



NVIDIA License System

User Guide

Table of Contents

Chapter 1. Introduction to NVIDIA License System.....	1
1.1. Introduction to NVIDIA Software Licensing.....	1
1.2. About Service Instances.....	2
1.2.1. About Cloud License Service (CLS) Instances.....	2
1.2.2. About Delegated License Service (DLS) Instances.....	3
1.3. About the NVIDIA Licensing Portal.....	3
1.4. High Availability for NVIDIA License System DLS Instances.....	4
1.5. Heartbeat Checks for DLS Failover.....	5
1.5.1. Recovery Actions for DLS Failover.....	6
1.5.2. Restarting the DLS Virtual Appliance.....	7
Chapter 2. Installing and Configuring the DLS Virtual Appliance.....	9
2.1. Platform Requirements for a DLS Virtual Appliance or Containerized DLS Software Image.....	10
2.2. Host Name Resolution Requirements for a DLS Virtual Appliance.....	10
2.3. Communications Ports Requirements.....	13
2.4. Sizing Guidelines for a DLS Appliance.....	16
2.4.1. Throughput for a DLS Appliance.....	16
2.4.2. Scalability for a DLS Appliance.....	16
2.4.3. Burst Load Performance for a DLS Appliance.....	18
2.5. Installing a VM-Based DLS Virtual Appliance.....	18
2.5.1. Installing the DLS Virtual Appliance Image on a Supported Hypervisor.....	19
2.5.1.1. Installing the DLS Virtual Appliance on Citrix Hypervisor.....	19
2.5.1.2. Installing the DLS Virtual Appliance on Microsoft Windows Server with Hyper-V.....	20
2.5.1.3. Installing the DLS Virtual Appliance on Red Hat Enterprise Linux KVM.....	22
2.5.1.4. Installing the DLS Virtual Appliance on Red Hat Virtualization.....	24
2.5.1.5. Installing the DLS Virtual Appliance on Ubuntu Hypervisor.....	26
2.5.1.6. Installing the DLS Virtual Appliance on VMware vSphere.....	28
2.5.2. Setting the IP Address of a DLS Virtual Appliance from the Hypervisor.....	29
2.5.3. Changing the IP Address of a VMware VM Set During DLS Appliance Installation.....	31
2.6. Deploying a Containerized DLS Software Image.....	31
2.6.1. Contents of the Containerized DLS Software Image Download.....	32
2.6.2. Setting Properties for a Containerized DLS Software Image.....	32
2.6.2.1. Setting Environment Variables to Specify String Properties.....	32
2.6.2.2. Specifying Integer Values as Environment Variables.....	33

2.6.2.3. Specifying Integer Values as Actual Values.....	35
2.6.3. Deploying the Containerized DLS Software Image on a Supported Platform.....	37
2.6.3.1. Deploying the Containerized DLS Software Image on Docker.....	37
2.6.3.2. Deploying the Containerized DLS Software Image on Kubernetes Platforms.....	38
2.6.3.3. Deploying the Containerized DLS Software Image on Podman.....	39
2.6.4. Volume Mappings for a Containerized DLS Software Image.....	41
2.6.5. Troubleshooting Issues with Deploying the Containerized DLS Software Image.....	42
2.6.5.1. Containerized DLS Appliance Fails to Start on Kubernetes Platforms.....	42
2.6.5.2. Volume Mode Access for DLS Appliance Resources on Kubernetes Platforms Is Incorrect.....	42
2.6.5.3. Data Validation Errors Prevent the DLS Appliance Container Pod from Starting on Kubernetes.....	43
2.6.5.4. The Management Interface of the NVIDIA Licensing Application Fails to Load.....	44
2.6.5.5. HA Configuration or Online Upgrade Fails.....	44
2.6.5.6. Windows Client Cannot Return Licenses Because the Return Endpoint Is Unavailable.....	45
2.6.5.7. Client Cannot Obtain License if Client Configuration Token Specifies the Domain Name.....	45
2.6.5.8. Node Health Information Is Missing from the Service Instance Page.....	45
2.6.5.9. The podman-compose Command Fails with Missing Networks Error.....	46
2.6.5.10. Re-deploying a Containerized DLS Software Image on Red Hat OpenShift Container Platform Fails.....	46
2.7. Installing the DLS Software Image on Red Hat Enterprise Linux OS.....	47
2.8. Configuring User Accounts on a DLS Virtual Appliance.....	47
2.8.1. Registering the DLS Administrator User.....	48
2.8.2. Retrieving the DLS Administrator User's Reset Secret.....	49
2.8.3. Changing the DLS Administrator User Name and Password.....	50
2.8.4. Creating the DLS sudo User Account.....	51
2.8.5. Integrating a DLS Instance with an LDAP Server.....	52
2.8.6. Logging in to a DLS Virtual Appliance.....	53
2.8.7. Gathering Log Files for Analysis and Troubleshooting.....	54
2.9. Configuring an HA Cluster of DLS Instances.....	54
2.9.1. Setting the Maximum Cluster Size for a DLS Instance.....	55
2.9.1.1. Setting the Maximum Cluster Size for a VM-Based DLS Instance.....	55
2.9.1.2. Setting the Maximum Cluster Size for a Containerized DLS Instance.....	55
2.9.2. Creating or Expanding an HA Cluster of DLS Instances.....	56

2.9.3. Removing a Node from an HA Cluster.....	58
2.9.4. Marking a Node as the Primary Node in an HA Cluster.....	58
2.10. Configuring a VM-Based DLS Virtual Appliance.....	59
2.10.1. Setting the Static IP Address of a DLS Virtual Appliance.....	59
2.10.2. Reverting the Network Configuration of a DLS Appliance to DHCP.....	61
2.10.3. Changing the Host Name of a VM-Based DLS Virtual Appliance.....	62
2.10.4. Expanding the Disk Space on a DLS Virtual Appliance.....	62
Chapter 3. Configuring a Service Instance.....	64
3.1. Roles Required for Configuring a Service Instance.....	65
3.2. Proxy Server Requirements and Firewall Rules for a CLS Instance.....	66
3.3. Changing the Name and Description of a DLS Instance.....	67
3.4. Creating a License Server on the NVIDIA Licensing Portal.....	67
3.5. Performing an Express CLS Installation.....	70
3.6. Converting a Legacy NVIDIA vGPU Software License Server to an NLS License Server.....	71
3.6.1. Converting a Legacy License Server to an NLS License Server on an Express CLS Installation.....	72
3.6.2. Converting a Legacy License Server to an NLS License Server on a Custom CLS Instance.....	73
3.6.3. Converting a Legacy License Server to an NLS License Server on a DLS Instance.....	74
3.7. Creating or Registering a Service Instance.....	75
3.7.1. Creating a CLS Instance on the NVIDIA Licensing Portal.....	75
3.7.2. Registering an on-Premises DLS Instance with the NVIDIA Licensing Portal.....	77
3.7.3. Enabling your Organization to Register DLS Instances Manually.....	78
3.7.4. Registering a DLS Instance on an Air-Gapped Network with the NVIDIA Licensing Portal.....	79
3.8. Deleting a Service Instance.....	81
3.9. Binding a License Server to a Service Instance.....	83
3.10. Freeing a License Server from a Service Instance.....	84
3.11. Installing a License Server on a Service Instance.....	85
3.11.1. Installing a License Server on a CLS Instance.....	85
3.11.2. Installing a License Server on a DLS Instance.....	86
3.12. Changing the Leasing Mode of a License Server.....	88
Chapter 4. Managing Licenses on a License Server.....	89
4.1. Where to Perform Tasks for Managing Licenses.....	89
4.2. Roles Required for Managing Licenses on a CLS Instance.....	90
4.3. Navigating to the License Server Details Page for a License Server.....	90

4.4. Managing License Pools.....	91
4.4.1. Creating a License Pool.....	91
4.4.2. Deleting a License Pool.....	93
4.4.3. Managing Licenses and Licensed Products in a License Pool.....	94
4.4.4. Merging Two License Pools.....	96
4.4.5. Migrating Licenses Between License Pools.....	98
4.5. Managing Fulfillment Conditions.....	101
4.5.1. About Match Conditions.....	101
4.5.2. Creating a Fulfillment Condition.....	102
4.5.3. Deleting a Fulfillment Condition.....	105
4.5.4. Editing a Fulfillment Condition.....	106
4.5.5. Changing the Order of Fulfillment Conditions.....	109
4.6. Generating a Client Configuration Token.....	110
4.6.1. Generating a Client Configuration Token for a CLS Instance.....	111
4.6.2. Generating a Client Configuration Token for a DLS Instance.....	114
4.7. Generating Node-Locked Licenses.....	117
4.8. Disabling and Enabling a License Server, License Pool, or Fulfillment Condition.....	118
4.9. Editing License Server Settings.....	119
4.10. Manually Releasing Leases from a Server.....	122
4.10.1. Manual Release of Specific Clients Licensed to an NLS Service Instance.....	123
4.10.2. Manual Force Bulk Release of All Clients Licensed to an NLS Service Instance.....	124
4.10.3. Forcibly Releasing a Node-Locked License.....	125
4.11. Supporting Non-Persistent Desktop Pools.....	126
4.11.1. Supporting Non-Persistent Desktop Pools from a CLS Instance.....	126
4.11.2. Supporting Non-Persistent Desktop Pools from a DLS Instance.....	127
Chapter 5. Configuring a Licensed Client of NVIDIA License System.....	129
5.1. Configuring a Licensed Client with a Networked License.....	129
5.1.1. Configuring a Licensed Client with a Networked License on Windows with Default Settings.....	130
5.1.2. Configuring a Licensed Client with a Networked License on Linux with Default Settings.....	130
5.1.3. Configuring a Licensed Client with a Networked License with Custom Settings.....	131
5.1.3.1. Configuring a Licensed Client with a Networked License on Windows with Custom Settings.....	132
5.1.3.2. Configuring a Licensed Client with a Networked License on Linux with Custom Settings.....	134
5.1.4. Generating an Encrypted Credentials File.....	137
5.1.4.1. Generating an Encrypted Credentials File on Windows.....	138

5.1.4.2. Generating an Encrypted Credentials File on Linux.....	139
5.2. Configuring a Licensed Client with an NVIDIA vGPU Software Node-Locked License.....	139
5.2.1. Configuring a Licensed Client with a Node-Locked License on Windows.....	140
5.2.2. Configuring a Licensed Client with a Node-Locked License on Linux.....	141
5.3. Verifying the NVIDIA vGPU Software License Status of a Licensed Client.....	142
Chapter 6. Administering a Service Instance.....	144
6.1. Migrating a DLS Instance.....	144
6.1.1. Upgrading a DLS Appliance in Place.....	147
6.1.1.1. Initiating an In-Place Upgrade to a DLS Appliance.....	147
6.1.1.2. Upgrading a Container-Based DLS Appliance in Place.....	147
6.1.1.3. Upgrading a VM-Based DLS Appliance in Place.....	149
6.1.1.4. Troubleshooting the In-Place Upgrade of a DLS Appliance.....	151
6.1.2. Performing a File-Based Migration of a DLS Instance.....	151
6.1.2.1. Initiating an Upgrade on a DLS Instance.....	152
6.1.2.2. Preparing a New Appliance for a Standalone DLS Instance.....	152
6.1.2.3. Preparing a New Appliance for a VM-Based Node in an HA Cluster.....	153
6.1.2.4. Generating a Migration File for the DLS Instance that You are Migrating...	153
6.1.2.5. Transferring Migration Data to the DLS Instance on an Upgraded Virtual Appliance.....	154
6.1.2.6. Synchronizing Changes with the Primary Node in an HA Cluster.....	155
6.2. Setting the Validity Period of a Lease Authorization Token for a Service Instance..	156
6.2.1. Setting the Validity Period of a Lease Authorization Token for a CLS Instance..	156
6.2.2. Setting the Validity Period of a Lease Authorization Token for a DLS Instance..	158
6.3. Specifying the Retention Period for Client Registrations.....	160
6.4. Gathering License Usage Statistics.....	160
6.5. Configuring Email Alerts for Licensing Events.....	161
6.5.1. Disk Space Threshold Configuration.....	168
6.6. Setting the Retention Period of Events on a DLS Instance.....	169
6.7. Configuring a DLS Virtual Appliance with a Third-Party Signed SSL Certificate.....	169
6.7.1. Obtaining a Third-Party Signed SSL Certificate for a DLS Virtual Appliance.....	170
6.7.2. Installing a Third-Party Signed SSL Certificate on a DLS Virtual Appliance.....	170
6.8. Hiding Organization and Virtual Group Information on a DLS Instance.....	171
6.9. Reconfiguring the Rsyslog Tool in a DLS Virtual Appliance.....	172
6.10. Configuring NTP on a DLS Virtual Appliance.....	173
6.11. Setting the Maximum Size of the PostgreSQL Database Volume.....	174
6.12. Troubleshooting a DLS Instance.....	174
6.12.1. Log File Locations and Types for a DLS Virtual Appliance.....	175

6.12.2. Storing Log Files for a DLS Virtual Appliance on a Network File Share.....	175
6.12.3. Exporting and Importing Event Records from a DLS Instance.....	177
6.12.4. Restarting a DLS Instance's Internal Services.....	178
6.12.5. Recovering from an Incorrect DLS Name or Address in a Client Configuration Token.....	178
6.12.6. Recovering from an HA Configuration Failure in an IPv6 Network.....	179
6.12.7. DLS has overlapping disk partitions.....	179
Chapter 7. Managing License Servers on the NVIDIA Licensing Portal.....	180
7.1. Where to Perform Tasks for Managing a License Server.....	180
7.2. Roles Required for Managing License Servers on the NVIDIA Licensing Portal.....	181
7.3. Managing Licenses and Licensed Products on a License Server.....	181
7.4. Returning Licenses from a License Server on a DLS Instance to the NVIDIA Licensing Portal.....	183
7.5. Moving a License Server to Another Virtual Group.....	184
7.6. Deleting a License Server.....	184
7.7. Editing Default Service Instances.....	185
7.7.1. Edit the Service Instance Designated as Default.....	187
Chapter 8. Managing Contacts on the NVIDIA Licensing Portal.....	189
8.1. Role-Based Access to an Organization and Virtual Groups.....	189
8.1.1. Organization Administrator.....	190
8.1.2. Organization User.....	190
8.1.3. Virtual Group Administrator.....	191
8.1.4. Virtual Group User.....	192
8.2. Roles Required for Managing Contacts on the NVIDIA Licensing Portal.....	192
8.3. Adding a Contact on the NVIDIA Licensing Portal.....	193
8.4. Removing a Contact on the NVIDIA Licensing Portal.....	194
8.5. Changing the Role of a Contact on the NVIDIA Licensing Portal.....	194
8.6. Requesting Access to the NVIDIA Enterprise Support Portal.....	195
Chapter 9. Managing Virtual Groups.....	196
9.1. Roles for Managing Virtual Groups.....	196
9.2. Creating a Virtual Group.....	197
9.3. Deleting a Virtual Group.....	198
9.4. Adding a Contact to a Virtual Group.....	198
9.5. Removing a Contact from a Virtual Group.....	199
9.6. Managing Entitlements in a Virtual Group.....	199
9.7. Sample Business Scenario for Virtual Groups.....	200
Appendix A. Migrating Licenses from a Legacy NVIDIA vGPU Software License Server.....	202

A.1. Tasks for Preparing to Migrate Licenses.....	202
A.2. CLS Instances Only: Tasks for Configuring CLS Instances.....	202
A.3. DLS Instances Only: Tasks for Installing and Configuring the DLS Virtual Appliance.....	203
A.4. DLS Instances Only: Tasks for Configuring DLS Instances.....	204
A.5. Tasks for Managing Licenses on a License Server.....	204
A.6. Tasks for Configuring a Licensed Client.....	205
A.7. Tasks for Decommissioning a Legacy NVIDIA vGPU software License Server.....	205
A.7.1. Uninstalling the NVIDIA vGPU Software License Server on Windows.....	205
A.7.2. Uninstalling the NVIDIA vGPU Software License Server on Linux.....	207
Appendix B. Getting Started with the NLS RESTful APIs.....	209
B.1. Getting Started with the CLS RESTful API.....	209
B.1.1. Creating a Licensing State API Key.....	209
B.1.2. Getting Information Required in Other RESTful API Calls to a CLS Instance.....	210
B.1.3. Getting License Server Information for a CLS Instance.....	211
B.1.4. Listing Licenses that Are Being Served by a CLS Instance.....	212
B.2. Getting Started with the DLS RESTful API.....	213
B.2.1. Exporting and Importing API Keys for a DLS Instance.....	213
B.2.1.1. Exporting API Keys for a DLS Instance.....	214
B.2.1.2. Importing API Keys for a DLS Instance.....	214
B.2.2. Logging in to a DLS Appliance from the RESTful API.....	214
B.2.3. Getting Information Required in Other RESTful API Calls to a DLS Instance.....	215
B.2.4. Getting License Server Information for a DLS Instance.....	216
B.2.5. Listing Licenses that Are Being Served by a DLS Instance.....	217
Appendix C. Performance Data for a CLS Instance.....	219
C.1. Throughput for a CLS Virtual Appliance.....	219
C.2. Scalability for a CLS Instance.....	219
C.3. Burst Load Performance for a CLS Instance.....	220
C.4. NVIDIA Cloud Licensing Service (CLS) Service Level Objectives.....	220

List of Figures

Figure 1. Starting the Uninstaller from Windows Control Panel.....	206
Figure 2. Running the License Server Uninstaller on Windows.....	207
Figure 3. Running the License Server Uninstaller on Linux.....	208

Chapter 1. Introduction to NVIDIA License System

The NVIDIA License System is used to provide software licenses to licensed NVIDIA software products. The licenses that the NVIDIA License System provides are obtained from the NVIDIA Licensing Portal.



