Virtual GPU Software R535 for Microsoft Windows Server

Release Notes

Table of Contents

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 Microsoft Windows Server.

> 💬 **Note:** The most current version of the documentation for this release of NVIDIA vGPU software can be found online at [NVIDIA Virtual GPU Software Documentation](#).

1.1. NVIDIA vGPU Software Driver Versions

Each release in this release family of NVIDIA vGPU software includes a specific version of the NVIDIA Windows driver and NVIDIA Linux driver.

NVIDIA vGPU Software Version	NVIDIA Windows Driver Version	NVIDIA Linux Driver Version
16.5	538.46	535.161.08
16.4	538.33	535.161.07
16.3	538.15	535.154.05
16.2	537.70	535.129.03
16.1	537.13	535.104.05
16.0	536.25	535.54.03

> 💬 **Note:** You must use [NVIDIA License System](#) with every release in this release family of NVIDIA vGPU software. All releases in this release family of NVIDIA vGPU software are **incompatible** with all releases of the NVIDIA vGPU software license server.

1.2. Updates in Release 16.5

NVIDIA vGPU software 16.5 resolves an issue that affects graphics cards that are supported only by NVIDIA AI Enterprise.

1.3. Updates in Release 16.4

New Features in Release 16.4

▶ Security updates - see *Security Bulletin: NVIDIA GPU Display Driver - February 2024*, which is posted shortly after the release date of this software and is listed on the NVIDIA Product Security page

▶ Miscellaneous bug fixes

Newly Supported Hardware and Software in Release 16.4

▶ Newly supported guest OS releases:

▶ Microsoft Windows 11 23H2

1.4. Updates in Release 16.3

New Features in Release 16.3

▶ Miscellaneous bug fixes

Hardware and Software Support Introduced in Release 16.3

▶ Newly supported graphics cards:

▶ NVIDIA L2

▶ NVIDIA L20

▶ Newly supported guest OS releases:

▶ Red Hat Enterprise Linux 8.9

1.5. Updates in Release 16.2

New Features in Release 16.2

▶ Security updates - see *Security Bulletin: NVIDIA GPU Display Driver - October 2023*, which is posted shortly after the release date of this software and is listed on the NVIDIA Product Security page

▶ Miscellaneous bug fixes

1.6. Updates in Release 16.1

New Features in Release 16.1

▶ Miscellaneous bug fixes

Hardware and Software Support Introduced in Release 16.1

▶ Support for the for the following GPUs:

  ▶ NVIDIA L40S

  ▶ NVIDIA RTX 5000 Ada

▶ Support for SUSE Linux Enterprise Server 12 SP5 as a guest OS

1.7. Updates in Release 16.0

New Features in Release 16.0

▶ Security updates - see *Security Bulletin: NVIDIA GPU Display Driver - June 2023*, which is posted shortly after the release date of this software and is listed on the [NVIDIA Product Security](#) page

▶ Miscellaneous bug fixes

Newly Supported Hardware and Software in Release 16.0

▶ Newly supported guest OSes:

  ▶ Red Hat Enterprise Linux 8.8

Feature Support Withdrawn in Release 16.0

▶ Graphics cards no longer supported:

  ▶ Graphics cards that support only C-series vGPUs, namely:

    ▶ NVIDIA H800 PCIe 80GB

    ▶ NVIDIA H100 PCIe 80GB

    ▶ NVIDIA A800 PCIe 80GB

    ▶ NVIDIA A800 PCIe 80GB liquid cooled

    ▶ NVIDIA A800 HGX 80GB

    ▶ NVIDIA A100 PCIe 80GB

    ▶ NVIDIA A100 PCIe 80GB liquid cooled

- ▶ NVIDIA A100X
- ▶ NVIDIA A100 HGX 80GB
- ▶ NVIDIA A100 PCIe 40GB
- ▶ NVIDIA A100 HGX 40GB
- ▶ NVIDIA A30
- ▶ NVIDIA A30X

Instead, these graphics cards are supported with NVIDIA AI Enterprise.

- ▶ Guest OSes no longer supported:

  - ▶ Red Hat Enterprise Linux 8.7 and 8.4

Chapter 2. Validated Platforms

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

2.1. Supported NVIDIA GPUs and Validated Server Platforms

This release of NVIDIA vGPU software on Microsoft Windows Server provides support for several NVIDIA GPUs running on validated server hardware platforms.

For a list of validated server platforms, refer to NVIDIA Virtual GPU Certified Servers.

The supported products for each type of NVIDIA vGPU software deployment depend on the GPU.

