acquired successfully.
```

Status

Open

Ref.

- ▶ 200358191

- ▶ 200319854
- ▶ 1895945

5.10. Since 5.1: No **Manage License** option available in **NVIDIA X Server Settings** by default

Description

By default, the **Manage License** option is not available in **NVIDIA X Server Settings**. This option is missing because the GUI for licensing on Linux is disabled by default to work around the issue that is described in [A segmentation fault in DBus code causes nvidia-gridd to exit on Red Hat Enterprise Linux and CentOS](#).

Version

NVIDIA vGPU software 5.1

Workaround

This workaround requires `sudo` privileges.



Do not use this workaround with Red Hat Enterprise Linux 6.8 and 6.9 or CentOS 6.8 and 6.9. To prevent a segmentation fault in DBus code from causing the `nvidia-gridd` service from exiting, the GUI for licensing must be disabled with these OS versions.

1. If **NVIDIA X Server Settings** is running, shut it down.
2. If the `/etc/nvidia/gridd.conf` file does not already exist, create it by copying the supplied template file `/etc/nvidia/gridd.conf.template`.
3. As root, edit the `/etc/nvidia/gridd.conf` file to set the `EnableUI` option to `TRUE`.
4. Start the `nvidia-gridd` service.

```
# sudo service nvidia-gridd start
```

When **NVIDIA X Server Settings** is restarted, the **Manage License** option is now available.

Status

Open

5.11. Since 5.1: The `nvidia-gridd` service fails because the required configuration is not provided

Description

The `nvidia-gridd` service exits with an error because the required configuration is not provided.

The known issue described in [A segmentation fault in Dbus code causes nvidia-gridd to exit on Red Hat Enterprise Linux and CentOS](#) causes the **NVIDIA X Server Settings** page for managing licensing settings through a GUI to be disabled by default. As a result, if the required license configuration is not provided through the configuration file, the service exits with an error.

Details of the error can be obtained by checking the status of the `nvidia-gridd` service.

```
# service nvidia-gridd status
nvidia-gridd.service - NVIDIA Grid Daemon
Loaded: loaded (/usr/lib/systemd/system/nvidia-gridd.service; enabled; vendor
       preset: disabled)
       Active: failed (Result: exit-code) since Wed 2017-11-01 19:25:07 IST; 27s ago
       Process: 11990 ExecStopPost=/bin/rm -rf /var/run/nvidia-gridd (code=exited,
       status=0/SUCCESS)
       Process: 11905 ExecStart=/usr/bin/nvidia-gridd (code=exited, status=0/SUCCESS)
Main PID: 11906 (code=exited, status=1/FAILURE)
Nov 01 19:24:35 localhost.localdomain systemd[1]: Starting NVIDIA Grid Daemon...
Nov 01 19:24:35 localhost.localdomain nvidia-gridd[11906]: Started (11906)
Nov 01 19:24:35 localhost.localdomain systemd[1]: Started NVIDIA Grid Daemon.
Nov 01 19:24:36 localhost.localdomain nvidia-gridd[11906]: Failed to open
       config file : /etc/nvidia/gridd.conf error :No such file or directory
Nov 01 19:25:07 localhost.localdomain nvidia-gridd[11906]: Service provider
       detection complete.
Nov 01 19:25:07 localhost.localdomain nvidia-gridd[11906]: Shutdown (11906)
Nov 01 19:25:07 localhost.localdomain systemd[1]: nvidia-gridd.service: main
       process exited, code=exited, status=1/FAILURE
Nov 01 19:25:07 localhost.localdomain systemd[1]: Unit nvidia-gridd.service
       entered failed state.
Nov 01 19:25:07 localhost.localdomain systemd[1]: nvidia-gridd.service failed.
```

Workaround

Use a configuration file to license NVIDIA vGPU software on Linux as explained in [Virtual GPU Client Licensing User Guide](#).

Status

Open

Ref. #

200359469

5.12. Since 5.1: The **Apply** button is disabled after change to unlicensed mode

Description

After the mode is changed from licensed **Quadro Virtual Datacenter Workstation Edition** mode to **Unlicensed Tesla** mode, the **Apply** button on the **Manage GRID License** page is disabled. As a result, **NVIDIA X Server Settings** cannot be used to switch to Tesla (Unlicensed) mode on a licensed system.

Workaround

1. Start **NVIDIA X Server Settings** by using the method for launching applications provided by your Linux distribution.
2. In the **NVIDIA X Server Settings** window that opens, click **Manage GRID License**.
3. Clear the **Primary Server** field.
4. Select the **Tesla (unlicensed)** option.
5. Click **Apply**.

Status

Open

Ref. #

200359624

5.13. Licenses remain checked out when VMs are forcibly powered off

Description

NVIDIA vGPU software licenses remain checked out on the license server when non-persistent VMs are forcibly powered off.

The NVIDIA service running in a VM returns checked out licenses when the VM is shut down. In environments where non-persistent licensed VMs are not cleanly shut down, licenses on the license server can become exhausted. For example, this issue can occur in automated test environments where VMs are frequently changing and are not

guaranteed to be cleanly shut down. The licenses from such VMs remain checked out against their MAC address for seven days before they time out and become available to other VMs.

Resolution

If VMs are routinely being powered off without clean shutdown in your environment, you can avoid this issue by shortening the license borrow period. To shorten the license borrow period, set the `LicenseInterval` configuration setting in your VM image. For details, refer to *Virtual GPU Client Licensing User Guide*.

Status

Closed

Ref.

1694975

5.14. Multiple WebGL tabs in Microsoft Internet Explorer may trigger TDR on Windows VMs

Description

Running intensive WebGL applications in multiple IE tabs may trigger a TDR on Windows VMs.

Workaround

Disable hardware acceleration in IE.

To enable software rendering in IE, refer to the Microsoft knowledge base article [How to enable or disable software rendering in Internet Explorer](#).

Status

Open

Ref.

200148377

Notice

ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE.

Information furnished is believed to be accurate and reliable. However, NVIDIA Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use. No license is granted by implication of otherwise under any patent rights of NVIDIA Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all other information previously supplied. NVIDIA Corporation products are not authorized as critical components in life support devices or systems without express written approval of NVIDIA Corporation.

HDMI

HDMI, the HDMI logo, and High-Definition Multimedia Interface are trademarks or registered trademarks of HDMI Licensing LLC.

OpenCL

OpenCL is a trademark of Apple Inc. used under license to the Khronos Group Inc.

Trademarks

NVIDIA, the NVIDIA logo, NVIDIA GRID, vGPU, Pascal, Quadro, and Tesla are trademarks or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

© 2013-2018 NVIDIA Corporation. All rights reserved.