Note: NVIDIA vGPU software releases earlier than 13.0 do **not** support NVIDIA License System. For full details of NVIDIA vGPU software releases that support NVIDIA License System, refer to [NVIDIA License System Release Notes](#).

1.1. Introduction to NVIDIA Software Licensing

To activate licensed functionalities, a licensed client must obtain a software license when it is booted.

NVIDIA License System supports the types of licensing for licensed clients:

- ▶ **Networked-licensing:** A client with a network connection obtains a license by leasing it from a NVIDIA License System service instance. The service instance serves the license to the client over the network from a pool of floating licenses obtained from the NVIDIA Licensing Portal. The license is returned to the service instance when the licensed client no longer requires the license.
- ▶ **Node locked-licensing:** A client system without a network connection or on an air-gapped network can obtain a node-locked NVIDIA vGPU software license from a file installed locally on the client system.



Note: Support for node-locked licensing was introduced in NVIDIA vGPU software 15.0. It is **not** supported in earlier NVIDIA vGPU software releases.

1.2. About Service Instances

A service instance is required to serve licenses to licensed clients.

NVIDIA License System supports the following types of service instances:

- ▶ **Cloud License Service (CLS) instance.** A CLS instance is hosted on the NVIDIA Licensing Portal.
- ▶ **Delegated License Service (DLS) instance.** A DLS instance is hosted on-premises at a location that is accessible from your private network, such as inside your data center.

To provide isolation for performance, security, and ease of administration, you can deploy multiple service instances as needed. For example, you can deploy service instances in distinct physical locations by deploying a DLS instance in each of your data centers. You can also use a mixture of CLS and DLS instances to serve your licenses to licensed clients.

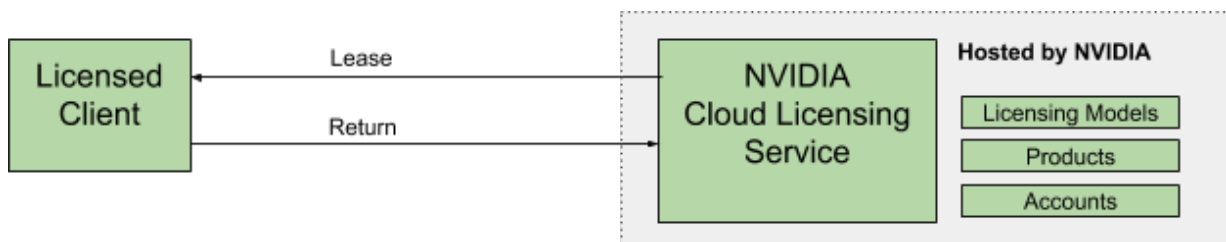


Note: You **cannot** mix CLS and DLS instances in a high availability (HA) cluster of service instances. Each node in an HA cluster **must** be a DLS instance.

1.2.1. About Cloud License Service (CLS) Instances

A Cloud License Service (CLS) instance is hosted on the NVIDIA Licensing Portal.

Because a CLS instance is hosted on the NVIDIA Licensing Portal, you do not need to download licenses from the NVIDIA Licensing Portal and upload them to the instance.

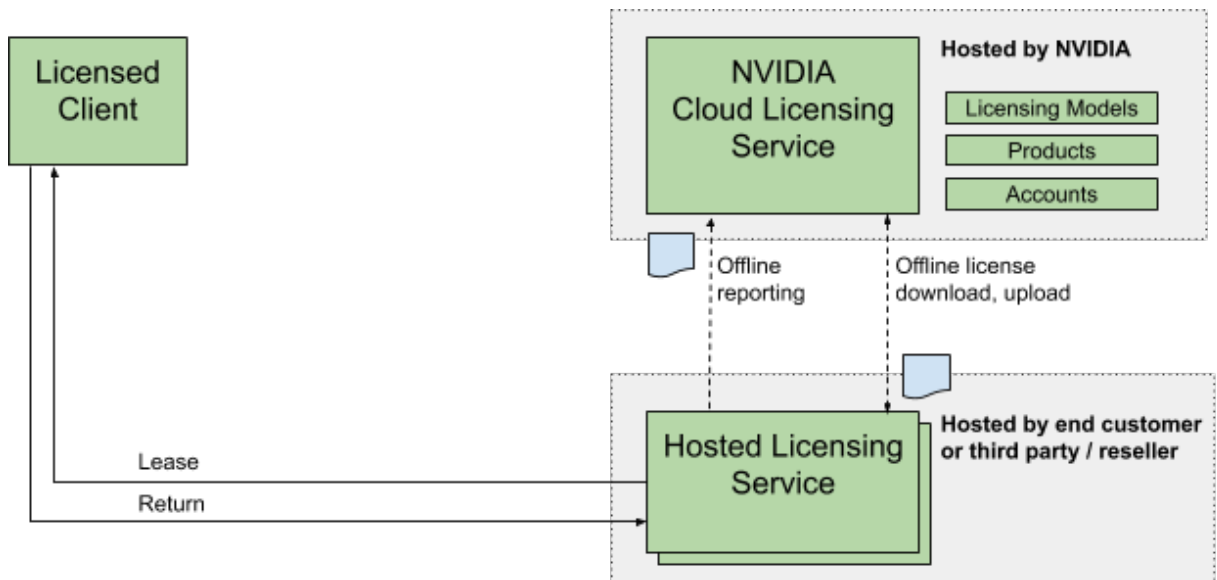


Hosting a CLS instance on a cloud service provides robustness and dynamic scalability for the CLS instance. Because a CLS instance is maintained by NVIDIA and the cloud service provider, feature and maintenance updates are generally transparent to users.

1.2.2. About Delegated License Service (DLS) Instances

A Delegated License Service (DLS) instance is hosted on-premises at a location that is accessible from your private network, such as inside your data center.

Because a DLS instance is fully disconnected from the NVIDIA Licensing Portal, you must download licenses from the NVIDIA Licensing Portal and upload them to the instance manually.



1.3. About the NVIDIA Licensing Portal

The NVIDIA Licensing Portal provides access to the entitlements that you purchased and the licenses that they contain.

To be able to download NVIDIA vGPU software licenses, you must create at least one license server on the NVIDIA Licensing Portal and allocate licenses in your entitlements to the server. You can also distribute your licenses across multiple license servers as necessary, add new licensed products to an existing server, and delete license servers that you no longer require.

To help you manage your entitlements and licenses on the NVIDIA Licensing Portal, you can add other users as registered contacts in the organization associated with your NVIDIA Enterprise Account. To secure your entitlements and licenses, NVIDIA Licensing Portal provides role-based access for all registered contacts. For more information, see [Managing Contacts on the NVIDIA Licensing Portal](#).

By default, all entitlements are associated with a top-level organization and are accessible to all contacts in the organization. If you need to allow only specific groups

of contacts within your organization to access specific entitlements, you can partition your entitlements into isolated segments. However, if a single collection of entitlements that spans your entire organization meets your business needs, you can leave all your entitlements in the top-level organization.

To partition your entitlements into isolated segments, NVIDIA Licensing Portal provides the ability to create virtual groups and assign entitlements and contacts to them. For more information, see [Managing Virtual Groups](#).

1.4. High Availability for NVIDIA License System DLS Instances

To provide licensed clients with continued access to licenses if a DLS instance fails, you can configure DLS instances for high availability.

High availability requires at least two DLS instances in a *failover* configuration:

- ▶ A *primary* DLS instance, which is actively serving licenses to licensed clients
- ▶ One or more *secondary* DLS instances, which act as a backup for the primary DLS instance



Note:

- ▶ NVIDIA License System supports a maximum cluster size of nine DLS instances.
- ▶ Support for clusters of more than two instances was introduced in NVIDIA vGPU software 16.1. Earlier NVIDIA vGPU software releases support clusters of only **two** instances.

Configuring multiple DLS instances in a failover configuration increases availability because simultaneous failure of multiple instances is rare. The primary and secondary DLS instances work together to ensure that licenses in the enterprise remain continually available to licensed clients.

If the primary DLS instance fails, failover occurs. One of the secondary DLS instances becomes the primary instance and begins to serve licenses. In a cluster of more than two instances, the secondary instance that has the longest uptime becomes the primary instance. The DLS instance that failed becomes a secondary instance when it is returned to service. The next time that failover occurs, the primary DLS instance becomes a secondary DLS instance again.



Note: To ensure that licenses in the enterprise remain continually available after failure of the primary DLS instance, return the failed DLS instance to service as quickly as possible to restore high availability support. After failure of a DLS instance in a cluster of two instances, the remaining instance becomes a single point of failure.

During normal operation, the primary DLS instance continually updates the secondary DLS instances with information about the licenses that are being served to clients.

For more information about configuring DLS instances for high availability, see [Configuring an HA Cluster of DLS Instances](#).

1.5. Heartbeat Checks for DLS Failover

In a High Availability (HA) cluster, the Heartbeat service, running on every node, monitors the health of DLS appliances and triggers a failover when an unhealthy node is detected based on one of the following criteria. In the event of a failover, one of the secondary nodes assumes the role of the primary node to serve the requests, and the previous primary node takes on the role of the secondary node.

Failover Criteria

- ▶ **Service health check using health end point of service:**
 - ▶ This check verifies whether all services are running in a healthy state and are responsive.
 - ▶ Every service has a health endpoint. The Heartbeat service will monitor the endpoint to check if the service is responsive.
- ▶ **Database connection check:**
 - ▶ The database connection check verifies whether the database is responsive.
 - ▶ As part of this check, a lightweight query will be executed.
 - ▶ If the database is not responsive, it will result in a client request failure.
- ▶ **Resource exhaustion check:**
 - ▶ This check verifies whether any resources (CPU, RAM, and disk) have exceeded the predefined thresholds.
 - ▶ Exhaustion of resources impacts system stability, so this is a proactive action for a failover.
- ▶ **Application failure:**
 - ▶ Application failure check verifies whether client requests are serviced successfully.
 - ▶ If the percentage of failures over a predefined time interval crosses the predefined threshold, a failover gets triggered.
 - ▶ This is distinct from the service health check, which verifies if the service is responsive.
 - ▶ This check monitors failures in servicing requests. Only authentication and lease operations are monitored for application failure.
- ▶ **Secondary node monitoring of primary node health:**
 - ▶ This check verifies if the primary node is running and if the lease service is responsive.

- ▶ As part of the lease response check, a read query (`get lease count`) is executed by the lease monitoring service.

In summary, a failover occurs when any of the following conditions is detected:

- ▶ Services are not responsive.
- ▶ The database is not responsive.
- ▶ Resources are exhausted.
- ▶ Failures are observed for authentication and lease operations.
- ▶ A secondary node observes that the primary node is down.

1.5.1. Recovery Actions for DLS Failover

You can perform the following actions for recovering a DLS virtual appliance from a failover.

▶ **Resource exhaustion:**

Take appropriate actions based on the respective resources:

- ▶ For CPU or RAM exhaustion: Increase the assigned vCPU or RAM, or check the maximum supported load for a single DLS instance and reduce the load.
For the performance numbers, see [Sizing Guidelines for a DLS Appliance](#).
- ▶ For disk space exhaustion: Increase the available disk space or free up more disk space.
For the disk space cleanup script, see [Expanding the Disk Space on a DLS Virtual Appliance](#).
- ▶ Application restart might help free up RAM and CPU resources.

▶ **Database connection failure:**

- ▶ Restart the application to bring the appliance to a healthy state if the database is recoverable.
- ▶ Check for resource limits, such as memory, CPU, and disk space, to ensure they are not exhausted in unlikely scenarios where failover check of resource exhaustion does not detect in advance.
- ▶ Follow the instructions described in [Restarting the DLS Virtual Appliance](#) to restart.
- ▶ To verify if the database has been restored, you can do either of the following checks:
 - ▶ Check whether the user interface is performing correctly.
 - ▶ Check database connection by using the health endpoint after 10 minutes.

```
curl -k --location "https://<Node_IP>/service_instance_manager/v1/health?op=db"
```

The expected response code is displayed as follows:

```
{"isServiceHealthy":true}
```


- ▶ **Service health check failure:**
 - ▶ Restart the application to bring the appliance to a healthy state by following the instructions described in [Restarting the DLS Virtual Appliance](#).
- ▶ **Application failure:**
 - ▶ Restart the application to bring the appliance to a healthy state by following the instructions described in [Restarting the DLS Virtual Appliance](#).
- ▶ **Power failure:**
 - ▶ Ensure that power is restored.

If none of the preceding actions can recover the appliance successfully, contact NVIDIA Enterprise Support.

1.5.2. Restarting the DLS Virtual Appliance

To restart a DLS virtual appliance, choose an appropriate action based on your appliance.

VM-Based DLS Appliances

There are two methods to initiate the restart:

- ▶ Method 1: Reboot the VM.
- ▶ Method 2: Log in using the `rsu_admin` user, and then execute the `systemctl` command to restart `applicationStartup`.


```
sudo systemctl restart applicationStartup
```

Container-Based DLS Appliances

▶ Docker containers:

Run the following command to restart.

```
$ docker-compose -f /path/to/docker-compose.yml restart
```

▶ Kubernetes containerization platform:

Follow these steps to restart.

1. Delete pods.

```
kubectl delete deployments.apps nls-si-0
kubectl delete deployments.apps postgres-nls-si-0
```

For more information, see [The docker compose up Command](#).

2. Re-create pods.

```
kubectl create -f postgres-nls-si-0-deployment.yaml
kubectl create -f nls-si-0-deployment.yaml
```

For more information, see [KubectI Restart Pod](#).

▶ Podman containerization platform:

Run these commands to restart.

```
podman-compose --filer /path/to/deployment-file.yaml down  
podman-compose --file /path/to/deployment-file.yaml up -d
```

For more information, see [The podman restart Command](#).

Chapter 2. Installing and Configuring the DLS Virtual Appliance

To simplify the installation and administration of the DLS, the DLS is distributed as a software image to be installed or deployed on a supported platform. The DLS is a secure, hardened environment in which access to the application software is strictly controlled.

The following types of DLS software image are available:

- ▶ A virtual appliance image to be installed in a VM on a supported hypervisor
- ▶ A containerized software image for deployment on a supported container orchestration platform
- ▶ An installable package for installation on a supported OS running in a virtualized server environment on your choice of hypervisor or on a bare-metal server

Each DLS software image is configured with a single standard user account for accessing the DLS appliance and an account with `sudo` user privileges for installing updates to the DLS appliance software. Modifications to these accounts are strictly controlled. You cannot add other user accounts to the software image. However, you can use a lightweight directory access protocol (LDAP) directory instead of the configured accounts for managing user access to a DLS appliance.

In each VM-based virtual appliance image, the DLS application software is containerized to allow limited access to the OS so that non-root users can install security compliance and scanning tools in the VM. However, a container orchestration platform cannot control or restrict access to the OS on which the platform is running.

In the package for installation on a supported OS, the DLS application software is containerized to ensure that all software dependencies are met and to isolate the DLS application software from the underlying OS.

As far as possible, the DLS appliances based on all types of software image are functionally equivalent. Therefore, whether to use a container-based DLS appliance, a VM-based DLS appliance, or a DLS appliance from a package for installation on a supported OS depends on the requirements of your IT infrastructure or the policies of your IT department.

For additional guidelines, refer to the following resources from the vendors of platforms that support NVIDIA License System:

- ▶ **Citrix:** [Containers or Virtual Machines? It Doesn't Have to Be One or the Other.](#)

- ▶ **Microsoft:** [Containers vs. virtual machines](#)
- ▶ **Red Hat:** [Containers vs VMs](#)
- ▶ **VMware:** [Why use containers vs. VMs?](#)

2.1. Platform Requirements for a DLS Virtual Appliance or Containerized DLS Software Image

Before proceeding, ensure that you have a platform suitable for hosting a DLS virtual appliance or containerized DLS software image.

- ▶ The hosting platform must be a physical host running a supported hypervisor or container orchestration platform.
- ▶ The minimum resource requirements for the VM in which the DLS virtual appliance or DLS container will run is as follows:
 - ▶ **Number of vCPUs:** 4
 - ▶ **RAM:** 8 Gbytes
 - ▶ **Disk Size:** 15 Gbytes

For additional guidelines, refer to [Sizing Guidelines for a DLS Appliance](#).

- ▶ The platform must have a fixed (unchanging) IP address. The IP address may be assigned dynamically by DHCP or statically configured, but must be constant.
- ▶ The platform's date and time must be set accurately. NTP is recommended.



Note: Before proceeding with the installation, refer to [NVIDIA License System Release Notes](#) for details of supported hypervisors, details of supported container orchestration platforms, and known issues.

2.2. Host Name Resolution Requirements for a DLS Virtual Appliance

The platform that hosts a DLS virtual appliance must be identified by its IP address or its fully qualified domain name. It can also be identified by its CNAME. If you want to identify the platform by its fully qualified domain name, ensure that the required DNS entries are set before installing the DLS virtual appliance. If you want to identify the platform by its default host name, you must set a DNS entry that maps the default host name to the fully qualified domain name.

The process for setting these DNS entries is separate from the process for installing the DLS virtual appliance. Use the standard interfaces of the name resolution service that you are using to set the required DNS entries.



Tip: Whenever possible, set DNS entries for the platform that hosts a DLS virtual appliance. If the DNS entries are set, the DLS appliance can be accessed through both its IP address and its fully qualified domain name.