> **Note:** All GPUs that support graphics acceleration are supported as a secondary device in a bare-metal deployment. Tesla M6 is also supported as the primary display device in a bare-metal deployment.

GPUs Based on the NVIDIA Ada Lovelace Architecture

GPU	Supported NVIDIA vGPU Software Products[1],[2],[3]	
	NVIDIA vGPU	DDA
Since 16.1: NVIDIA L40S	N/A	▶ vWS ▶ vApps
NVIDIA L40	N/A	▶ vWS ▶ vApps
Since 16.3: NVIDIA L20	N/A	▶ vWS ▶ vApps

	Supported NVIDIA vGPU Software Products[1,2,3]	
GPU	NVIDIA vGPU	DDA
NVIDIA L4	N/A	▶ vWS ▶ vApps
Since 16.3: NVIDIA L2	N/A	▶ vWS ▶ vApps
NVIDIA RTX 6000 Ada	N/A	▶ vWS ▶ vApps
Since 16.1: NVIDIA RTX 5000 Ada	N/A	▶ vWS ▶ vApps

GPUs Based on the NVIDIA Ampere Architecture

	Supported NVIDIA vGPU Software Products[1,2,3]	
GPU	NVIDIA vGPU	DDA
NVIDIA A40[4]	N/A	▶ vWS ▶ vApps
NVIDIA A16	N/A	▶ vWS ▶ vApps
NVIDIA A10	N/A	▶ vWS ▶ vApps
NVIDIA A2	N/A	▶ vWS ▶ vApps
NVIDIA RTX A6000[4]	N/A	▶ vWS ▶ vApps
NVIDIA RTX A5500[4]	N/A	▶ vWS ▶ vApps
NVIDIA RTX A5000[4]	N/A	▶ vWS ▶ vApps

GPUs Based on the NVIDIA Turing Architecture

GPU	Supported NVIDIA vGPU Software Products[1,2,3]	
	NVIDIA vGPU	DDA
Tesla T4	N/A	▶ vWS ▶ vApps
Quadro RTX 6000 [4]	N/A	▶ vWS ▶ vApps
Quadro RTX 6000 passive[4]	N/A	▶ vWS ▶ vApps
Quadro RTX 8000[4]	N/A	▶ vWS ▶ vApps
Quadro RTX 8000 passive[4]	N/A	▶ vWS ▶ vApps

GPUs Based on the NVIDIA Volta Architecture

GPU	Supported NVIDIA vGPU Software Products[1,2,3]	
	NVIDIA vGPU	DDA
Tesla V100 SXM2	N/A	▶ vWS ▶ vApps
Tesla V100 SXM2 32GB	N/A	▶ vWS ▶ vApps
Tesla V100 PCIe	N/A	▶ vWS ▶ vApps
Tesla V100 PCIe 32GB	N/A	▶ vWS ▶ vApps
Tesla V100S PCIe 32GB	N/A	▶ vWS ▶ vApps
Tesla V100 FHHL	N/A	▶ vWS

	Supported NVIDIA vGPU Software Products[1,2,3]	
GPU	NVIDIA vGPU	DDA
		▶ vApps

GPUs Based on the NVIDIA Pascal™ Architecture

	Supported NVIDIA vGPU Software Products[1,2,3]	
GPU	NVIDIA vGPU	DDA
Tesla P4	N/A	▶ vWS ▶ vApps
Tesla P6	N/A	▶ vWS ▶ vApps
Tesla P40	N/A	▶ vWS ▶ vApps
Tesla P100 PCIe 16 GB	N/A	▶ vWS ▶ vApps
Tesla P100 SXM2 16 GB	N/A	▶ vWS ▶ vApps
Tesla P100 PCIe 12GB	N/A	▶ vWS ▶ vApps

GPUs Based on the NVIDIA Maxwell Graphic Architecture

	Supported NVIDIA vGPU Software Products[1,2,3]	
GPU	NVIDIA vGPU	DDA
Tesla M6	N/A	▶ vWS ▶ vApps
Tesla M10	N/A	▶ vWS ▶ vApps
Tesla M60	N/A	▶ vWS ▶ vApps

2.1.1. Switching the Mode of a GPU that Supports Multiple Display Modes

Some GPUs support display-off and display-enabled modes but must be used in NVIDIA vGPU software deployments in display-off mode.

The GPUs listed in the following table support multiple display modes. As shown in the table, some GPUs are supplied from the factory in display-off mode, but other GPUs are supplied in a display-enabled mode.