Determining Whether the Forward Pointer and Reverse Pointer DNS Entries Are Correct

For each mapping between a domain name and an IP address, ensure that you set both the forward pointer and reverse pointer DNS entries. A DLS virtual appliance requires the reverse pointer entry to determine the domain name of the DLS virtual appliance when creating a client configuration token.



Note: For the reverse pointer DNS entry on a Windows DNS server, use all lowercase for the fully qualified domain name.

To determine whether the forward pointer and reverse pointer DNS entries have been set correctly, type the following commands in a shell on any UNIX or Linux host on the same network as the DLS virtual appliance:

- For the forward pointer entry, type:

```
$ host domain-name
domain-name
```

The domain name for which you want to determine whether the forward pointer DNS entry is correct.

If the DNS entry has been set correctly, the command displays the IP address that is mapped to the domain name.

- For the reverse pointer entry, type:

```
$ host ip-address
ip-address
```

The IP address for which you want to determine whether the reverse pointer DNS entry is correct.

If the DNS entry has been set correctly, the command displays the domain name that is mapped to the IP address.

How a DLS Instance Determines Whether a Fully Qualified Domain Name Is Set

How a DLS instance determines whether a fully qualified domain name is set depends on the type of DLS instance.

- ▶ **VM-based DLS virtual appliance:** When the VM that hosts a DLS instance starts, the DLS instance checks whether a fully qualified domain name is mapped to the IP address of the VM. If a name is mapped to the IP address of the VM, the DLS instance retrieves the name to display in the user interface of the **NVIDIA Licensing** application on the appliance.
- ▶ **Containerized DLS Software Image:** The fully qualified domain name is set through an environment variable. When the container within which the DLS software image is deployed is started, the DLS instance checks whether this environment variable is set.



Note: Licensed clients outside the container must be able to resolve the fully qualified domain name. The environment variable **must not** specify a name that can be resolved only by the container orchestrator.

- ▶ If this environment variable is set, the DLS instance retrieves the name to display in the user interface of the **NVIDIA Licensing** application on the appliance.
- ▶ If this environment variable is not set, a user **cannot** connect to the container in which the DLS software image is deployed through a fully qualified domain name, even if a fully qualified domain name is mapped to an IP address in the DNS server.

For more information, refer to [Setting Properties for a Containerized DLS Software Image](#).

How the Host Name of the VM or Container that Hosts a DLS Instance Is Set

How the host name of the VM or container that hosts a DLS instance is set depends on the type of DLS instance.

- ▶ **VM-based DLS virtual appliance:** The host name is preset in the virtual appliance image.
 - ▶ The host name of a standalone DLS virtual appliance is preset to `nls-si-0`.
 - ▶ The host names of the DLS virtual appliances in an HA cluster are preset to `nls-si-0` and `nls-si-1`.

The host name can be changed as explained in [Changing the Host Name of a VM-Based DLS Virtual Appliance](#).

- ▶ **Containerized DLS Software Image:** The host name is set through an environment variable. If this environment variable is not set, the container orchestration platform sets the host name to the container ID. For more information, refer to [Setting Properties for a Containerized DLS Software Image](#).
- ▶ **CNAME Identification:** The platform that hosts a DLS virtual appliance can also be identified by its CNAME.

2.3. Communications Ports Requirements

To enable communication between a licensed client and a CLS or DLS instance, specific ports must be open in your firewall or proxy server. If you are using an HA cluster of DLS instances with a firewall or proxy server between the DLS instances, additional ports must be open in your firewall or proxy server.

Communications Ports Between a Licensed Client and a CLS Instance

To enable communication between a licensed client and a CLS instance, the following ports must be open in your firewall or proxy server:

Port	Protocol	Egress / Ingress	Protocol / Service	From	To
80	TLS, TCP	Egress	License release	Client	CLS
443	TLS, TCP	Egress	License acquisition License renewal	Client	CLS



Note:

- ▶ Port 80 is used by all Windows VMs to release the licenses during the time the VM is shutdown.
- ▶ Port 443 is used for returning the license at the time of license renewal and for license releases during normal operations by all VMs except Windows VMs.

Communications Ports Between a Licensed Client and a DLS Instance

To enable communication between a licensed client and a DLS instance, the following ports must be open in your firewall or proxy server:

Port	Protocol	Egress / Ingress	Protocol / Service	From	To
80	TLS, TCP	Egress	License release	Client	DLS
443	TLS, TCP	Egress	License acquisition	Client	DLS

Port	Protocol	Egress / Ingress	Protocol / Service	From	To
			License renewal		
			License release		

**Note:**

- ▶ Port 80 is used by all Windows VMs to release the licenses during the time the VM is shutdown.
- ▶ Port 443 is used for returning the license at the time of license renewal and for license releases during normal operations by all VMs except Windows VMs.
- ▶ The following ports for client to DLS connections are no longer required, but are supported for backward compatibility: 8081, 8082.
- ▶ For a container-based DLS appliance, ports 80 and 443 might not be available for the DLS appliance to bind to, either because the container orchestrator administrator has restricted access to these ports or because the ports are in use by other containers. In this situation, you must set the environment variables `DLS_HTTP_PORT` and `DLS_HTTPS_PORT` to the ports that the DLS appliance must bind to. For more information, refer to [Setting Properties for a Containerized DLS Software Image](#).

Communications Ports Between DLS Instances in an HA Cluster

If you are using an HA cluster of DLS instances with a firewall or proxy server between the DLS instances, the following ports must also be open in the firewall or proxy server:

Port	Container Instance	VM-Based Instance	Protocol	Egress/ Ingress	Protocol/ Service	From	To
443	-	#	TLS, TCP	Both	Licensing Services	Primary DLS	Secondary DLS
4369	#	#	EPMD (peer discovery)	Both	RabbitMQ, Erlang	Primary DLS	Secondary DLS
5671	#	#	AMQP, TLS	Both	RabbitMQ	Primary DLS	Secondary DLS
8080	#	#	HTTPS	Both	Licensing Services	Primary DLS	Secondary DLS
8081	#	#	HTTPS	Both	Licensing Services	Primary DLS	Secondary DLS

Port	Container Instance	VM-Based Instance	Protocol	Egress/Ingress	Protocol/Service	From	To
8082	#	#	HTTPS	Both	Licensing Services	Primary DLS	Secondary DLS
8083	#	#	HTTPS	Both	Licensing Services	Primary DLS	Secondary DLS
8084	#	#	HTTPS	Both	Licensing Services	Primary DLS	Secondary DLS
8085	#	#	HTTPS	Both	Licensing Services	Primary DLS	Secondary DLS

Port is used.

- Port is not used.

The following ports that were required to be open in earlier releases are no longer used:

- ▶ 22
- ▶ 1883
- ▶ 5672
- ▶ 8883
- ▶ 15672
- ▶ 25671
- ▶ 25672
- ▶ 61613
- ▶ 61614

Communications Ports in the Container Orchestrator

The following ports must be available in the container orchestrator to allow the DLS appliance container to bind to them:

Port	Service
8080, 18080	Administration Service
8081, 18081	Authorization Service
8082, 18082	Licensing Services
8083, 18083	File Installation
8084, 18084	Service Instance
8085	Virtual Appliance Service

Port	Service
9001	Process Manager (<code>supervisord</code>)
25672	RabbitMQ

2.4. Sizing Guidelines for a DLS Appliance

Use the measured performance numbers to determine the optimum configuration for your container-based or VM-based DLS appliances based on the expected number and frequency of requests from licensed clients.



Note: For the corresponding data for a CLS instance, refer to [Performance Data for a CLS Instance](#).

2.4.1. Throughput for a DLS Appliance

Throughput measures the number of clients that a VM-based or container-based DLS appliance can process in 1 second.



Note:

- ▶ All measurements were taken with a CPU clock speed of 3 GHz.
- ▶ For a VM-based DLS appliance, the disk size is preset to 15 Gbytes in the virtual appliance image. If you want to retain diagnostic log files for longer than the default period of three months, increase the disk size in the appliance.
- ▶ The measurements for a container-based DLS appliance were taken from a Docker container in an Ubuntu VM. Only the containers for the DLS appliance were running in the VM while the measurements were taken.
- ▶ If you increase the number of vCPUs, you must increase the total RAM in GB to twice the number of vCPUs. For a container-based DLS appliance, increasing the number of vCPUs increases the number of worker threads. If the number of vCPUs is increased without an increase in the total RAM, some workers or services might fail.

Number of vCPUs	Total RAM (GB)	RAM Consumed (GB)	Throughput (Clients per Second)	Average Response Time (ms)
4	8	4.5	5	74
6	12	6	9	84
8	16	8	13	87

2.4.2. Scalability for a DLS Appliance

Scalability measures the maximum number of licensed clients that a VM-based or container-based DLS appliance can serve in a specific interval. A DLS appliance serves a

licensed client by performing a licensing operation for the client, namely the borrowing, return, or renewal of a license. Registration of a licensed client is not considered a licensing operation because it occurs only once for any client.

These measurements capture the maximum number of licensed clients that DLS appliances with varying numbers of vCPUs and amounts of RAM can serve when licenses are borrowed for 12 hours. The maximum number of clients is directly proportional to the length of time for which licenses are borrowed. If the length of time increases, the maximum number of clients also increases.

To calculate the maximum number of clients from the length of time for which licenses are borrowed, use the following formula:

$$\text{max-clients} = \text{clients-per-second} \times 60 \times (\text{renewal-interval-percentage} / 100) \times \text{borrow-time-in-hours} \times 60$$

max-clients

The maximum number of licensed clients that a DLS appliance can serve.

clients-per-second

The number of clients per second that the DLS appliance can serve as measured by scalability runs for each configuration. Use the number of clients per second in the table for the appropriate configuration.

renewal-interval-percentage

The percentage of the length of time for which a license is borrowed to which the renewal interval is set.

When these measurements were taken, the renewal interval was set to 15% of the length of time in hours for which licenses were borrowed.

borrow-time-in-hours

The length of time in hours for which licenses are borrowed.

When these measurements were taken, licenses were borrowed for 12 hours.

This formula prevents the accumulation of requests on a DLS instance, ensuring smooth operation.



Note:

- ▶ The measurements were taken with the throughput measured in [Throughput for a DLS Appliance](#).
- ▶ All measurements were taken with a CPU clock speed of 2.6 GHz.
- ▶ The measurements for a container-based DLS appliance were taken from a Docker container in an Ubuntu VM. Only the containers for the DLS appliance were running in the VM while the measurements were taken.

Number of vCPUs	Total RAM (GB)	Clients per Second	Calculation	Maximum Number of Clients
4	8	5	$5 \times 60 \times 0.15 \times 12 \times 60$	32,000
6	12	9	$9 \times 60 \times 0.15 \times 12 \times 60$	58,000
8	16	13	$13 \times 60 \times 0.15 \times 12 \times 60$	84,000

2.4.3. Burst Load Performance for a DLS Appliance

Burst load performance measures the time that a VM-based or container-based DLS appliance requires to process the requests received from a large number of clients in a short interval of time. Burst load performance does **not** measure concurrency or throughput.



Note:

- ▶ Burst processing times are illustrative only because they are for retry logic in performance tests that use simulated client drivers. Times may differ with real client drivers.
- ▶ The test systems were configured with 4 vCPUs and 8 GB of RAM.
- ▶ The measurements for a container-based DLS appliance were taken from a Docker container in an Ubuntu VM. Only the containers for the DLS appliance were running in the VM while the measurements were taken.

Number of Clients	Interval	Processing Time
100	0-1 second	5 seconds
1,000	0-20 seconds	50 seconds
5,000	0-5 minutes	5 minutes
10,000	0-10 minutes	8 minutes and 42 seconds

2.5. Installing a VM-Based DLS Virtual Appliance

To simplify the installation process, a VM-based DLS virtual appliance is supplied as a virtual appliance image to be installed on a supported hypervisor.

After verifying the requirements in [Platform Requirements for a DLS Virtual Appliance or Containerized DLS Software Image](#) and reviewing the guidelines in [Sizing Guidelines for a DLS Appliance](#), install and configure the DLS virtual appliance by following this sequence of instructions:

1. [Installing the DLS Virtual Appliance Image on a Supported Hypervisor](#)
2. **If necessary:** [Setting the IP Address of a DLS Virtual Appliance from the Hypervisor](#)
3. [Registering the DLS Administrator User](#)
4. **Optional:** [Creating or Expanding an HA Cluster of DLS Instances](#)
5. **Optional:** [Setting the Static IP Address of a DLS Virtual Appliance](#)

2.5.1. Installing the DLS Virtual Appliance Image on a Supported Hypervisor

DLS virtual appliance images are available for several hypervisors. You use standard interfaces of the hypervisor to install the DLS virtual appliance on your chosen hypervisor.

The DLS virtual appliance image for each supported hypervisor **except** Red Hat Enterprise Linux KVM specifies the minimum configuration for the VM as listed in [Platform Requirements for a DLS Virtual Appliance or Containerized DLS Software Image](#). You are not required to specify the VM configuration when you install the DLS virtual appliance for these hypervisors. After installing the DLS virtual appliance, you can use standard interfaces of the hypervisor to change the configuration of the VM if necessary.



Note: If subnet conflicts occur, contact your network administrator for an appropriate subnet address and then follow these steps.

1. Modify the `back-tier` property of the `networks` element in the `/etc/dls/config/docker-compose.yml` file using the `rsu_admin` user.
2. Restart the VM for the change to take effect.

2.5.1.1. Installing the DLS Virtual Appliance on Citrix Hypervisor

The DLS image for Citrix Hypervisor is distributed as a ZIP archive that contains an XVA file, which is a format that is specific to Xen-based hypervisors.

Use the **Citrix XenCenter Import** wizard to perform this task on the Citrix Hypervisor host on which you want to run the DLS virtual appliance.

For additional information, see [Import VMs From XVA](#) on the Citrix product documentation website.

1. Download the ZIP archive that contains the XVA file that contains the DLS virtual appliance image to the hypervisor host.
2. Extract the contents of the ZIP archive that you downloaded.
3. In **Citrix XenCenter**, from the **File** menu, choose **Import**.
4. Browse for and select the downloaded XVA file, and click **Next**.
5. Select the server on which the imported VM will be placed and click **Next**.
6. Select the storage repository where the virtual disks for the newly imported VM will be stored and click **Import**.
7. Select the default virtual network interfaces in the template for the virtual appliance and click **Next**.
8. Review the settings for importing the virtual machine and click **Finish** to create the virtual machine.
9. Start the VM that you created.

Allow approximately 15 minutes after the VM is started for the installation of the DLS virtual appliance to complete and for the DLS virtual appliance to start. What to do after the DLS virtual appliance starts depends on whether the VM has been assigned an IP address automatically, for example, by a DHCP server:

- ▶ If the VM has been assigned an IP address, what to do next depends on whether you are performing a new installation or are upgrading an existing DLS instance:
 - ▶ If you are performing a new installation, register the DLS administrator user on the appliance as explained in [Registering the DLS Administrator User](#).
 - ▶ If you are upgrading an existing DLS instance, migrate the existing DLS instance as explained in [Migrating a DLS Instance](#).
- ▶ Otherwise, set the IP address of the DLS virtual appliance from the hypervisor as explained in [Setting the IP Address of a DLS Virtual Appliance from the Hypervisor](#).

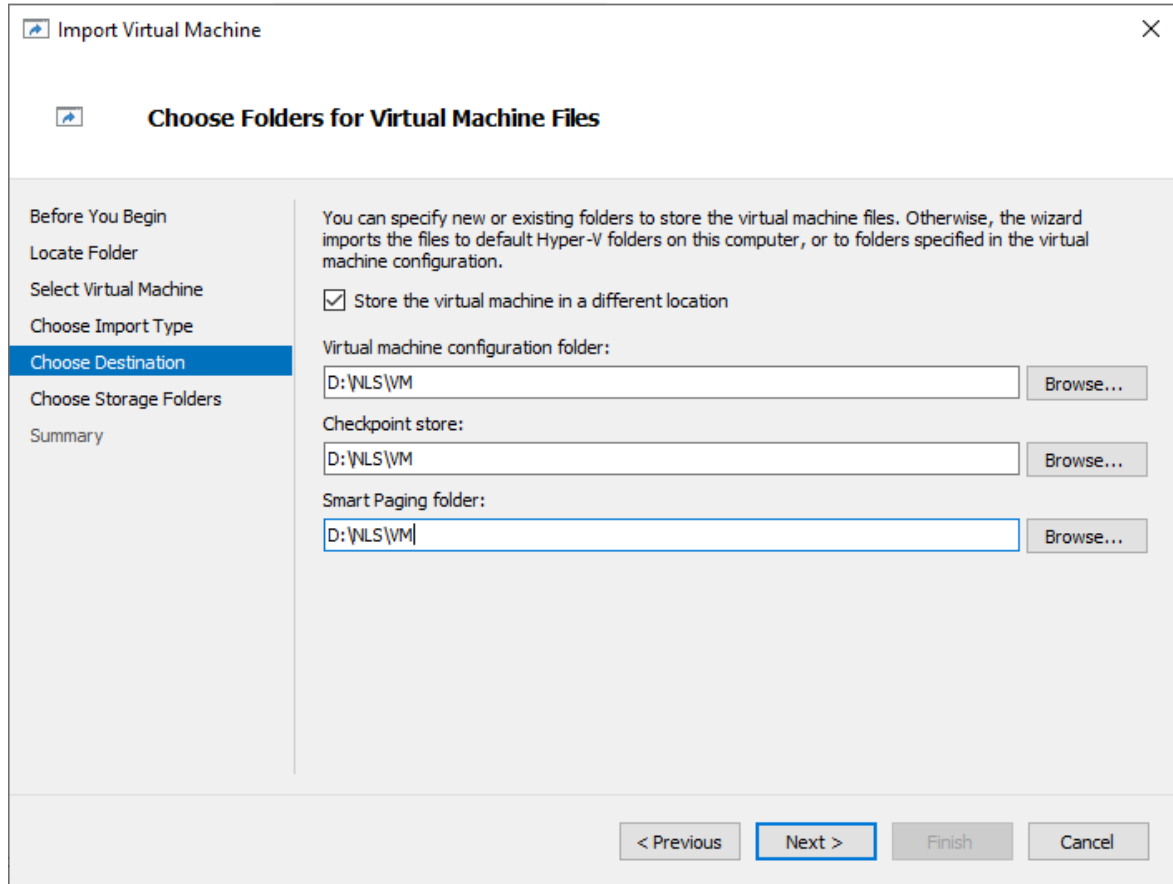
2.5.1.2. Installing the DLS Virtual Appliance on Microsoft Windows Server with Hyper-V

The DLS image for Microsoft Windows Server with Hyper-V is distributed as a ZIP archive.

Use **Hyper-V Manager** to perform this task on the Microsoft Windows Server with Hyper-V host on which you want to run the DLS virtual appliance.

For additional information, see [Import a Virtual Machine](#) on the Microsoft documentation website.

1. Download the ZIP archive that contains the DLS virtual appliance image to the hypervisor host.
2. Extract the contents of the ZIP archive that you downloaded.
3. From the **Hyper-V Manager Actions** menu, choose **Import Virtual Machine**.
4. Browse for and select the folder to which you extracted the DLS virtual appliance image and click **Next**.
5. When prompted to choose the type of import, select the **Copy the virtual machine (create a new unique ID)** option and click **Next**.
6. Browse for and select the folders in which you want to store virtual machine (VM) files and click **Next**.



7. Browse for and select the folder for storing the virtual disk.



Note: If you are creating two VMs for a cluster of DLS instances, ensure that you choose a unique folder for each VM to prevent errors from **Hyper-V Manager**.

8. Review the settings for importing the VM and click **Finish** to create the VM.
9. After the VM has been created, specify the virtual switch that the VM should use.
 - a). From the **Actions** menu under the name of the imported VM, choose **Settings**.
 - b). Under **Hardware** in the left navigation bar of the **Settings** window, select **Network Adapter**, select the virtual switch from the **Virtual switch** drop-down list, and click **Apply**.
10. Start the imported VM.
 - a). From the **Actions** menu under the name of the imported VM, choose **Connect**.
 - b). In the **Virtual Machine Connection** window that opens, click **Start**.

A command window opens when the installation of the imported VM is started. Use this window to log in to the DLS virtual appliance **only** if you need to set the IP address of the DLS virtual appliance from the hypervisor.

Allow approximately 15 minutes after the VM is started for the installation of the DLS virtual appliance to complete and for the DLS virtual appliance to start. What to do after

the DLS virtual appliance starts depends on whether the VM has been assigned an IP address automatically, for example, by a DHCP server:

- ▶ If the VM has been assigned an IP address, what to do next depends on whether you are performing a new installation or are upgrading an existing DLS instance:
 - ▶ If you are performing a new installation, register the DLS administrator user on the appliance as explained in [Registering the DLS Administrator User](#).
 - ▶ If you are upgrading an existing DLS instance, migrate the existing DLS instance as explained in [Migrating a DLS Instance](#).
- ▶ Otherwise, set the IP address of the DLS virtual appliance from the hypervisor as explained in [Setting the IP Address of a DLS Virtual Appliance from the Hypervisor](#).

After the VM has started, you can get its IP address from the **Networking** tab for the VM in **Hyper-V Manager**.

2.5.1.3. Installing the DLS Virtual Appliance on Red Hat Enterprise Linux KVM

The DLS image for Red Hat Enterprise Linux KVM is distributed as a ZIP archive that contains a QEMU copy-on-write (QCOW2) image file. After preparing the QCOW2 file, install the image by using **Virtual Machine Manager** to create a VM from the QCOW2 file. Perform this task from the hypervisor host.

1. Download the ZIP archive that contains the QCOW2 image file to the hypervisor host.
2. Extract the contents of the ZIP archive that you downloaded.
3. Copy the QCOW2 image file to the `/var/lib/libvirt/images` directory on the hypervisor host.
4. Start **Virtual Machine Manager**.
5. Add a connection to the hypervisor host.
 - a). In the **Virtual Machine Manager** window, from the **File** menu, choose **Add Connection**.
 - b). In the **Add Connection** window that opens, set the options in the following table and click **Connect**.

Option	Setting
Hypervisor	From the drop-down list, select QEMU/KVM .
Connect to remote host over SSH	Select this option.
User Name	In this text-entry field, type root .
Hostname	In this text-entry field, type the IP address or the fully qualified host name of the Red Hat Enterprise Linux KVM host.

The connection is added to the **Virtual Machine Manager** window.

6. Context click the connection that you added in the previous step and choose **New**. The **Create a new virtual machine** wizard starts.

7. In the first **New VM** window, select the **Import existing disk image** option and click **Forward**.
8. In the second **New VM** window, import the downloaded QCOW2 image file and choose the operating system to install.
 - a). Click **Browse**.
 - b). In the **Choose Storage Volume** window that opens, select the downloaded QCOW2 image file and click **Choose Volume**.
 - c). Back in the second **New VM** window, type **ubuntu 22.04** in the search box and from the list of operating systems that opens, select **Ubuntu 22.04 (ubuntu)**.
 - d). Click **Forward**.
9. In the third **New VM** window, set **Memory** to 8192 MiB and **CPUs** to 4, and click **Forward**.
10. In the final **New VM** window, specify the VM name and the network that the VM will use.
 - a). In the **Name** text-entry field, type your choice of name for the VM that you are creating.
 - b). Select the **Customize configuration before install** option.
 - c). From the Network Selection drop-down list, select the network that the VM will use.
 - d). Click **Finish**.
11. In the window for reviewing a new VM, modify the VM as follows.
 - a). Change the display server to **VNC Server**.
In the left navigation bar, select **Display Spice** and from the **Type** drop-down list, select **VNC Server**.
 - b). Remove **USB Redirector 1**, **USB Redirector 2**, and **Channel spice**.
For each item to remove, context-click the item in the left navigation bar and from the menu that pops up, choose **Remove Hardware**.
 - c). Change **Video QXL** to **VGA**.
In the left navigation bar, select **Video QXL** and from the **Model** drop-down list, select **VGA**.
12. Click **Apply** to save your changes to the configuration and click **Begin Installation**.
13. After the VM is created, start the VM and log in to the VM to get the IP address of the VM.
You must log in to the VM to get its IP address because the IP address is not shown in the **Virtual Machine Manager** console.
 - a). Click the play button to start the VM on the hypervisor host.
 - b). Use the command window that opens when the VM starts to log in to the DLS virtual appliance as the **dls_admin** user with the preset password **welcome**.
 - c). Get the address of the VM.

```
$ ip addr
```

Allow approximately 15 minutes after the VM is started for the installation of the DLS virtual appliance to complete and for the DLS virtual appliance to start. What to do after

the DLS virtual appliance starts depends on whether the VM has been assigned an IP address automatically, for example, by a DHCP server:

- ▶ If the VM has been assigned an IP address, what to do next depends on whether you are performing a new installation or are upgrading an existing DLS instance:
 - ▶ If you are performing a new installation, register the DLS administrator user on the appliance as explained in [Registering the DLS Administrator User](#).
 - ▶ If you are upgrading an existing DLS instance, migrate the existing DLS instance as explained in [Migrating a DLS Instance](#).
- ▶ Otherwise, set the IP address of the DLS virtual appliance from the hypervisor as explained in [Setting the IP Address of a DLS Virtual Appliance from the Hypervisor](#).

2.5.1.4. Installing the DLS Virtual Appliance on Red Hat Virtualization

The DLS image for Red Hat Virtualization is distributed as a ZIP archive that contains a QEMU copy-on-write (QCOW2) image file. After preparing the QCOW2 file, install the image by using **Virtual Machine Manager** to create a VM from the QCOW2 file. Perform this task from the hypervisor host.

1. Download the ZIP archive that contains the QCOW2 image file to the hypervisor host.
2. Extract the contents of the ZIP archive that you downloaded.
3. Copy the QCOW2 image file to the `/var/lib/libvirt/images` directory on the hypervisor host.
4. Start **Virtual Machine Manager**.
5. Add a connection to the hypervisor host.
 - a). In the **Virtual Machine Manager** window, from the **File** menu, choose **Add Connection**.
 - b). In the **Add Connection** window that opens, set the options in the following table and click **Connect**.

Option	Setting
Hypervisor	From the drop-down list, select QEMU/KVM .
Connect to remote host over SSH	Select this option.
User Name	In this text-entry field, type root .
Hostname	In this text-entry field, type the IP address or the fully qualified host name of the Red Hat Enterprise Linux KVM host.

The connection is added to the **Virtual Machine Manager** window.

6. Context click the connection that you added in the previous step and choose **New**. The **Create a new virtual machine** wizard starts.
7. In the first **New VM** window, select the **Import existing disk image** option and click **Forward**.

8. In the second **New VM** window, import the downloaded QCOW2 image file and choose the operating system to install.
 - a). Click **Browse**.
 - b). In the **Choose Storage Volume** window that opens, select the downloaded QCOW2 image file and click **Choose Volume**.
 - c). Back in the second **New VM** window, type **Ubuntu 22.04** in the search box and from the list of operating systems that opens, select **Ubuntu 22.04 (ubuntu)**.
 - d). Click **Forward**.
9. In the third **New VM** window, set **Memory** to 8192 MiB and **CPUs** to 4, and click **Forward**.
10. In the final **New VM** window, specify the VM name and the network that the VM will use.
 - a). In the **Name** text-entry field, type your choice of name for the VM that you are creating.
 - b). Select the **Customize configuration before install** option.
 - c). From the Network Selection drop-down list, select the network that the VM will use.
 - d). Click **Finish**.
11. In the window for reviewing a new VM, modify the VM as follows.
 - a). Change the display server to **VNC Server**.
In the left navigation bar, select **Display Spice** and from the **Type** drop-down list, select **VNC Server**.
 - b). Remove **USB Redirector 1**, **USB Redirector 2**, and **Channel spice**.
For each item to remove, context-click the item in the left navigation bar and from the menu that pops up, choose **Remove Hardware**.
 - c). Change **Video QXL** to **VGA**.
In the left navigation bar, select **Video QXL** and from the **Model** drop-down list, select **VGA**.
12. Click **Apply** to save your changes to the configuration and click **Begin Installation**.
13. After the VM is created, start the VM and log in to the VM to get the IP address of the VM.
You must log in to the VM to get its IP address because the IP address is not shown in the **Virtual Machine Manager** console.
 - a). Click the play button to start the VM on the hypervisor host.
 - b). Use the command window that opens when the VM starts to log in to the DLS virtual appliance as the **dls_admin** user with the preset password **welcome**.
 - c). Get the address of the VM.

```
$ ip addr
```

Allow approximately 15 minutes after the VM is started for the installation of the DLS virtual appliance to complete and for the DLS virtual appliance to start. What to do after the DLS virtual appliance starts depends on whether the VM has been assigned an IP address automatically, for example, by a DHCP server:

- ▶ If the VM has been assigned an IP address, what to do next depends on whether you are performing a new installation or are upgrading an existing DLS instance:
 - ▶ If you are performing a new installation, register the DLS administrator user on the appliance as explained in [Registering the DLS Administrator User](#).
 - ▶ If you are upgrading an existing DLS instance, migrate the existing DLS instance as explained in [Migrating a DLS Instance](#).
- ▶ Otherwise, set the IP address of the DLS virtual appliance from the hypervisor as explained in [Setting the IP Address of a DLS Virtual Appliance from the Hypervisor](#).

2.5.1.5. Installing the DLS Virtual Appliance on Ubuntu Hypervisor

The DLS image for Ubuntu Hypervisor is distributed as a ZIP archive that contains a QEMU copy-on-write (QCOW2) image file. After preparing the QCOW2 file, install the image by using **Virtual Machine Manager** to create a VM from the QCOW2 file.

Perform this task from the hypervisor host.

1. Download the ZIP archive that contains the QCOW2 image file to the hypervisor host.
2. Extract the contents of the ZIP archive that you downloaded.
3. Copy the QCOW2 image file to the `/var/lib/libvirt/images` directory on the hypervisor host.
4. Start **Virtual Machine Manager**.
5. Add a connection to the hypervisor host.
 - a). In the **Virtual Machine Manager** window, from the **File** menu, choose **Add Connection**.
 - b). In the **Add Connection** window that opens, set the options in the following table and click **Connect**.

Option	Setting
Hypervisor	From the drop-down list, select QEMU/KVM .
Connect to remote host over SSH	Select this option.
User Name	In this text-entry field, type root .
Hostname	In this text-entry field, type the IP address or the fully qualified host name of the Red Hat Enterprise Linux KVM host.

The connection is added to the **Virtual Machine Manager** window.

6. Context click the connection that you added in the previous step and choose **New**. The **Create a new virtual machine** wizard starts.
7. In the first **New VM** window, select the **Import existing disk image** option and click **Forward**.
8. In the second **New VM** window, import the downloaded QCOW2 image file and choose the operating system to install.
 - a). Click **Browse**.

- b). In the **Choose Storage Volume** window that opens, select the downloaded QCOW2 image file and click **Choose Volume**.
 - c). Back in the second **New VM** window, type **ubuntu 22.04** in the search box and from the list of operating systems that opens, select **Ubuntu 22.04 (ubuntu)**.
 - d). Click **Forward**.
9. In the third **New VM** window, set **Memory** to 8192 MiB and **CPUs** to 4, and click **Forward**.
 10. In the final **New VM** window, specify the VM name and the network that the VM will use.
 - a). In the **Name** text-entry field, type your choice of name for the VM that you are creating.
 - b). Select the **Customize configuration before install** option.
 - c). From the Network Selection drop-down list, select the network that the VM will use.
 - d). Click **Finish**.
 11. In the window for reviewing a new VM, modify the VM as follows.
 - a). Change the display server to **VNC Server**.
In the left navigation bar, select **Display Spice** and from the **Type** drop-down list, select **VNC Server**.
 - b). Remove **USB Redirector 1**, **USB Redirector 2**, and **Channel spice**.
For each item to remove, context-click the item in the left navigation bar and from the menu that pops up, choose **Remove Hardware**.
 - c). Change **Video QXL** to **VGA**.
In the left navigation bar, select **Video QXL** and from the **Model** drop-down list, select **VGA**.
 12. Click **Apply** to save your changes to the configuration and click **Begin Installation**.
 13. After the VM is created, start the VM and log in to the VM to get the IP address of the VM.
You must log in to the VM to get its IP address because the IP address is not shown in the **Virtual Machine Manager** console.
 - a). Click the play button to start the VM on the hypervisor host.
 - b). Use the command window that opens when the VM starts to log in to the DLS virtual appliance as the **dls_admin** user with the preset password **welcome**.
 - c). Get the address of the VM.

```
$ ip addr
```

Allow approximately 15 minutes after the VM is started for the installation of the DLS virtual appliance to complete and for the DLS virtual appliance to start. What to do after the DLS virtual appliance starts depends on whether the VM has been assigned an IP address automatically, for example, by a DHCP server:

- If the VM has been assigned an IP address, what to do next depends on whether you are performing a new installation or are upgrading an existing DLS instance:

- ▶ If you are performing a new installation, register the DLS administrator user on the appliance as explained in [Registering the DLS Administrator User](#).
- ▶ If you are upgrading an existing DLS instance, migrate the existing DLS instance as explained in [Migrating a DLS Instance](#).
- ▶ Otherwise, set the IP address of the DLS virtual appliance from the hypervisor as explained in [Setting the IP Address of a DLS Virtual Appliance from the Hypervisor](#).

2.5.1.6. Installing the DLS Virtual Appliance on VMware vSphere

The DLS image for VMware vSphere is distributed as a ZIP archive that contains an Open Virtual Appliance (OVA) file.

Use the **VMware vSphere Client** to perform this task on the ESXi server on which you want to run the DLS virtual appliance.

For additional information, see the following topics on the VMware Docs site:

- ▶ [Log in to vCenter Server by Using the vSphere Web Client](#)
 - ▶ [Deploy an OVF or OVA Template](#)
1. Download the ZIP archive that contains the OVA file that contains the DLS image for VMware vSphere.
 2. Extract the contents of the ZIP archive that you downloaded.
 3. Log in to vCenter Server by using the **VMware vSphere Client**.
 4. From the **VMware vSphere Client Actions** menu, choose **Deploy OVF Template**.
 5. Select the **Local file** option, browse for and select the downloaded OVA file, and click **Next**.
 6. Enter your choice of virtual machine name, select a location for the virtual machine, and click **Next**.
 7. Select a compute resource where the virtual machine will be created and click **Next**.
 8. Review the details of the template that you are deploying and click **Next**.
 9. Select the storage for the virtual appliance configuration and disk files and click **Next**.
 10. Leave the destination network as-is, set the **IP allocation** option to **Static - Manual**, and click **Next**.
 11. Set the virtual network and IP allocation properties for the VM and click **Next**.
 - a). In the **ipaddress** text-entry field, type the IP address that you want to assign to the VM.