GPU	Mode as Supplied from the Factory
NVIDIA A40	Display-off
NVIDIA L40	Display-off
Since 16.1: NVIDIA L40S	Display-off
Since 16.3: NVIDIA L20	Display-off
Since 16.1: NVIDIA RTX 5000 Ada	Display enabled
NVIDIA RTX 6000 Ada	Display enabled
NVIDIA RTX A5000	Display enabled
NVIDIA RTX A5500	Display enabled
NVIDIA RTX A6000	Display enabled

A GPU that is supplied from the factory in display-off mode, such as the NVIDIA A40 GPU, might be in a display-enabled mode if its mode has previously been changed.

To change the mode of a GPU that supports multiple display modes, use the `displaymodeselector` tool, which you can request from the [NVIDIA Display Mode Selector Tool](#) page on the NVIDIA Developer website.

> **Note:** Only the GPUs listed in the table support the `displaymodeselector` tool. Other GPUs that support NVIDIA vGPU software do not support the `displaymodeselector` tool and, unless otherwise stated, do not require display mode switching.

---

[1] The supported products are as follows:

   ▶   vWS: NVIDIA RTX Virtual Workstation

   ▶   vPC: NVIDIA Virtual PC

   ▶   vApps: NVIDIA Virtual Applications

[2] N/A indicates that the deployment is not supported.
[3] vApps is supported only on Windows operating systems.
[4] This GPU is supported only in displayless mode. In displayless mode, local physical display connectors are disabled.

2.1.2. Switching the Mode of a Tesla M60 or M6 GPU

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. If you are unsure which mode your GPU is in, use the `gpumodeswitch` tool to find out the mode.

> **Note:**
>
> Only Tesla M60 and M6 GPUs support the `gpumodeswitch` tool. Other GPUs that support NVIDIA vGPU do not support the `gpumodeswitch` tool and, except as stated in Switching the Mode of a GPU that Supports Multiple Display Modes, do not require mode switching.
>
> Even in compute mode, Tesla M60 and M6 GPUs do **not** support NVIDIA Virtual Compute Server vGPU types.

For more information, refer to *gpumodeswitch User Guide*.

2.2. Hypervisor Software Releases

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

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

Software	Version Supported
Microsoft Windows Server 2022	Windows Server 2022 with Hyper-V role
Microsoft Windows Server 2019	Windows Server 2019 with Hyper-V role

2.3. Guest OS Support

NVIDIA vGPU software supports several Windows releases and Linux distributions as a guest OS using GPU pass-through.

Microsoft Windows Server with Hyper-V role supports GPU pass-through over Microsoft Virtual PCI bus. This bus is supported through paravirtualized drivers.

> **Note:**

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.

NVIDIA vGPU software supports **only** 64-bit guest operating systems. No 32-bit guest operating systems are supported.

2.3.1. Windows Guest OS Support

NVIDIA vGPU software supports **only** the 64-bit Windows releases listed as a guest OS on Microsoft Windows Server.

Note:

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

2.3.1.1. Windows Guest OS Support in Release 16.5

▶ Windows Server 2022

▶ Windows Server 2019

▶ Windows 11 23H2 and all Windows 11 releases supported by Microsoft up to and including this release

▶ Windows 10 2022 Update (22H2) and all Windows 10 releases supported by Microsoft up to and including this release

2.3.1.2. Windows Guest OS Support in Release 16.4

▶ Windows Server 2022

▶ Windows Server 2019

▶ Windows 11 23H2 and all Windows 11 releases supported by Microsoft up to and including this release

▶ Windows 10 2022 Update (22H2) and all Windows 10 releases supported by Microsoft up to and including this release

2.3.1.3. Windows Guest OS Support in Release 16.3

▶ Windows Server 2022

▶ Windows Server 2019

▶ Windows 11 22H2 and all Windows 11 releases supported by Microsoft up to and including this release

▶ Windows 10 2022 Update (22H2) and all Windows 10 releases supported by Microsoft up to and including this release

2.3.1.4. Windows Guest OS Support in Release 16.2

▶ Windows Server 2022

▶ Windows Server 2019

▶ Windows 11 22H2 and all Windows 11 releases supported by Microsoft up to and including this release

▶ Windows 10 2022 Update (22H2) and all Windows 10 releases supported by Microsoft up to and including this release

2.3.1.5. Windows Guest OS Support in Release 16.1

▶ Windows Server 2022

▶ Windows Server 2019

▶ Windows 11 22H2 and all Windows 11 releases supported by Microsoft up to and including this release

▶ Windows 10 2022 Update (22H2) and all Windows 10 releases supported by Microsoft up to and including this release

2.3.1.6. Windows Guest OS Support in Release 16.0

▶ Windows Server 2022

▶ Windows Server 2019

▶ Windows 11 22H2 and all Windows 11 releases supported by Microsoft up to and including this release

▶ Windows 10 2022 Update (22H2) and all Windows 10 releases supported by Microsoft up to and including this release

2.3.2. Linux Guest OS Support

NVIDIA vGPU software supports **only** the 64-bit Linux distributions listed as a guest OS on Microsoft Windows Server.