Note: To change this address after completing the installation, you **must** follow the instructions in [Changing the IP Address of a VMware VM Set During DLS Appliance Installation](#). If you use any other method to change this address, it reverts to its original setting when the VM is restarted.

- b). In the **netmask** text-entry field, type the subnet mask of the VM's network in classless inter-domain routing (CIDR) format **without the leading slash character (/)**.

To get a subnet mask in CIDR format from its decimal equivalent, refer to the table on page 2 of [IETF RFC 1878: Variable Length Subnet Table For IPv4](#).

For example, the subnet mask in CIDR format of the decimal equivalent 255.255.255.0 is 24.

- c). In the **gateway** text-entry field, type the IP address of the VM's default gateway.



Note: If you leave the **gateway** field empty, the DLS virtual appliance uses DHCP settings.

- d). **Optional:** In the **dns_server_one** text-entry field, type the IP address of the first DNS server to be used for name resolution.
- e). **Optional:** In the **dns_server_two** text-entry field, type the IP address of the second DNS server to be used for name resolution.

12. Review all the details of the virtual machine that you are creating and click **Finish**.

13. Start the VM that you created.



Note: If subnet conflicts occur, contact your network administrator for an appropriate subnet address and then follow these steps.

- a). Modify the `back-tier` property of the `networks` element in the `/etc/dls/config/docker-compose.yml` file using the `rsu_admin` user.
- b). Restart the VM for the change to take effect.

Allow approximately 15 minutes after the VM is started for the installation of the DLS virtual appliance to complete and for the DLS virtual appliance to start. What to do after the DLS virtual appliance starts depends on whether the VM has been assigned an IP address automatically, for example, by a DHCP server:

- ▶ If the VM has been assigned an IP address, what to do next depends on whether you are performing a new installation or are upgrading an existing DLS instance:
 - ▶ If you are performing a new installation, register the DLS administrator user on the appliance as explained in [Registering the DLS Administrator User](#).
 - ▶ If you are upgrading an existing DLS instance, migrate the existing DLS instance as explained in [Migrating a DLS Instance](#).
- ▶ Otherwise, set the IP address of the DLS virtual appliance from the hypervisor as explained in [Setting the IP Address of a DLS Virtual Appliance from the Hypervisor](#).

2.5.2. Setting the IP Address of a DLS Virtual Appliance from the Hypervisor

If the VM that hosts a DLS virtual appliance has **not** been assigned an IP address automatically, you must set the IP address from the hypervisor. Each DLS virtual

appliance provides a shell script specifically for this purpose and is configured with a user account for running the script.



Note: You can perform this task **only** on a VM-based DLS virtual appliance. You **cannot** perform this task on a containerized DLS software image. Instead, you can use standard interfaces of the OS on which the container orchestration platform is running to make this change.



Note: The IP address of a VMware VM that hosts a DLS appliance can be set when the appliance is installed from an OVF template. If the IP address was set this way, you **must** follow the instructions in [Changing the IP Address of a VMware VM Set During DLS Appliance Installation](#). If you use any other method to change this address, it reverts to its original setting when the VM is restarted.

1. Use the hypervisor management console of the appliance to log in as the user **dls_admin** to the VM that hosts the DLS virtual appliance.
If the **dls_admin** user has not been registered, you can still log in to the VM as the **dls_admin** user with the default password **welcome**.
2. Run the `/etc/adminscripts/set-static-ip-cli.sh` script.

```
$ /etc/adminscripts/set-static-ip-cli.sh
```
3. When prompted, enter the details of the IP address.
The script presents any default values that are already set for the virtual appliance's network.
 - a). Enter the number that denotes the IP version that the virtual appliance's network uses.
 - ▶ For an IPv4 network, type **4**.
 - ▶ For an IPv6 network, type **6**.
 - b). Enter the IP address that you want to assign to the DLS virtual appliance.
 - c). Enter the IP address of the DLS virtual appliance's default gateway.



Note: If you omit the default gateway address, the DLS virtual appliance uses DHCP settings.

- d). Enter the IP address of the first DNS server to be used for name resolution.
- e). Enter the IP address of the second DNS server to be used for name resolution.
- f). Enter the subnet mask of the DLS virtual appliance's network in classless inter-domain routing (CIDR) format **without the leading slash character (/)**.
To get a subnet mask in CIDR format from its decimal equivalent, refer to the table on page 2 of [IETF RFC 1878: Variable Length Subnet Table For IPv4](#).
For example, the subnet mask in CIDR format of the decimal equivalent 255.255.255.0 is 24.

After the IP address has been set, log files containing progress message from the script are available in the `/tmp/static-ip-cli-logs` directory.

What to do next depends on whether you are performing a new installation or are upgrading an existing DLS instance:

- ▶ If you are performing a new installation, register the DLS administrator user on the appliance as explained in [Registering the DLS Administrator User](#).
- ▶ If you are upgrading an existing DLS instance, migrate the existing DLS instance as explained in [Migrating a DLS Instance](#).

2.5.3. Changing the IP Address of a VMware VM Set During DLS Appliance Installation

The IP address of a VMware VM that hosts a DLS appliance can be set when the appliance is installed from an OVF template. If the IP address was set this way, you can change it **only** by editing vApps options within VM that hosts the DLS appliance. If you use any other method to change this address, it reverts to its original setting when the VM is restarted.

1. Log in to vCenter Server by using the **VMware vSphere Web Client**.
For detailed instructions, refer to [Log in to vCenter Server by Using the vSphere Web Client](#).
2. From the **vCenter Server** inventory, navigate to the VM that hosts the DLS appliance.
3. Shut down the VM to which you navigated in the previous step.
4. On the **Configure** tab, expand **Settings**, select **vApp options**, and click **Edit**.
5. In the list of vApp options, select **Networkproperties.IPaddress** and click **Set Value**.
6. Enter the IP address that you want to assign to the VM that hosts the DLS appliance.
7. Save your changes and start the VM.

2.6. Deploying a Containerized DLS Software Image

For deployments on a supported container orchestration platform, the DLS is supplied as a containerized software image.

After verifying the requirements in [Platform Requirements for a DLS Virtual Appliance or Containerized DLS Software Image](#), deploy a containerized DLS software image by following this sequence of instructions:

1. [Setting Properties for a Containerized DLS Software Image](#)
2. [Deploying the Containerized DLS Software Image on a Supported Platform](#)
3. [Registering the DLS Administrator User](#)
4. **Optional:** [Creating or Expanding an HA Cluster of DLS Instances](#)

2.6.1. Contents of the Containerized DLS Software Image Download

The containerized DLS software image is distributed as the ZIP archive `nls-3.3.1-bios.zip`.

This ZIP archive contains the following artifacts:

`dls_appliance_3.3.1.tar.gz`

The DLS application container.

`dls_pgsql_3.3.1.tar.gz`

The PostgreSQL database container.



Note: The version of both containers must be the same.

The DLS application container and the PostgreSQL database container share a common volume for exchanging information between them. If both containers are deployed on a multinode cluster, they must be run on the same worker node.

2.6.2. Setting Properties for a Containerized DLS Software Image

To enable communications with the container within which the DLS appliance is running, you must set properties such as host names, IP addresses, and port numbers. Set these properties before deploying the containerized DLS software image on a supported platform.

Set the properties for a containerized DLS software image by editing the appropriate deployment file for your container orchestration platform.

Container Orchestration Platform	File
Docker	<code>docker-compose.yml</code>
Kubernetes	<code>nls-si-0-deployment.yml</code>
Podman	<code>nls-si-0-deployment/compose.yml</code>
Red Hat OpenShift Container Platform	<code>nls-si-0-deployment.yml</code>
VMware Tanzu Application Platform	<code>nls-si-0-deployment.yml</code>

2.6.2.1. Setting Environment Variables to Specify String Properties

You can set environment variables to specify string properties for all supported container orchestration platforms.

Set environment variables to specify string properties by editing the appropriate section of the deployment file for your container platform.

Deployment File	Section to Edit
docker-compose.yml	environment:
nls-si-0-deployment/compose.yml	environment:
nls-si-0-deployment.yaml	- env:

The environment variables for specifying string properties are as follows:

Required: DLS_DB_HOST

The IP address of the database container.

Ensure that the IP address of the database container is a private IP address on the same subnet as the DLS appliance container. The database container must **not** be accessible over any public subnet.

Required: DLS_PUBLIC_IP

The IP address of the container orchestration platform. To access the **NVIDIA Licensing** application, the user connects to the container in which the DLS software image is deployed through this address.

Optional: DLS_PRIVATE_HOSTNAME

The host name of the container in which the DLS software image is deployed.

If this environment variable is not set, the container orchestration platform sets the host name to the container ID.

Optional: FQDN

The fully qualified domain name that is mapped to the IP address that is specified by the `DLS_PUBLIC_IP` environment variable.

Ensure that this name can be resolved by licensed clients. Do **not** specify a name that can be resolved only by the container orchestrator.

If this environment variable is not set, a user **cannot** connect to the container in which the DLS software image is deployed through a fully qualified domain name.

2.6.2.2. Specifying Integer Values as Environment Variables

You can specify integer values such as port numbers for Docker, Podman, Red Hat OpenShift Container Platform, and VMware Tanzu Application Platform as environment variables.



Note: Kubernetes does not accept strings for integer values. Therefore, you cannot specify integer values for Kubernetes as environment variables. Instead, you must specify integer values for Kubernetes as explained in [Specifying Integer Values as Actual Values](#).

To specify integer values by setting environment variables, edit the appropriate section of the deployment file for your container platform.

Deployment File	Section to Edit
docker-compose.yml	environment:
nls-si-0-deployment/compose.yml	environment:
nls-si-0-deployment.yaml	- env:

The following environment variables apply to Docker, Red Hat OpenShift Container Platform, and VMware Tanzu Application Platform:

Optional: CPU_CORE_LIMIT

The maximum number of CPU cores that can be assigned to the DLS appliance container pod.

Default: 4

Optional: DLS_HA_MAX_NODES_ENV

The maximum number of instances in a cluster to which this instance is to be added.

Allowed values: 2-9

Default: 2

Optional: DLS_HTTP_PORT

The port on which the container in which the DLS software image is deployed listens for HTTP requests.

Default: 80

Optional: DLS_HTTPS_PORT

The port on which the container in which the DLS software image is deployed listens for HTTPS requests.

Default: 443

Optional: DLS_RABBITMQ_SSL_PORT

The Secure Sockets Layer (SSL) port through which the nodes in and HA cluster of DLS instances communicate with each other.

Default: 5671

The following environment variables apply **only** to Red Hat OpenShift Container Platform and VMware Tanzu Application Platform:

Optional: DLS_EXPOSED_HTTP_PORT

The port on which the container orchestrator listens for and forwards the request to the port that is specified by the `DLS_HTTP_PORT` environment variable.

Not applicable to Docker or Podman.

Default: 30000

Optional: DLS_EXPOSED_HTTPS_PORT

The port on which the container orchestrator listens for and forwards the request to the port that is specified by the `DLS_HTTP_PORTS` environment variable.

Not applicable to Docker or Podman.

Default: 30001

Optional: DLS_VA_SERVICE_PORT

The port in the container orchestrator to which the DLS appliance container binds.

Not applicable to Docker or Podman.

Default: 8085

2.6.2.3. Specifying Integer Values as Actual Values

Kubernetes does not accept strings for integer values. Therefore, you cannot specify integer values for Kubernetes as environment variables. Instead, you must specify integer values for Kubernetes as actual values.

To specify integer values as actual values, replace the references to environment variables with actual values in the `ports:` section of the `nls-si-0-deployment.yaml` and `nls-si-0-service.yaml` deployment files.

- ▶ In `nls-si-0-deployment.yaml`, the references to environment variables in the `ports:` section are as follows:

```
ports:
  - containerPort: ${DLS_HTTP_PORT:-80}
    hostPort: ${DLS_EXPOSED_HTTP_PORT:-80}
  - containerPort: ${DLS_HTTPS_PORT:-443}
    hostPort: ${DLS_EXPOSED_HTTPS_PORT:-443}
  - containerPort: ${DLS_RABBITMQ_SSL_PORT:-5671}
    hostPort: ${DLS_RABBITMQ_SSL_PORT:-5671}
  - containerPort: 8085
    hostPort: ${DLS_VA_SERVICE_PORT:-8085}
```

- ▶ In `nls-si-0-service.yaml`, the references to environment variables in the `ports:` section are as follows:

```
ports:
  - name: "DLS_HTTP_CONTAINER"
    port: ${DLS_HTTP_PORT:-80}
    targetPort: ${DLS_HTTP_PORT:-80}
    nodePort: ${DLS_EXPOSED_HTTP_PORT:-30000}
  - name: "DLS_HTTPS_CONTAINER"
    port: ${DLS_HTTPS_PORT:-443}
    targetPort: ${DLS_HTTPS_PORT:-443}
    nodePort: ${DLS_EXPOSED_HTTPS_PORT:-30001}
  - name: "DLS_RABBITMQ_SSL_PORT"
    port: ${DLS_RABBITMQ_SSL_PORT:-5671}
    targetPort: ${DLS_RABBITMQ_SSL_PORT:-5671}
    nodePort: ${DLS_RABBITMQ_SSL_PORT:-5671}
  - name: "DLS_VA_SERVICE_PORT"
    port: 8085
    targetPort: 8085
    nodePort: ${DLS_VA_SERVICE_PORT:-8085}
```

The values with which to replace these references to environment variables are as follows:

Both files: \${DLS_HTTP_PORT:-80}

The port on which the container in which the DLS software image is deployed listens for HTTP requests.

nls-si-0-deployment.yaml: `{DLS_EXPOSED_HTTP_PORT:-80}`

nls-si-0-service.yaml: `{DLS_EXPOSED_HTTP_PORT:-30000}`

The port on which the container orchestrator listens for and forwards the request to the port that is specified by the `DLS_HTTP_PORT` environment variable.

Both files: `{DLS_HTTPS_PORT:-443}`

The port on which the container in which the DLS software image is deployed listens for HTTPS requests.

nls-si-0-deployment.yaml: `{DLS_EXPOSED_HTTPS_PORT:-443}`

nls-si-0-service.yaml: `{DLS_EXPOSED_HTTPS_PORT:-30001}`

The port on which the container orchestrator listens for and forwards the request to the port that is specified by the `DLS_HTTP_PORTS` environment variable.

Both files: `{DLS_RABBITMQ_SSL_PORT:-5671}`

The Secure Sockets Layer (SSL) port through which the nodes in and HA cluster of DLS instances communicate with each other.

Both files: `{DLS_VA_SERVICE_PORT:-8085}`

The port in the container orchestrator to which the DLS appliance container binds.

This example shows the `ports:` sections in the `nls-si-0-deployment.yaml` and `nls-si-0-service.yaml` deployment files in which the references to environment variables are replaced with actual values.

- ▶ In `nls-si-0-deployment.yaml`, the `ports:` section is as follows:

```
ports:
  - containerPort: 80
    hostPort: 80
  - containerPort: 443
    hostPort: 443
  - containerPort: 5671
    hostPort: 5671
  - containerPort: 8085
    hostPort: 8085
```

- ▶ In `nls-si-0-service.yaml`, the `ports:` section is as follows:

```
ports:
  - name: "DLS_HTTP_CONTAINER"
    port: 80
    targetPort: 80
    nodePort: 30000
  - name: "DLS_HTTPS_CONTAINER"
    port: 443
    targetPort: 443
    nodePort: 30001
  - name: "DLS_RABBITMQ_SSL_PORT"
    port: 5671
    targetPort: 5671
    nodePort: 30002
  - name: "DLS_VA_SERVICE_PORT"
    port: 8085
    targetPort: 8085
    nodePort: 30003
```

2.6.3. Deploying the Containerized DLS Software Image on a Supported Platform

The containerized DLS software image is distributed with default deployment files for several container orchestration platforms. You use standard interfaces of the container orchestration platform to deploy the containerized DLS software image on your chosen platform.

The default deployment file for each supported container orchestration platform specifies the resource reservations for the container that are listed in [Platform Requirements for a DLS Virtual Appliance or Containerized DLS Software Image](#). You are not required to specify the resource reservations for the container when you deploy the containerized DLS software image.

Follow the instructions for the container orchestration platform that you are using.

Container Orchestration Platform	Instructions
Docker	Deploying the Containerized DLS Software Image on Docker
Kubernetes	Deploying the Containerized DLS Software Image on Kubernetes Platforms
Podman	Deploying the Containerized DLS Software Image on Podman
Red Hat OpenShift Container Platform	Deploying the Containerized DLS Software Image on Kubernetes Platforms
VMware Tanzu Application Platform	Deploying the Containerized DLS Software Image on Kubernetes Platforms

2.6.3.1. Deploying the Containerized DLS Software Image on Docker

Ensure that the required environment variables have been set as explained in [Setting Properties for a Containerized DLS Software Image](#).

1. Import the DLS appliance artifact and the PostgreSQL database artifact.

- a). Import the DLS appliance artifact.

```
docker load --input dls_appliance_3.3.1.tar.gz
```

- b). Import the PostgreSQL database artifact.

```
$ docker load --input dls_pgsql_3.3.1.tar.gz
```

2. Change to the directory that contains the `docker-compose.yml` file.
3. Start the DLS appliance and PostgreSQL database containers.

```
$ docker-compose up
```

2.6.3.2. Deploying the Containerized DLS Software Image on Kubernetes Platforms

Perform this task if you are using Kubernetes, Red Hat OpenShift Container Platform, or VMware Tanzu Application Platform.