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

2.3.2.1. Linux Guest OS Support in Release 16.5

▶ Red Hat Enterprise Linux 8.9

▶ Red Hat Enterprise Linux 8.8

▶ Red Hat Enterprise Linux 8.6

▶ Red Hat Enterprise Linux 7.9

▶ Rocky Linux 8.4

▶ **Deprecated:** CentOS Linux 8 (2105)

▶ **Deprecated:** CentOS 7.6-7.8

▶ Ubuntu 20.04 LTS

▶ SUSE Linux Enterprise Server 15 SP2

▶ SUSE Linux Enterprise Server 12 SP5

▶ SUSE Linux Enterprise Server 12 SP2

2.3.2.2.   Linux Guest OS Support in Release 16.4

▶ Red Hat Enterprise Linux 8.9

▶ Red Hat Enterprise Linux 8.8

▶ Red Hat Enterprise Linux 8.6

▶ Red Hat Enterprise Linux 7.9

▶ Rocky Linux 8.4

▶ **Deprecated:** CentOS Linux 8 (2105)

▶ **Deprecated:** CentOS 7.6-7.8

▶ Ubuntu 20.04 LTS

▶ SUSE Linux Enterprise Server 15 SP2

▶ SUSE Linux Enterprise Server 12 SP5

▶ SUSE Linux Enterprise Server 12 SP2

2.3.2.3.   Linux Guest OS Support in Release 16.3

▶ Red Hat Enterprise Linux 8.9

▶ Red Hat Enterprise Linux 8.8

▶ Red Hat Enterprise Linux 8.6

▶ Red Hat Enterprise Linux 7.9

▶ Rocky Linux 8.4

▶ **Deprecated:** CentOS Linux 8 (2105)

▶ **Deprecated:** CentOS 7.6-7.8

▶ Ubuntu 20.04 LTS

▶ SUSE Linux Enterprise Server 15 SP2

▶ SUSE Linux Enterprise Server 12 SP5

▶ SUSE Linux Enterprise Server 12 SP2

2.3.2.4. Linux Guest OS Support in Release 16.2

- ▶ Red Hat Enterprise Linux 8.8
- ▶ Red Hat Enterprise Linux 8.6
- ▶ Red Hat Enterprise Linux 7.9
- ▶ Rocky Linux 8.4
- ▶ **Deprecated:** CentOS Linux 8 (2105)
- ▶ **Deprecated:** CentOS 7.6-7.8
- ▶ Ubuntu 20.04 LTS
- ▶ SUSE Linux Enterprise Server 15 SP2
- ▶ SUSE Linux Enterprise Server 12 SP5
- ▶ SUSE Linux Enterprise Server 12 SP2

2.3.2.5. Linux Guest OS Support in Release 16.1

- ▶ Red Hat Enterprise Linux 8.8
- ▶ Red Hat Enterprise Linux 8.6
- ▶ Red Hat Enterprise Linux 7.9
- ▶ Rocky Linux 8.4
- ▶ **Deprecated:** CentOS Linux 8 (2105)
- ▶ **Deprecated:** CentOS 7.6-7.8
- ▶ Ubuntu 20.04 LTS
- ▶ SUSE Linux Enterprise Server 15 SP2
- ▶ SUSE Linux Enterprise Server 12 SP5
- ▶ SUSE Linux Enterprise Server 12 SP2

2.3.2.6. Linux Guest OS Support in Release 16.0

- ▶ Red Hat Enterprise Linux 8.8
- ▶ Red Hat Enterprise Linux 8.6
- ▶ Red Hat Enterprise Linux 7.9
- ▶ Rocky Linux 8.4
- ▶ **Deprecated:** CentOS Linux 8 (2105)
- ▶ **Deprecated:** CentOS 7.6-7.8
- ▶ Ubuntu 20.04 LTS
- ▶ SUSE Linux Enterprise Server 15 SP2

▶ SUSE Linux Enterprise Server 12 SP2

Chapter 3. Known Issues

3.1. 16.0, 16.1 Only: NVWMI functions for faking EDID have no effect

Description

The NVIDIA Enterprise Management Toolkit (NVWMI) functions for faking Extended Display Identification Data (EDID), namely, `fakeEDID`, `fakeEDIDAll`, and `fakeEDIDOnPort` have no effect. This issue affects only Windows guest VMs and can prevent a VM from being enabled with multiple displays. When this issue occurs, `unable to fake EDID` events can be seen in **Event Viewer**.