Ensure that the following prerequisites are met:

- ▶ The required environment variables have been set as explained in [Setting Properties for a Containerized DLS Software Image](#).
- ▶ A private repository for the DLS appliance artifact and the PostgreSQL database artifact has been created.
- ▶ If a registry secret is required for pulling the artifacts from the private repository, the registry secret has been created.

In the deployment files for the DLS appliance artifact and the PostgreSQL database artifact, the name of the registry secret is preset to `registry-secret`. If you create a secret with this name, you don't need to edit the deployment files to reference the secret.



Note: The DLS application container and the PostgreSQL database container share a common volume for exchanging information between them. If both containers are deployed on a multinode cluster, they must be run on the same worker node.

1. Import the DLS appliance artifact and the PostgreSQL database artifact into your private repository.
2. Edit the deployment files for the DLS appliance artifact and the PostgreSQL database artifact to pull these artifacts from the private repository into which they were imported.

In each file, replace the string `<POPULATE THIS WITH PRIVATE REPOSITORY>` with the name of the private repository.

- a). For the DLS appliance artifact, edit the following line in the file `nls-si-0-deployment.yaml`:

```
image: <POPULATE THIS WITH PRIVATE REPOSITORY>:appliance
```

- b). For the PostgreSQL database artifact, edit the following line in the file `postgres-nls-si-0-deployment.yaml`:

```
image: <POPULATE THIS WITH PRIVATE REPOSITORY>:pgsql
```

3. If a registry secret is required for pulling the artifacts from the private repository, edit the deployment files for the DLS appliance artifact and the PostgreSQL database artifact to reference the secret.

In each file, replace the string `registry-secret` with the name of your registry secret.

- a). For the DLS appliance artifact, edit the file `nls-si-0-deployment.yaml`.
- b). For the PostgreSQL database artifact, edit the file `postgres-nls-si-0-deployment.yaml`.

4. Start the PostgreSQL database container pod with the supplied deployment file `postgres-nls-si-0-deployment.yaml`.

```
$ kubectl create -f directory/postgres-nls-si-0-deployment.yaml
```

directory

The full path to the directory that contains the file `postgres-nls-si-0-deployment.yaml`.

5. Get the IP address of the PostgreSQL database container pod.

```
$ kubectl get pods -o wide
```

6. Set the `DLS_DB_HOST` environment variable in the file `nls-si-0-deployment.yaml` to the IP address that you got in the previous step.



Note: To enable the DLS appliance container pod to create and modify files on the volumes mapped to it, ensure that the required file access permissions are set on these volumes.

7. Start the DLS appliance container pod with the supplied deployment file `nls-si-0-deployment.yaml`.

```
$ kubectl create -f directory/nls-si-0-deployment.yaml
```

directory

The full path to the directory that contains the file `nls-si-0-deployment.yaml`.

2.6.3.3. Deploying the Containerized DLS Software Image on Podman

Ensure that the required environment variables have been set as explained in [Setting Properties for a Containerized DLS Software Image](#).

1. Import the DLS appliance artifact and the PostgreSQL database artifact.

- a). Import the DLS appliance artifact.

```
$ podman load --input dls_appliance_3.3.1.tar.gz
```

- b). Import the PostgreSQL database artifact.

```
$ podman load --input dls_pgsql_3.3.1.tar.gz
```

2. Change to the `podman` directory.
3. **Optional:** Edit the `postgres-nls-si-0-deployment/compose.yaml` file to customize the subnets with which you want to create the `postgres-nls-si-0-deployment_back-tier` network.

By default, the `postgres-nls-si-0-deployment_back-tier` network is created with the following subnets:

- ▶ **IPv4:** `172.16.238.0/28`
- ▶ **IPv6:** `2001:3984:3989::/64`

4. Create the `postgres-nls-si-0-deployment_back-tier` network, configurations volume mapping, and `postgres-data` volume mapping, and deploy the PostgreSQL database container.

```
$ podman-compose -f postgres-nls-si-0-deployment/compose.yaml up -d
```

For information about the configurations and `postgres-data` volume mappings, refer to [Volume Mappings for a Containerized DLS Software Image](#).

5. Confirm that the `postgres-nls-si-0-deployment_back-tier` network, configurations volume mapping, and `postgres-data` volume mapping were created and that the PostgreSQL database container was deployed.

- a). Confirm that the `postgres-nls-si-0-deployment_back-tier` network has been created.

```
# podman network ls
NETWORK ID      NAME                DRIVER
2f259bab93aa    podman              bridge
272c45824a80    postgres-nls-si-0-deployment_back-tier  bridge
```

- b). Confirm that the properties of the `postgres-nls-si-0-deployment_back-tier` network, for example, its subnets, are correct.

```
# podman network inspect postgres-nls-si-0-deployment_back-tier
[
  {
    "name": "postgres-nls-si-e-deployment_back-tier",
    "id": "272cu5824a80fd8117261f28e820f920999fudes6a03ua781e9ecf1015a76201",
    "driver": "bridge",
    "network_interface": "cni-podman1",
    "created": "2023-03-08T07:42:03.88777482",
    "subnets": [
      {
        "subnet": "172.16.238.0/28",
        "gateway": "172.16.238.1"
      },
      {
        "subnet": "2001:3984:3989::/64",
        "gateway": "2001:3984:3989::1"
      }
    ],
    "ipv6_enabled": true,
    "internal": false,
    "dns_enabled": true,
    "labels": {
      "com.docker.compose.project": "postgres-nls-si-0-deployment",
      "io.podman.compose.project": "postgres-nls-si-0-deployment"
    },
    "ipam_options": {
      "driver": "host-local"
    }
  }
]
```

- c). Confirm that the `configurations` volume mapping and `postgres-data` volume mapping were created.

```
# podman volume ls
DRIVER          VOLUME NAME
local           configurations
local           postgres-data
```

- d). Confirm that the PostgreSQL database container was deployed.

```
# podman container ls
```

6. Get the IP address of the PostgreSQL database container pod.

```
$ podman inspect postgres-nls-si-0-deployment_postgres-nls-si-0_1
```

7. Set the `DLS_DB_HOST` environment variable in the file `nls-si-0-deployment/compose.yml` to the IP address that you got in the previous step.

8. Create the `logs` volume mapping and `rabbitmq_data` volume mapping, and deploy the DLS appliance container.

```
$ podman-compose -f nls-si-0-deployment/compose.yml up -d
```

For information about the `logs` and `rabbitmq_data` volume mappings, refer to [Volume Mappings for a Containerized DLS Software Image](#).

9. Confirm that the `logs` volume mapping and `rabbitmq_data` volume mapping were created and that the DLS appliance container was deployed.
 - a). Confirm that the `logs` volume mapping and `rabbitmq_data` volume mapping were created.

```
# podman volume ls
DRIVER          VOLUME NAME
...
local          logs
...
local          rabbitmq_data
```

- b). Confirm that the DLS appliance container was deployed.

```
# podman container ls
```

2.6.4. Volume Mappings for a Containerized DLS Software Image

When a containerized DLS software image is deployed, several volume mappings are created for maintaining the state of the **NVIDIA Licensing** application in the DLS appliance.

The mapping of the volumes to the container filesystem can be found in the respective **deployment files** depending on the Orchestrator platform that the containerized appliance is being deployed on.

Platform	File
Docker	docker-compose.yml
Kubernetes	nls-si-0-deployment.yaml
Podman	compose.yml
Openshift	nls-si-0-deployment.yaml
Tanzu	nls-si-0-deployment.yaml

The volume mappings that are created are as follows:

configurations

Contains the state of the DLS appliance container to enable its dynamic properties to be retrieved if the container fails.

logs

Contains log files created by the **NVIDIA Licensing** application in the DLS appliance.

postgres-data

Contains **NVIDIA Licensing** application data created by the PostgreSQL database. Because this volume contains information about available and checked out licenses, ensure that access to this volume is limited.

rabbitmq_data

Contains information about the real-time replication of data between nodes in an HA cluster of DLS instances.



Note: These volume mappings maintain the state of the **NVIDIA Licensing** application in the DLS appliance. Therefore, do **not** modify these volume mappings.

2.6.5. Troubleshooting Issues with Deploying the Containerized DLS Software Image

2.6.5.1. Containerized DLS Appliance Fails to Start on Kubernetes Platforms

When this issue occurs, the service enters a crash loop, container creating, pending sequence before the container enters an error state.

This issue might affect Kubernetes, Red Hat OpenShift Container Platform, or VMware Tanzu Application Platform. It does **not** affect Docker.

1. Ensure that the file access permissions on the volumes allow write access by the database container and the DLS appliance container, for example `707 (rwx---rwx)`.
2. If the containerized DLS software image is deployed on a multinode cluster, ensure that the database container and the DLS appliance container are running on the same worker node.
3. If the Kubernetes cluster does not dynamically provision persistent volumes when creating persistent volume claims, ensure that the persistent volumes are created manually with file access permissions that allow the database container and the DLS appliance container to write to them.
4. Ensure that the versions of the database container and the DLS appliance container are identical.

2.6.5.2. Volume Mode Access for DLS Appliance Resources on Kubernetes Platforms Is Incorrect

The DLS application container and the PostgreSQL database container share a common volume for exchanging information between them. If the volume mode access is incorrect, start-up of only the first container pod to be started is successful. The remaining container pod fails to start because the volume is already mounted on the first container pod to be started.

This issue might affect Kubernetes, Red Hat OpenShift Container Platform, or VMware Tanzu Application Platform. It does **not** affect Docker.

1. If both containers are deployed on a multinode cluster, ensure that the DLS application container and the PostgreSQL database container run on the same worker node.

2. Ensure that the PostgreSQL database container pod is started **before** the DLS appliance container pod.

For more information, refer to [Deploying the Containerized DLS Software Image on Kubernetes Platforms](#).

2.6.5.3. Data Validation Errors Prevent the DLS Appliance Container Pod from Starting on Kubernetes

This issue occurs if port numbers are specified as references to environment variables in the deployment files. Because Kubernetes does not accept string values for port numbers, you cannot specify port numbers for Kubernetes by setting environment variables.

This issue might affect Kubernetes. It does **not** affect other supported container orchestration platforms.

When this issue occurs, the following error messages are displayed:

```
error: error validating ".": error validating data:[
Validation error (Deployment.spec.template.spec.containers[0].ports[0].containerPort):
invalid type for io.k8s.api.core.v1.ContainerPort.containerPort: got "string",
expected "integer",
Validation error (Deployment.spec.template.spec.containers[0].ports[0].hostPort):
invalid type for io.k8s.api.core.v1.ContainerPort.hostPort: got "string", expected
"integer",
Validation error (Deployment.spec.template.spec.containers[0].ports[1].containerPort):
invalid type for io.k8s.api.core.v1.ContainerPort.containerPort: got "string",
expected "integer",
Validation error (Deployment.spec.template.spec.containers[0].ports[1].hostPort):
invalid type for io.k8s.api.core.v1.ContainerPort.hostPort: got "string", expected
"integer",
Validation error (Deployment.spec.template.spec.containers[0].ports[2].containerPort):
invalid type for io.k8s.api.core.v1.ContainerPort.containerPort: got "string",
expected "integer",
Validation error (Deployment.spec.template.spec.containers[0].ports[2].hostPort):
invalid type for io.k8s.api.core.v1.ContainerPort.hostPort: got "string", expected
"integer",
Validation error (Deployment.spec.template.spec.containers[0].ports[3].hostPort):
invalid type for io.k8s.api.core.v1.ContainerPort.hostPort: got "string", expected
"integer"];
```

1. Edit the deployment files to specify port numbers as integer values instead of references to environment variables.
For instructions, refer to [Specifying Integer Values as Actual Values](#).
2. Start the DLS appliance container pod with the supplied deployment file `nls-si-0-deployment.yaml`.

```
$ kubectl create -f directory/nls-si-0-deployment.yaml
directory
```

The full path to the directory that contains the file `nls-si-0-deployment.yaml`.

2.6.5.4. The Management Interface of the NVIDIA Licensing Application Fails to Load

When this issue occurs, the containerized DLS appliance starts correctly but the management interface of the **NVIDIA Licensing** application is not loaded into the browser.

1. Ensure that the `DLS_DB_HOST` environment variable is set to the IP address of the database container.
2. Ensure that the environment variables for port mappings set in the deployment files are correct and that the ports are open on the Kubernetes node.
3. Ensure that the port specified by the `DLS_EXPOSED_HTTPS_PORT` environment variable is open in the firewall on the node that is hosting the DLS appliance container.
4. If the Kubernetes cluster does not dynamically provision persistent volumes when creating persistent volume claims, ensure that the persistent volumes are created manually with file access permissions that allow the database container and the DLS appliance container to write to them.
5. Ensure that the versions of the database container and the DLS appliance container are identical.

2.6.5.5. HA Configuration or Online Upgrade Fails

High availability (HA) configuration or online upgrade between two containerized DLS appliances fails if port mappings or volume configurations are incorrect.

1. Ensure that the same ports are exposed and mapped internally for both containers. The following environment variables must be the same for both containers.
 - ▶ `DLS_EXPOSED_HTTP_PORT`
 - ▶ `DLS_EXPOSED_HTTPS_PORT`
2. Ensure that the `DLS_RABBITMQ_SSL_PORT` environment variable is set to 5671 for both containers
3. Ensure that ports 5671, 8081 and 8084 are open on the containers within which the containerized DLS appliances are deployed.
4. Ensure that Ports 8080 through 8085 are open on each worker node.
5. Ensure that the sizes of the volumes `rabbitmq-data`, `postgres-data`, `logs` and `configurations` are equal to or exceed the minimum recommended storage space.

Volume	Minimum Recommended Storage Space
<code>rabbitmq-data</code>	2 GiB
<code>postgres-data</code>	10 GiB
<code>logs</code>	500 MiB
<code>configurations</code>	1 GiB

6. Ensure that the file access permissions on the folders inside the `rabbitmq-data` volume allow write access by the containers.

2.6.5.6. Windows Client Cannot Return Licenses Because the Return Endpoint Is Unavailable

A licensed Windows client returns a license by sending an HTTP request to the DLS instance on port 80. If the container orchestrator within which a containerized DLS appliance is deployed cannot receive requests on this port, the client cannot return a license.

How to resolve this issue depends on whether you can control the port on which the container orchestrator listens for HTTP requests.

- ▶ If you can control the port on which the container orchestrator listens for HTTP requests from licensed clients, ensure that the `DLS_EXPOSED_HTTP_PORT` environment variable is set to 80.
- ▶ Otherwise, use a load balancer to do path based routing of all license return requests to the URL `https://dls-ip-address:dls-exposed-https-port/leasing/v1/lessor/shutdown`

dls-ip-address

The IP address that is specified in the `DLS_PUBLIC_IP` environment variable.

dls-exposed-https-port

The port number that is specified in the `DLS_EXPOSED_HTTPS_PORT` environment variable.

For more information about these environment variables, refer to [Setting Properties for a Containerized DLS Software Image](#).

2.6.5.7. Client Cannot Obtain License if Client Configuration Token Specifies the Domain Name

A configuration error for a containerized DLS instance can prevent a licensed client from obtaining a license if the client configuration token specifies a fully qualified domain name. However, this error does not prevent a client from obtaining a license if the client configuration token specifies an IP address.

1. Set the `FQDN` environment variable to the fully qualified domain name that is mapped to the IP address of the container in which the DLS software image is deployed.

Ensure that this name can be resolved by licensed clients. Do **not** specify a name that can be resolved only by the container orchestrator.

2. Restart the container in which the DLS appliance is deployed.

2.6.5.8. Node Health Information Is Missing from the Service Instance Page

Insufficient storage for a containerized DLS instance can prevent node health information from being displayed on the **Service Instance** page of the management interface of the **NVIDIA Licensing** application.

1. Force reload the **Service Instance** page in the browser.

2. If the node health information is still missing after the page is reloaded, check the amount of storage space available in the volumes created for the DLS appliance container.

For more information about these volumes, refer to [Volume Mappings for a Containerized DLS Software Image](#).

3. If any volumes are full, resize the volumes and restart the container in which the DLS appliance is deployed.

2.6.5.9. The `podman-compose` Command Fails with Missing Networks Error

In some deployments, the `podman-compose up` command might fail with the error `RuntimeError: missing networks: back-tier`.

This failure is caused by a known issue with Podman Compose 1.0.3 and earlier versions. Ensure that you are using one of the supported versions of Podman Compose listed in [NVIDIA License System Release Notes](#). If necessary, install the latest version of Podman from GitHub.

```
# pip3 install https://github.com/containers/podman-compose/archive/devel.tar.gz
```

2.6.5.10. Re-deploying a Containerized DLS Software Image on Red Hat OpenShift Container Platform Fails

Re-deploying a containerized DLS software image on Red Hat OpenShift Container Platform might fail because permission issues prevent a new PostgreSQL database container from being deployed. This issue can cause an in-place upgrade to fail because an in-place upgrade requires the containerized DLS software image to be re-deployed.

When this issue occurs, the following error messages are displayed:

```
chown: /etc/dls/config/ldap3-2.9.1.dist-info: Permission denied
chown: /etc/dls/config/ldap3: Permission denied
chown: /etc/dls/config/chardet-4.0.0.dist-info: Permission denied
chown: /etc/dls/config/chardet: Permission denied
chown: /etc/dls/config/certifi-2022.12.7.dist-info: Permission denied
chown: /etc/dls/config/certifi: Permission denied
```

1. Change to the directory where the `configuration` volume is mounted.
2. Change to the `/etc/dls/config/` directory.
3. Forcibly remove the directories for which permission is denied and their contents.

```
$ cd /etc/dls/config/
$ \rm -rf ldap3-2.9.1.dist-info ldap3 \
chardet-4.0.0.dist-info chardet \
certifi-2022.12.7.dist-info certifi
```

4. Try again to deploy the containerized DLS software image on Red Hat OpenShift Container Platform.

For detailed instructions, refer to [Deploying the Containerized DLS Software Image on Kubernetes Platforms](#).