Status

Resolved in NVIDIA vGPU software 16.2

Ref.

4309888

3.2. 16.0 Only: The NVIDIA L40 GPU brand is incorrectly identified if GSP firmware is disabled

Description

If GPU System Processor (GSP) firmware is disabled, the NVIDIA Virtual GPU Manager incorrectly identifies the brand of the NVIDIA L40 GPU. This incorrect identification of the GPU brand might cause performance degradation with some applications that are optimised for features of the NVIDIA L40 that are not available in the incorrect brand. However, the output from the `nvidia-smi` command is **not** affected.

This issue occurs only if GPU System Processor (GSP) firmware is disabled. It does not occur if GSP firm is enabled.

Status

Resolved in NVIDIA vGPU software 16.1

Ref.

4142288

3.3. NVIDIA Control Panel crashes if a user session is disconnected and reconnected

Description

On all supported Windows Server guest OS releases, **NVIDIA Control Panel** crashes if a user session is disconnected and then reconnected while **NVIDIA Control Panel** is open.

Version

This issue affects all supported Windows Server guest OS releases.

Status

Open

Ref.

4086605

3.4. NVIDIA vGPU software graphics driver for Windows sends a remote call to `ngx.download.nvidia.com`

Description

After the NVIDIA vGPU software graphics for windows has been installed in the guest VM, the driver sends a remote call to `ngx.download.nvidia.com` to download and install additional components. Such a remote call might be a security issue.

Workaround

Before running the NVIDIA vGPU software graphics driver installer, disable the remote call to `ngx.download.nvidia.com` by setting the following Windows registry key:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\NVIDIA Corporation\Global\NGXCore]
"EnableOTA"=dword:00000000
```

> **Note:** If this Windows registry key is set to 1 or deleted, the remote call to `ngx.download.nvidia.com` is enabled again.

Status

Open

Ref.

4031840

3.5. Multiple RDP session reconnections on Windows Server 2022 can consume all frame buffer

Description

Multiple RDP session reconnections in a Windows Server 2022 guest VM can consume all the frame buffer of a vGPU or physical GPU. When this issue occurs, users' screens becomes black, their sessions are disconnected but left intact, and they cannot log on again. The following error message is written to the event log on the hypervisor host:

```
The Desktop Window Manager process has exited.
(Process exit code: 0xe0464645, Restart count: 1, Primary display device ID: )
```

Version

This issue affects only the Windows Server 2022 guest OS.

Workaround

Periodically restart the Windows Server 2022 guest VM to prevent all frame buffer from being consumed.

Status

Open

Ref. #

3583766

3.6. NLS client fails to acquire a license with the error `The allowed time to process response has expired`

Description

A licensed client of NVIDIA License System (NLS) fails to acquire a license with the error `The allowed time to process response has expired`. This error can affect clients of a Cloud License Service (CLS) instance or a Delegated License Service (DLS) instance.

This error occurs when the time difference between the system clocks on the client and the server that hosts the CLS or DLS instance is greater than 10 minutes. A common cause of this error is the failure of either the client or the server to adjust its system clock when daylight savings time begins or ends. The failure to acquire a license is expected to prevent clock windback from causing licensing errors.

Workaround

Ensure that system clock time of the client and any server that hosts a DLS instance match the current time in the time zone where they are located.

To prevent this error from occurring when daylight savings time begins or ends, enable the option to automatically adjust the system clock for daylight savings time:

- ▶ **Windows:** Set the **Adjust for daylight saving time automatically** option.
- ▶ **Linux:** Use the `hwclock` command.

Status

Not a bug

Ref.

3859889

3.7.　With multiple active sessions, **NVIDIA Control Panel** incorrectly shows that the system is unlicensed

Description

In an environment with multiple active desktop sessions, the **Manage License** page of **NVIDIA Control Panel** shows that a licensed system is unlicensed. However, the `nvidia-smi` command and the management interface of the NVIDIA vGPU software license server correctly show that the system is licensed. When an active session is disconnected and reconnected, the **NVIDIA Display Container** service crashes.

The **Manage License** page incorrectly shows that the system is unlicensed because of stale data in **NVIDIA Control Panel** in an environment with multiple sessions. The data is stale because **NVIDIA Control Panel** fails to get and update the settings for remote sessions when multiple sessions or no sessions are active in the VM. The **NVIDIA Display Container** service crashes when a session is reconnected because the session is not active at the moment of reconnection.

Status

Open

Ref.

3761243

3.8.　VP9 and AV1 decoding with web browsers are not supported on Microsoft Windows Server 2019

Description

VP9 and AV1 decoding with web browsers are not supported on Microsoft Windows Server 2019 and later supported releases. This issue occurs because starting with Windows Server 2019, the required codecs are not included with the OS and are not available through the **Microsoft Store** app. As a result, hardware decoding is not available

for viewing YouTube videos or using collaboration tools such as Google Meet in a web browser.

Version

This issue affects Microsoft Windows Server releases starting with Windows Server 2019.

Status

Not an NVIDIA bug

Ref.

200756564

3.9.   After an upgrade of the Linux graphics driver from an RPM package in a licensed VM, licensing fails

Description

After the NVIDIA vGPU software graphics driver for Linux is upgraded from an RPM package in a licensed VM, licensing fails. The `nvidia-smi vgpu -q` command shows the driver version and license status as N/A. Restarting the `nvidia-gridd` service fails with a `Unit not found` error.

Workaround

Perform a clean installation of the NVIDIA vGPU software graphics driver for Linux from an RPM package.

1. Remove the currently installed driver.
2. Install the new version of the driver.
   ```
   $ rpm -iv nvidia-linux-grid-525_535.161.08_amd64.rpm
   ```

Status

Open

Ref.

3512766

3.10. After an upgrade of the Linux graphics driver from a Debian package, the driver is not loaded into the VM

Description

After the NVIDIA vGPU software graphics driver for Linux is upgraded from a Debian package, the driver is not loaded into the VM.

Workaround

Use one of the following workarounds to load the driver into the VM:

▶ Reboot the VM.

▶ Remove the `nvidia` module from the Linux kernel and reinsert it into the kernel.

1. Remove the `nvidia` module from the Linux kernel.
   ```
   $ sudo rmmod nvidia
   ```
2. Reinsert the `nvidia` module into the Linux kernel.
   ```
   $ sudo modprobe nvidia
   ```

Status

Not a bug

Ref.

200748806

3.11. A licensed client might fail to acquire a license if a proxy is set

Description

If a proxy is set with a system environment variable such as `HTTP_PROXY` or `HTTPS_PROXY`, a licensed client might fail to acquire a license.

Workaround

Perform this workaround on each affected licensed client.

1. Add the address of the NVIDIA vGPU software license server to the system environment variable `NO_PROXY`.

   The address must be specified exactly as it is specified in the client's license server settings either as a fully-qualified domain name or an IP address. If the `NO_PROXY` environment variable contains multiple entries, separate the entries with a comma (`,`).

   If high availability is configured for the license server, add the addresses of the primary license server and the secondary license server to the system environment variable `NO_PROXY`.

2. Restart the NVIDIA driver service that runs the core NVIDIA vGPU software logic.

   ▶ On Windows, restart the **NVIDIA Display Container** service.

   ▶ On Linux, restart the `nvidia-gridd` service.

Status

Closed

Ref.

200704733

3.12. Disconnected sessions cannot be reconnected or might be reconnected very slowly with NVWMI installed

Description

Disconnected sessions cannot be reconnected or might be reconnected very slowly when the NVIDIA Enterprise Management Toolkit (NVWMI) is installed. This issue affects Citrix Virtual Apps and Desktops and VMware Horizon sessions on Windows guest VMs.

Workaround

Uninstall NVWMI.

Status

Open

Ref. #

3262923

3.13. NVIDIA Control Panel fails to start if launched too soon from a VM without licensing information

Description

If NVIDIA licensing information is not configured on the system, any attempt to start **NVIDIA Control Panel** by right-clicking on the desktop within 30 seconds of the VM being started fails.

Workaround

Restart the VM and wait at least 30 seconds before trying to launch **NVIDIA Control Panel**.

Status

Open

Ref.

200623179

3.14. Citrix Virtual Apps and Desktops session corruption occurs in the form of residual window borders

Description

When a window is dragged across the desktop in a Citrix Virtual Apps and Desktops session, corruption of the session in the form of residual window borders occurs.

Version

This issue affects only Citrix Virtual Apps and Desktops version 7 2003

Workaround

Use Citrix Virtual Apps and Desktops version 7 1912 or 2006.

Status

Not an NVIDIA bug

Ref.

200608675

3.15.   On Linux, the frame rate might drop to 1 after several minutes

Description

On Linux, the frame rate might drop to 1 frame per second (FPS) after NVIDIA vGPU software has been running for several minutes. Only some applications are affected, for example, `glxgears`. Other applications, such as Unigine Heaven, are not affected. This behavior occurs because Display Power Management Signaling (DPMS) for the Xorg server is enabled by default and the display is detected to be inactive even when the application is running. When DPMS is enabled, it enables power saving behavior of the display after several minutes of inactivity by setting the frame rate to 1 FPS.

Workaround

1. If necessary, stop the Xorg server.

    ```
    # /etc/init.d/xorg stop
    ```

2. In a plain text editor, edit the `/etc/X11/xorg.conf file` to set the options to disable DPMS and disable the screen saver.

    a). In the `Monitor` section, set the `DPMS` option to `false`.

    ```
    Option "DPMS" "false"
    ```

    b). At the end of the file, add a `ServerFlags` section that contains option to disable the screen saver.

    ```
    Section "ServerFlags"
        Option "BlankTime" "0"
      EndSection
    ```

    c). Save your changes to `/etc/X11/xorg.conf file` and quit the editor.

3. Start the Xorg server.

    ```
    # etc/init.d/xorg start
    ```

Status

Open

Ref.

200605900

3.16. Microsoft DDA fails with some GPUs

Description

Microsoft Discrete Device Assignment (DDA) fails with GPUs that have more than 16 GB of GPU memory. After the NVIDIA vGPU software graphics driver is installed in the guest VM, a second display device appears on the GPU and the driver prompts for a reboot. After the reboot, the device disappears and the Microsoft Hyper-V Video device appears.

This issue occurs because less memory-mapped input/output (MMIO) space is configured for the operating system than the device requires.

Workaround

Perform this workaround in a **Windows Power Shell** window on the hypervisor host.

Set the upper MMIO space to the amount that the device requires to allow all of the MMIO to be mapped. Upper MMIO space starts at approximately 64 GB in address space.