2.7. Installing the DLS Software Image on Red Hat Enterprise Linux OS

For installation on a supported operating system, the Delegated License Server (DLS) component of the NVIDIA License System is supplied as an installable package. The package includes the containerization software and container images that are required to run the **NVIDIA Licensing** application on the operating system. The operating system can be running in a virtualized server environment on your choice of hypervisor or on a bare-metal server.

Ensure that the server on which you are installing the DLS software image has at least 15 GB of free disk space.

The DLS software image ZIP archive for the Red Hat Enterprise Linux OS contains the following artifacts:

- ▶ Docker container orchestrator RPM packages
- ▶ Appliance container images
 - ▶ Database container image
 - ▶ Application container image
- ▶ Installation script with the helper scripts
- ▶ README file
- ▶ VERSION file

1. Extract the contents of the ZIP archive to a local folder.
2. Update the file access permissions of the `installAppliance.sh` file to allow execute permission.


```
$ chmod +x installAppliance.sh
```

3. Run the `installAppliance.sh` installation script.
4. When the installation is complete, confirm that messages on the console indicate that the **NVIDIA Licensing** application has started.

2.8. Configuring User Accounts on a DLS Virtual Appliance

Each DLS software image is configured with a single standard user account for accessing the DLS appliance and an account with `sudo` user privileges for installing updates to the DLS appliance software. Modifications to these accounts are strictly controlled. You cannot add other user accounts to the software image. However, you can use a lightweight directory access protocol (LDAP) directory instead of the configured accounts for managing user access to a DLS appliance.

User Account	Purpose
dls_admin	DLS administrator account. This account provides access through a web-based management interface to the NVIDIA Licensing application on a DLS virtual appliance. The DLS administrator user name can be changed from the preset <code>dls_admin</code> name.
rsu_admin	DLS <code>sudo</code> user account. This user account has the elevated privileges required to install updates to the DLS appliance software that NVIDIA releases periodically. To comply with the terms of the GPL/LGPL v3 license under which the GPL/LGPL v3 licensed Open Source Software (OSS) libraries within the DLS virtual appliance are released, this account also has the elevated privileges required to update and upgrade these libraries.

 **Note:** This account is not available for containerized DLS software images.

2.8.1. Registering the DLS Administrator User

Each DLS virtual appliance is configured with a user account specifically for administering the DLS. This account provides access through a web-based management interface to the **NVIDIA Licensing** application on the appliance. Before administering a DLS virtual appliance, you must register this user to be able to access this management interface. If you intend to configure a cluster of DLS instances, you need perform this task only for the DLS instance from which you will configure the cluster. The registration of the DLS administrator user is propagated from this instance to the other instance when you configure the cluster.

1. Open a web browser and connect to the URL `https://dls-vm-ip-address`.

dls-vm-ip-address

The IP address or, if defined, the fully qualified domain name or the CNAME of the VM on which the DLS virtual appliance is installed.

You can get the IP address from the management console of your hypervisor.

2. On the **Set Up** page that opens, click **NEW INSTALLATION**.
3. On the **Register User** page that opens, provide the credentials for the DLS administrator user.



Note: If the DLS administrator user has already been registered, the login page opens instead of the **Register User** page.

- a). **Optional:** If you want to change the user name from the preset name `dls_admin`, replace the text in the **Username** field with your choice of user name.
- b). Provide a password for the DLS administrator user and confirm the password.

The password must be at least eight characters long and is case sensitive.



Note: You can change the DLS administrator user name and password at any time after the DLS administrator user is registered. For instructions, refer to [Changing the DLS Administrator User Name and Password](#).

4. Determine whether you want to enable an additional user that will be able to access the log files for the DLS virtual appliance.
 - ▶ If you **want** to enable this additional user, ensure that the **Create a diagnostic user** option remains selected.
 - ▶ Otherwise, deselect the **Create a diagnostic user** option.
5. Click **REGISTER**.
The **Register User** page is refreshed to confirm that the user has been registered and displays a local reset secret to enable you to reset the user's password.
6. Copy the local reset secret and store it securely, for example, by clicking the clipboard icon and pasting the local reset secret into a plain text file that is readable only by you. You will need this key to reset the DLS administrator user's password.
7. Click **CONTINUE TO LOGIN**.
8. On the login page that opens, type the user name of the DLS administrator user, provide the password that you set for this user, and click **LOGIN**.

If you want to use the virtual appliance for a single DLS instance, what to do next depends on whether you intend to use a static IP address for the virtual appliance that is hosting the DLS instance.

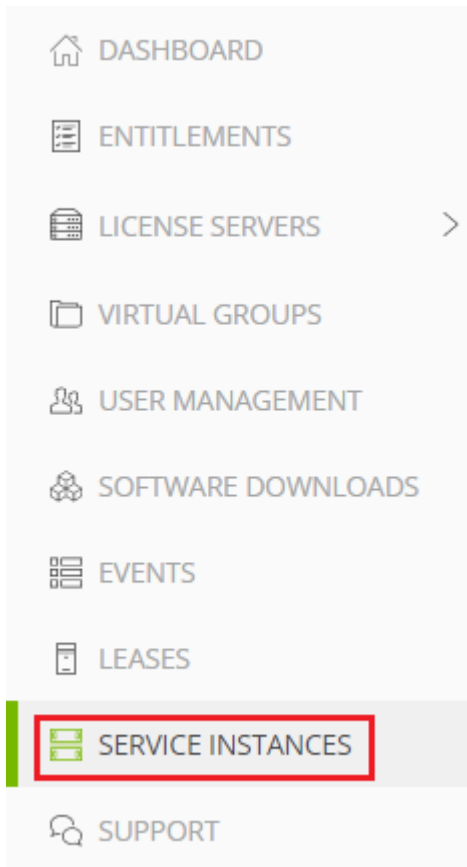
- ▶ If you want to use the virtual appliance in an HA cluster of DLS instances, configure the cluster as explained in [Configuring an HA Cluster of DLS Instances](#).
- ▶ If you want to use a static IP address for the virtual appliance that is hosting the DLS instance, set the address as explained in [Setting the Static IP Address of a DLS Virtual Appliance](#).
- ▶ Otherwise, configure the DLS instance as explained in [Configuring a Service Instance](#).

If you need to reset the DLS administrator user's password, follow the **Forgot Password?** link on the login page and, when prompted, type the local reset secret, provide a new password for this user, and confirm the new password.

2.8.2. Retrieving the DLS Administrator User's Reset Secret

If you need to reset the DLS administrator user's password but do not have the local reset secret, you can download a reset secret from the NVIDIA Licensing Portal.

1. If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
2. In the left navigation pane, click **SERVICE INSTANCES**.



3. In the list of service instances on the Service Instance page that opens, from the **Actions** menu for the DLS instance, choose **Download Reset Secret**. (Note that the menu is too narrow, so the text is truncated.)
4. When prompted, click **DOWNLOAD**.
A file named `dls_local_reset_secret_μμ-δδ-γγγγ-ηη-μμ-ss.tok` is saved to your default downloads folder.

When resetting the DLS administrator user's password, upload the reset secret to the DLS instance.

2.8.3. Changing the DLS Administrator User Name and Password

If you choose not to change the DLS administrator user name when registering the DLS administrator user, you can change it at any time after the DLS administrator user is registered. You can also change the password for the DLS administrator user.

1. Open a web browser and connect to the URL `https://dls-vm-ip-address`.

dls-vm-ip-address

The IP address or, if defined, the fully qualified domain name or the CNAME of the VM on which the DLS virtual appliance is installed.

You can get the IP address from the management console of your hypervisor.

2. At the top right of the **Dashboard** page in the **NVIDIA Licensing** application on the DLS appliance that opens, click **View settings**.
3. In the **My Info** window that opens, click **Change username/password**.
4. In the **Change Username/Password** window that opens, make the changes that you want and click **CHANGE USERNAME/PASSWORD**.
 - a). If you want to change the user name, replace the text in the **Username** field with your choice of user name.
 - b). In the **Current password** text-entry field, type the current password for the DLS administrator user.
 - c). In the **New password** text-entry field, type the password that you want for the DLS administrator user.
If you do not want to change the password, type the current password for the DLS administrator user. You cannot leave this field empty.
 - d). In the **Confirm new password** text-entry field, type the password that you typed in the **New password** text-entry field.

2.8.4. Creating the DLS `sudo` User Account

The predefined DLS `sudo` user account `rsu_admin` has the elevated privileges required to install updates to the DLS appliance software that NVIDIA releases periodically. To comply with the terms of the GPL/LGPL v3 license under which the GPL/LGPL v3 licensed Open Source Software (OSS) libraries within the DLS virtual appliance are released, this account also has the elevated privileges required to update and upgrade these libraries.

Perform this task for each DLS virtual appliance for which you want to create the DLS `sudo` user account. If the DLS virtual appliance is hosting a node in an HA cluster, the creation of the user is **not** automatically propagated to the other node in the cluster.



Note: You can perform this task **only** on a VM-based DLS virtual appliance. You **cannot** perform this task on a containerized DLS software image. Instead, you can use standard interfaces of the OS on which the container orchestration platform is running to make this change.

1. From the hypervisor, log in as the user `dls_admin` to the VM that hosts the DLS virtual appliance.
2. Run the `/etc/adminscripts/enable_sudo.sh` script.

```
$ /etc/adminscripts/enable_sudo.sh
```
3. When prompted, provide a password for the `rsu_admin` user.

The `sudo` user with elevated privileges `rsu_admin` is created.

2.8.5. Integrating a DLS Instance with an LDAP Server

You can use a lightweight directory access protocol (LDAP) directory instead of the configured accounts for managing user access to a DLS appliance. When integrated with an LDAP server, the DLS appliance uses Windows Challenge/Response, formerly NT (New Technology) LAN Manager, (NTLM) for authenticating users.

The **dls_diagnostics** and **dls_system** users have been replaced with the default user, which is the **dls_admin** user. For the initial installation of DLS and prior to registration on the UI, the default password for the **dls_admin** user is `welcome`. Once registration on the portal is done, the same password and username can be used to login to the VM Appliance from the hypervisor console.

- ▶ When a user updates the username of the DLS Admin user from UI portal on the browser, the username of the underlying provisioned user on the VM appliance also gets updated. This same behavior also applies for password updates.
- ▶ The **dls_admin** user has the same capabilities as that of **dls_system** and **dls_diagnostics**. In other words, all are able to configure appliance network properties, enable the `sudo` user, configure disk expansion, view log files, configuration files, etc.
- ▶ The **rsu_admin** user role would be used for an in-place upgrade, applying security patches, and the execution of NVIDIA-provided scripts. This is true unless it is used for OS-based package updates; in that case, there is no concern for DLS support.

If you want to enable secure LDAP (LDAPS), ensure that the following prerequisites are met:

- ▶ You have obtained the required certificate file in the correct format.
 - For a TLS certificate, the authentication certificate is the Root CA certificate in Base-64 encoded X.509 (.cer) format.
- ▶ The certificate file has the correct name.

Certificate File	File Name
TLS certificate (The Root CA Server Certificate in Base-64 encoded X.509 format)	A certificate with an extension <code>.cer</code>

For more information about LDAP, see [Basic LDAP Concepts](#).

1. Log in to the DLS virtual appliance that you are integrating with an LDAP directory.
2. In the left navigation pane, click **SETTINGS**.
The configuration for integration can be done from the **Service Instance Settings page** page that opens.

- When configuring LDAP integration, in the case of a DLS VM Appliance, users have to provide **Additional Details** to be able to integrate LDAP with the operating system, so that LDAP users can login to the VM Appliance using SSH or the hypervisor console. For more settings related to integrating LDAP with operating system, users can login to the DLS appliance VM from the hypervisor console using the **dls_admin** user, then edit the file:

```
/etc/ldap.conf
```

The logged-in user will have the same permissions as the **dls_admin** user on the VM appliance.

- Optional:** If you want to enable LDAPS, on the **Service Instance Settings page**, click **Use secure LDAP (LDAPS)?** to test the LDAPS connection.
- If LDAP users want to enable the `sudo` user **rsu_admin**, they must execute the admin script:

```
/etc/adminscripts/enable_sudo.sh
```



Note: Note that this script will not enable the **rsu_admin** user, but it will give `sudo` permissions to the currently logged-in LDAP user for a limited time of 1 hour. This will only occur if LDAP integrations have been enabled for the DLS Appliance.

- If there is no LDAP integration configured for the DLS appliance, you can create a **rsu_admin** user account with `sudo` permissions by executing the following script:

```
/etc/adminscripts/enable_sudo.sh
```

2.8.6. Logging in to a DLS Virtual Appliance

Each DLS virtual appliance is configured with a user account specifically for administering the DLS. This account provides access through a web-based management interface to the **NVIDIA Licensing** application on the appliance.

Ensure that the DLS administrator user has been registered for the appliance as explained in [Registering the DLS Administrator User](#).

- Open a web browser and connect to the URL `https://dls-vm-ip-address`.

dls-vm-ip-address

The IP address or, if defined, the fully qualified domain name or the CNAME of the VM on which the DLS virtual appliance is installed.

You can get the IP address from the management console of your hypervisor.

- On the login page that opens, provide the user credentials for the DLS administrator user on the DLS virtual appliance and click **LOGIN**.

If LDAP is used as an alternative to the fixed set of user accounts for user management, you can use the **rsu_admin** account to administer the DLS virtual appliance instead. See [Integrating a DLS Instance with an LDAP Server](#) for more information.

2.8.7. Gathering Log Files for Analysis and Troubleshooting

To collect the log files of a DLS Virtual Appliance to troubleshoot a problem, use the **dls_admin** user and run the `sudo /etc/adminscripts/collect_dls_logs.sh` command.

The `collect_dls_logs.sh` script collects the following information for analysis:

- ▶ **Critical information:**
 - ▶ CPU information from `/proc/cpuinfo`
 - ▶ RAM information from `/proc/meminfo`
 - ▶ DLS startup log file `/var/log/applicationStartup.log`
 - ▶ DLS VM appliance log file `/var/log/applianceOps.log`
 - ▶ IP address information log file `/var/log/ip_address.log`
 - ▶ Disk consumption (directories with more than 100 MB usage and database disk consumption) and free space
 - ▶ Docker logs for both containers
- ▶ **Non-critical logs:**
 - ▶ Chrony, timesyncd, and timedatectl output
 - ▶ Syslog information
 - ▶ Audit log files
- ▶ **Service log files:**
 - ▶ Log files for the DLS services

2.9. Configuring an HA Cluster of DLS Instances

To provide licensed clients with continued access to licenses if a DLS instance fails, you can configure a highly available (HA) cluster of two or more DLS instances in a failover configuration. A failover configuration consists of a *primary* instance, which is actively serving licenses to licensed clients, and one or more *secondary* instances, which act as a backup for the primary instance.

If you are configuring an HA cluster of containerized DLS instances, you **must** deploy the containers that host the instances on different Kubernetes clusters. To prevent the **NVIDIA Licensing** application from behaving abnormally, do **not** deploy the containers on different worker nodes in the same Kubernetes cluster. The application would behave abnormally because the orchestrator forwards requests to ports that are mapped onto the corresponding Kubernetes service.

If an attempt to configure an HA cluster of containerized DLS instances fails, refer to [HA Configuration or Online Upgrade Fails](#) for troubleshooting information.

2.9.1. Setting the Maximum Cluster Size for a DLS Instance

By default, a DLS instance can be added only to an HA cluster of two instances. If you want to add a DLS instance to a cluster of more than two DLS instances, you must set the maximum cluster size explicitly.

After setting the maximum cluster size for a DLS instance, you cannot change it.

How to set the maximum cluster size for a DLS instance depends on whether the instance is a VM-based instance or a containerized instance. For detailed instructions, refer to:

- ▶ [Setting the Maximum Cluster Size for a VM-Based DLS Instance](#)
- ▶ [Setting the Maximum Cluster Size for a Containerized DLS Instance](#)

2.9.1.1. Setting the Maximum Cluster Size for a VM-Based DLS Instance

Perform this task on each DLS instance that you want to add to a cluster of more than two DLS instances.

1. Use the hypervisor management console of the appliance to log in as the user **dls_admin** to the VM that hosts the DLS instance's virtual appliance.
If the **dls_admin** user has not been registered, you can still log in to the VM as the **dls_admin** user with the default password **welcome**.
2. Run `script /etc/adminscripts/enable_ha_max.sh`.

```
$ /etc/adminscripts/enable_ha_max.sh
```
3. When prompted, enter an integer in the range 2-9 that specifies the maximum number of instances in the cluster to which you want to add this instance.

After setting the maximum cluster size for the instance, add the instance to a cluster as explained in [Creating or Expanding an HA Cluster of DLS Instances](#).

2.9.1.2. Setting the Maximum Cluster Size for a Containerized DLS Instance

Perform this task on each DLS instance that you want to add to a cluster of more than two DLS instances.

1. Edit the appropriate deployment file for your container orchestration platform to uncomment the line that sets the environment variable `DLS_HA_MAX_NODES_ENV`.
To determine the deployment file for your container orchestration platform, refer to [Setting Properties for a Containerized DLS Software Image](#)

2. Set the `DLS_HA_MAX_NODES_ENV` environment variable to an integer in the range 2-9 that specifies the maximum number of instances in the cluster to which you want to add this instance.