```
Set-VM –HighMemoryMappedIoSpace mmio-space –VMName vm-name
```
mmio-space
   The amount of MMIO space that the device requires, appended with the appropriate unit of measurement, for example, **64GB** for 64 GB of MMIO space.

   The required amount of MMIO space depends on the amount of BAR1 memory on the installed GPUs and the number of GPUs assigned to the VM as follows:

   mmio-space = 2 # gpu-bar1-memory # assigned-gpus

gpu-bar1-memory
   The amount of BAR1 memory on one of the installed GPUs. For example, in a server in which eight GPUs are installed and each GPU has 32 GB of BAR1 memory, *gpu-bar1-memory* is 32 GB.
assigned-gpus
   The number of GPUs assigned to the VM.
vm-name
   The name of the VM to which the GPU is assigned.

The following example sets the upper MMIO space to 64 GB for the VM named `mygpuvm`, to which one GPU with 32 GB of BAR1 memory is assigned.

```
Set-VM –HighMemoryMappedIoSpace 64GB –VMName mygpuvm
```

For more information, see [Deploy graphics devices using Discrete Device Assignment](#) on the Microsoft technical documentation site.

Status

Not an NVIDIA bug

Ref.

2812853

3.17. DWM crashes randomly occur in Windows VMs

Description

Desktop Windows Manager (DWM) crashes randomly occur in Windows VMs, causing a blue-screen crash and the bug check `CRITICAL_PROCESS_DIED`. Computer Management shows problems with the primary display device.

Version

This issue affects Windows 10 1809, 1903 and 1909 VMs.

Status

Not an NVIDIA bug

Ref.

2730037

3.18. NVIDIA vGPU software graphics driver fails after Linux kernel upgrade with DKMS enabled

Description

After the Linux kernel is upgraded (for example by running `sudo apt full-upgrade`) with Dynamic Kernel Module Support (DKMS) enabled, the `nvidia-smi` command fails to run. If DKMS is enabled, an upgrade to the Linux kernel triggers a rebuild of the NVIDIA vGPU software graphics driver. The rebuild of the driver fails because the compiler version is incorrect. Any attempt to reinstall the driver fails because the kernel fails to build.

When the failure occurs, the following messages are displayed:

```
-> Installing DKMS kernel module:
      ERROR: Failed to run `/usr/sbin/dkms build -m nvidia -v  535.54.03 -k
5.3.0-28-generic`:
      Kernel preparation unnecessary for this kernel. Skipping...
      Building module:
      cleaning build area...
      'make' -j8 NV_EXCLUDE_BUILD_MODULES='' KERNEL_UNAME=5.3.0-28-generic
IGNORE_CC_MISMATCH='' modules...(bad exit status: 2)
      ERROR (dkms apport): binary package for nvidia:  535.54.03 not found
      Error! Bad return status for module build on kernel: 5.3.0-28-generic
(x86_64)
      Consult /var/lib/dkms/nvidia/ 535.54.03/build/make.log for more information.
      -> error.
      ERROR: Failed to install the kernel module through DKMS. No kernel module
was installed;
      please try installing again without DKMS, or check the DKMS logs for more
information.
      ERROR: Installation has failed. Please see the file '/var/log/nvidia-
installer.log' for details.
      You may find suggestions on fixing installation problems in the README
available on the Linux driver download page at www.nvidia.com.
```

Workaround

When installing the NVIDIA vGPU software graphics driver with DKMS enabled, use one of the following workarounds:

▶ Before running the driver installer, install the `dkms` package, then run the driver installer with the `-dkms` option.

▶ Run the driver installer with the `--no-cc-version-check` option.

Status

Not a bug.

Ref. #

2836271

3.19. Blue screen crash occurs or no devices are found after VM reset

Description

If a VM on Microsoft Windows Server with Hyper-V role is reset from the hypervisor host, a blue screen crash (BSOD) occurs on Windows VMs and the `nvidia-smi` command reports `No devices were found` on Linux VMs. This issue occurs only on Windows Server 2019 with Tesla T4 GPUs with SRIOV enabled, Quadro RTX 8000 passive GPUs, and Quadro RTX 6000 passive GPUs.

Workaround

Contact NVIDIA Enterprise Support for a workaround for this issue, referencing the knowledge base article *Workaround for Blue Screen Crashes On Hyper-V DDA With SRIOV-Enabled GPUs*. This article is available only to NVIDIA Enterprise Support personnel.

Status

Not an NVIDIA bug

Ref. #

200567935

3.20. Frame capture while the interactive logon message is displayed returns blank screen

Description

Because of a known limitation with NvFBC, a frame capture while the interactive logon message is displayed returns a blank screen.

An NvFBC session can capture screen updates that occur after the session is created. Before the logon message appears, there is no screen update after the message is shown

and, therefore, a black screen is returned instead. If the NvFBC session is created after this update has occurred, NvFBC cannot get a frame to capture.

Workaround

Press **Enter** or wait for the screen to update for NvFBC to capture the frame.

Status

Not a bug

Ref.

2115733

3.21. RDS sessions do not use the GPU with some Microsoft Windows Server releases

Description

When some releases of Windows Server are used as a guest OS, Remote Desktop Services (RDS) sessions do not use the GPU. With these releases, the RDS sessions by default use the Microsoft Basic Render Driver instead of the GPU. This default setting enables 2D DirectX applications such as Microsoft Office to use software rendering, which can be more efficient than using the GPU for rendering. However, as a result, 3D applications that use DirectX are prevented from using the GPU.

Version

- ▶ Windows Server 2019
- ▶ Windows Server 2016
- ▶ Windows Server 2012

Solution

Change the local computer policy to use the hardware graphics adapter for all RDS sessions.

1. Choose **Local Computer Policy** > **Computer Configuration** > **Administrative Templates** > **Windows Components** > **Remote Desktop Services** > **Remote Desktop Session Host** > **Remote Session Environment** .
2. Set the **Use the hardware default graphics adapter for all Remote Desktop Services sessions** option.

3.22.   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.

To prevent this issue, the GUI for licensing is disabled by default. You might encounter this issue if you have enabled the GUI for licensing and are using Red Hat Enterprise Linux 6.8 or 6.9, or CentOS 6.8 and 6.9.

Version

Red Hat Enterprise Linux 6.8 and 6.9

CentOS 6.8 and 6.9

Status

Open

Ref.

- 200358191
- 200319854
- 1895945

3.23.　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 <u>A segmentation fault in DBus code causes nvidia-gridd to exit on Red Hat Enterprise Linux and CentOS</u>.

Workaround

This workaround requires `sudo` privileges.

> 📃 **Note:** 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.

If you are licensing a physical GPU for vCS, you **must** use the configuration file `/etc/nvidia/gridd.conf`.

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

3.24. 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