How to set an integer value depends on your container orchestration platform. For more information, refer to the following topics:

- ▶ [Specifying Integer Values as Environment Variables](#)
- ▶ [Specifying Integer Values as Actual Values](#)

This example sets the maximum cluster size for a containerized DLS instance to 3.

```
DLS_HA_MAX_NODES_ENV=${DLS_HA_MAX_NODES_ENV:-3}
```

3. Restart the container.

After setting the maximum cluster size for the instance, add the instance to a cluster as explained in [Creating or Expanding an HA Cluster of DLS Instances](#).

2.9.2. Creating or Expanding an HA Cluster of DLS Instances

Any time after creating or configuring a standalone DLS instance, you can create an HA Cluster of DLS instances by converting the instance into a node in a cluster and adding a second instance to the cluster. You can also expand an existing cluster by adding a DLS instance to it.

Ensure that the following prerequisites are met:

- ▶ The DLS virtual appliance that will host the DLS instance to be added to the cluster has been installed and started.



Note: The version of this virtual appliance must be identical to the version of the virtual appliances that are hosting the instances that are already members of the cluster. You cannot configure an HA cluster in which the versions of the virtual appliances are different.

- ▶ The DLS administrator user has **not** been registered on the DLS virtual appliance that will host the DLS instance to be added to the cluster. The registration of the DLS administrator user is propagated to the other instance when you configure the cluster.



Note: If you are creating a cluster from a pair of newly created standalone DLS instances, ensure that the DLS administrator user has been registered **only** on one virtual appliance.

- ▶ If you are configuring a cluster of more than two nodes, ensure that the maximum cluster size has been set for each DLS instance that you want to be a member of the cluster. For detailed instructions, refer to [Setting the Maximum Cluster Size for a DLS Instance](#).

1. Log in to the DLS virtual appliance that will host or is already hosting the primary DLS instance.
 - ▶ If you are creating a cluster, log in to the DLS virtual appliance that hosts the DLS instance that you want to convert.
After the instance is converted, it will **initially** be the primary DLS instance.
 - ▶ If you want to expand an existing cluster, log in to the DLS virtual appliance that hosts the **primary** node in the cluster.
2. In the left navigation pane, click **SERVICE INSTANCE**.
3. On the **Service Instance** page that opens, under **Node Configuration**, set the **Enable High Availability** option.
The text-entry field and the **PING** button are activated, and the **CREATE HA CLUSTER** button is deactivated.
4. In the text-entry field, type the IP address or, if configured, the fully qualified domain name of the other virtual appliance to be configured in a cluster and click **PING**.
After the cluster is configured, this DLS virtual appliance will **initially** host a secondary DLS instance.
 - ▶ If the virtual appliance that will initially host the secondary DLS instance can be reached, the message `SUCCESS` appears next to the **PING** button and the **CREATE HA CLUSTER** button is activated.
 - ▶ Otherwise, the message `FAILURE` appears next to the **PING** button and the **CREATE HA CLUSTER** button remains deactivated.
5. Click **CREATE HA CLUSTER** to start the configuration and wait for it to complete.
The **Service Instance** page displays the progress of the HA cluster configuration. The configuration process takes approximately 10 minutes to complete.

When the configuration is complete, the **Service Instance** page is updated to show the node health of the cluster.

If you intend to use static IP addresses for the virtual appliances that are hosting the DLS instances in the cluster, set the address of each virtual appliance as explained in [Setting the Static IP Address of a DLS Virtual Appliance](#). Otherwise, configure the DLS instance on the virtual appliance that is hosting the **primary** DLS instance as explained in [Configuring a Service Instance](#).

If a client configuration token has already been generated for any instance in the cluster, regenerate the client configuration token for the instance as explained in [Generating a Client Configuration Token](#).

To fail over or change the roles of the DLS instances, restart the DLS virtual appliance that is hosting the **primary** DLS instance.



Note: If the instances in an HA cluster of DLS instances fail or are shut down at the same time, avoid a race condition by restarting only one instance and waiting until the startup of that instance is complete before starting the next instance.

2.9.3. Removing a Node from an HA Cluster

You can remove a **secondary** node from an HA cluster. If the secondary node is removed from a two-node cluster, the primary node is converted to a standalone DLS instance. You can perform this task to remove a secondary node even if it is down.

1. Log in to the DLS virtual appliance that hosts the **primary** node in the cluster.
2. In the left navigation pane, click **SERVICE INSTANCE**.
3. On the **Service Instance** page that opens, under **Node Health**, click **REMOVE** adjacent to the DLS virtual appliance that hosts the **secondary** node that you want to remove.
4. When asked if you want to remove the node, click **CONFIRM**.

What happens to the node that has been removed depends on the type of platform that is hosting the nodes in the cluster:

- ▶ For a cluster of instances on VM-based DLS virtual appliances, the virtual appliance that hosts the node is shut down and all data on the node is removed.
- ▶ For a cluster of instances on different container orchestrators, the container that hosts the node is **not** shut down. You must shut down the container manually.

If the secondary node is removed from a two-node cluster, the primary node is converted to a standalone DLS instance.

If a client configuration token has already been generated for any instance in the cluster, regenerate the client configuration token for the instance as explained in [Generating a Client Configuration Token](#).

If all nodes fail or are shut down, you must restart the last node to fail or be shut down **first**. If you restart the first node to fail or be shut down first, the node will not be functional until the other nodes are started.

2.9.4. Marking a Node as the Primary Node in an HA Cluster

Initially, the primary node in an HA cluster is the node that is hosted on the DLS appliance from which the cluster was created. It remains the primary node unless or until it fails, at which point failover occurs and the secondary node becomes the primary node. If you want to control which node is the primary node a cluster, you can mark a secondary node as the primary node in the cluster.

1. Log in to the DLS virtual appliance that hosts the secondary node that you want to mark as the primary node.
2. In the left navigation pane of the **NVIDIA Licensing** dashboard, click **SERVICE INSTANCE**.
3. On the **Service Instance** page that opens, locate the **Current Node Role** property and click **Mark As Primary**.

4. When asked to confirm that you want to mark the node as the primary node, click **Mark As Primary**.
The node assumes the primary role and starts to serve licenses to clients. All other nodes in the cluster, including the node that was the primary node, are marked as secondary nodes.

2.10. Configuring a VM-Based DLS Virtual Appliance

A VM-based DLS virtual appliance strictly controls access to the underlying OS. Therefore, the management interface of the **NVIDIA Licensing** application enables you to perform configuration tasks that change the configuration of the underlying OS.



Note: You can perform these tasks **only** on a VM-based DLS virtual appliance. You **cannot** perform them on a containerized DLS software image. Instead, you can use standard interfaces of the OS on which the container orchestration platform is running to make equivalent changes for a containerized DLS software image.

2.10.1. Setting the Static IP Address of a DLS Virtual Appliance

You can use the management interface to the **NVIDIA Licensing** application to replace the existing IP address of the appliance with a new static IP address. The existing IP address can be an address assigned by DHCP or another static IP address.



Note: You can perform this task **only** on a VM-based DLS virtual appliance. You **cannot** perform this task on a containerized DLS software image. Instead, you can use standard interfaces of the OS on which the container orchestration platform is running to make this change.



Note: The IP address of a VMware VM that hosts a DLS appliance can be set when the appliance is installed from an OVF template. If the IP address was set this way, you **must** follow the instructions in [Changing the IP Address of a VMware VM Set During DLS Appliance Installation](#). If you use any other method to change this address, it reverts to its original setting when the VM is restarted. In this situation, the **CONFIGURE IP ADDRESS** button is deactivated and dimmed.

The instance that the DLS virtual appliance is hosting must already be configured as a standalone DLS instance or as an instance in a HA cluster.



Note: You can set the static IP of the secondary node in an HA cluster from the primary node in the cluster.

1. If you aren't logged in already, log in to the DLS virtual appliance.
2. In the left navigation pane, click **SERVICE INSTANCE**.
3. On the **Service Instance** page that opens, under **Node Health**, click **CONFIGURE IP ADDRESS** adjacent to the DLS virtual appliance for which you are setting a static IP address.



CAUTION: If the DLS virtual appliance for which you are setting a static IP address is a node in an HA cluster and the type of any node is unknown, do not attempt to set the static IP address. Any change to the static IP address is not propagated to the node whose type is unknown because the node is unreachable.

4. In the **Configure Node IP Address** window that opens, provide the details of the IP address of the node and click **UPDATE**.
 - a). In the **Static IP address** text-entry field, type the IP address that you want to assign to the DLS virtual appliance.
 - b). In the **Gateway** text-entry field, type the IP address of the DLS virtual appliance's default gateway.



Note: If you leave the **Gateway** field empty, the DLS virtual appliance uses DHCP settings.

- c). In the **Netmask Prefix** text-entry field, type the subnet mask of the DLS virtual appliance's network in classless inter-domain routing (CIDR) format **without the leading slash character (/)**.

The netmask prefix must be an integer in the range 2-32.

To get a subnet mask in CIDR format from its decimal equivalent, refer to the table on page 2 of [IETF RFC 1878: Variable Length Subnet Table For IPv4](#).

For example, the subnet mask in CIDR format of the decimal equivalent 255.255.255.0 is 24.

- d). In the first **DNS Server** text-entry field, type the IP address of the first DNS server to be used for name resolution.
- e). In the second **DNS Server** text-entry field, type the IP address of the second DNS server to be used for name resolution.

If you are setting the IP address of the instance that you are logged in to, your browser will be disconnected from the instance after the update is complete. In this situation, you will need to log in to the DLS appliance again at the IP address that you set.



Note: Setting the IP address of an instance in an HA cluster causes a failover of the cluster. As a result of the failover, the roles of the primary and secondary instances in the cluster are reversed.

5. If necessary, log in to the DLS virtual appliance again by connecting to the URL `https://dls-vm-static-ip-address`.

dls-vm-static-ip-address

The static IP address that you set for the DLS virtual appliance.

If the DLS instance hasn't already been configured and is a standalone instance or the **primary** instance in an HA cluster, configure the instance as explained in [Configuring a Service Instance](#).

2.10.2. Reverting the Network Configuration of a DLS Appliance to DHCP

If necessary, you can revert the network configuration of a DLS appliance to DHCP after a static IP address has been assigned to the DLS appliance. After the network configuration is reverted to DHCP, the IP address of the appliance's network interface is assigned automatically. To revert the network configuration to DHCP, use the operating system command `nmcli`. Each DLS virtual appliance is configured with a user account with the elevated privileges required for running the `nmcli` command.

Perform this task from the hypervisor console, **not** from a secure shell (SSH) session. This task requires the DLS appliance's network service to be restarted, which would disconnect an SSH session.



Note: You can perform this task **only** on a VM-based DLS virtual appliance. You **cannot** perform this task on a containerized DLS software image. Instead, you can use standard interfaces of the OS on which the container orchestration platform is running to make this change.

In the `nmcli` commands in the following steps for changing the configuration of the DLS appliance's network interface, you can use the `modify` subcommand as an alternative to the `edit` subcommand.

Ensure that the `sudo` DLS user account `rsu_admin` has been created.

1. Use the hypervisor management console of the appliance to log in as the user **rsu_admin** to the VM that hosts the DLS appliance.
2. Get the connection name of the DLS appliance's network interface.

```
$ sudo nmcli connection show
```

The connection name of the DLS appliance's network interface in this example is `Wired connection 1`.

```
$ sudo nmcli connection show
```

NAME	UUID	TYPE	DEVICE
Wired connection 1	458e8070-3a5b-41a2-9946-25b519bfc8f4	ethernet	--

3. Change the assignment of an IP address to the DLS appliance's network interface from manual to automatic.

```
$ sudo nmcli conn edit "connection-name" ipv4.method auto
```

connection-name

The connection name that you obtained in the previous step.

This example changes the assignment of an IP address to the DLS appliance's network interface `Wired connection 1` from manual to automatic.

```
$ sudo nmcli conn edit "Wired connection 1" ipv4.method auto
```

4. Remove the static IP address that was set for the DLS appliance from the DNS settings of the appliance's network interface.

```
$ sudo nmcli conn edit "connection-name" -ipv4.addresses ip-address-to-remove
connection-name
```

The connection name that you specified in the previous step.

ip-address-to-remove

The IP address that you want to remove, which is the static IP address that was set for the DLS appliance's network interface.

This example removes the IP address 192.0.2.89 from the DNS settings of the DLS appliance's network interface `wired connection 1`.

```
$ sudo nmcli conn edit "Wired connection 1" -ipv4.addresses 192.0.2.89
```

5. Restart the DLS appliance's networking service.
 - a). Stop the DLS appliance's networking service.

```
$ sudo nmcli networking off
```

- b). Start the networking service.

```
$ sudo nmcli networking on
```

6. Restart the VM that hosts the DLS appliance.

2.10.3. Changing the Host Name of a VM-Based DLS Virtual Appliance

The host name of a VM-based DLS virtual appliance is preset in the virtual appliance image. If you require a specific host name, you can change the name from the hypervisor. Each DLS virtual appliance provides a shell script specifically for this purpose.



Note: You can perform this task **only** on a VM-based DLS virtual appliance. You **cannot** perform this task on a containerized DLS software image. Instead, you can use standard interfaces of the OS on which the container orchestration platform is running to make this change.

1. Use the hypervisor management console of the appliance to log in as the user `dls_admin` to the VM that hosts the DLS appliance.
If the `dls_admin` user has not been registered, you can still log in to the VM as the `dls_admin` user with the default password `welcome`.
2. Run the `/etc/adminscripts/set-hostname.sh` script.

```
$ /etc/adminscripts/set-hostname.sh
```
3. When prompted, enter your choice of new host name for the VM-based DLS virtual appliance.

2.10.4. Expanding the Disk Space on a DLS Virtual Appliance

You can use the management interface to the **NVIDIA Licensing** application to expand the disk space of the DLS virtual appliance.



Note: You can perform this task **only** on a VM-based DLS virtual appliance. You **cannot** perform this task on a containerized DLS software image. Instead, you can use standard

interfaces of the OS on which the container orchestration platform is running to make this change.

Perform this task on the Hypervisor where your DLS appliance is installed.

1. Turn off the virtual machine.
2. Expand the virtual hard disk associated to the VM through the hypervisor console.
3. Right-click on the VM and navigate to the **Edit Settings**.
4. Expand the disk space from the console.
5. Click OK to confirm.
6. Start the virtual machine.
7. Use the hypervisor management console of the appliance to log in as the user **dls_admin** to the VM that hosts the DLS virtual appliance.
If the **dls_admin** user has not been registered, you can still log in to the VM as the **dls_admin** user with the default password **welcome**.

8. Run the following script:

```
/etc/adminscripts/expand_disk.sh
```

Ignore the log message that says:

```
Information: you may need to update /etc/fstab
```

9. Validate the disk size for the `/dev/mapper/vgnls--si--0-root` using the `df -h` command.



Note:

In the case of ESXi, Hyper V, and KVM, each DLS virtual machine should be imported from the respective image of the hypervisor. If the virtual machines are cloned or created by snapshot, you will not be able to edit or expand the disk from the hypervisor console.

If you need to free up more disk space, you can log in as the **dls_admin** user and run the `sudo /etc/dls/scripts/db_data_purge.sh` command to clean up the disk.

Chapter 3. Configuring a Service Instance

How you configure a service instance depends on whether the service is a Cloud License Service (CLS) instance or a Delegated License Service (DLS) instance.

Before proceeding, determine whether you want to use an express CLS installation, a custom CLS instance, or a DLS instance to serve licenses to clients. For guidance, refer to [About Service Instances](#).

Express CLS Installation Instructions

An express CLS installation simplifies the initial configuration of a CLS instance. After you create a license server, NVIDIA License System automatically binds the license server to and installs the license server on the default CLS instance. If no default CLS instance exists, NVIDIA License System creates a default instance for you.

You administer and manage the default CLS instance through the NVIDIA Licensing Portal.

To perform an express installation, follow the instructions in [Performing an Express CLS Installation](#). No further action is required to complete the initial configuration of the CLS instance.

Custom CLS Instance Instructions

If the default CLS instance does not meet your needs, you can create a custom CLS instance. If you're creating a custom CLS instance, you must manually bind the license server to and install the license server on the CLS instance that you create.

You administer and manage a custom CLS instance through the NVIDIA Licensing Portal.

To configure a custom CLS instance, follow this sequence of instructions:

1. Instructions for creating a license server or converting a legacy NVIDIA vGPU software license server:
 - ▶ [Creating a License Server on the NVIDIA Licensing Portal](#)
 - ▶ [Converting a Legacy NVIDIA vGPU Software License Server to an NLS License Server](#)

2. [Creating a CLS Instance on the NVIDIA Licensing Portal](#)
3. [Binding a License Server to a Service Instance](#)
4. [Installing a License Server on a CLS Instance](#)

DLS Instance Instructions

You administer and manage a DLS instance through the **NVIDIA Licensing** application on the virtual appliance that hosts the DLS instance and through the NVIDIA Licensing Portal.

Before configuring a DLS instance, ensure that the virtual appliance that will host the instance has been installed and configured as explained in [Installing and Configuring the DLS Virtual Appliance](#).

To configure a DLS instance, follow this sequence of instructions:

1. **Optional:** [Changing the Name and Description of a DLS Instance](#)
2. Instructions for creating a license server or converting a legacy NVIDIA vGPU software license server:
 - ▶ [Creating a License Server on the NVIDIA Licensing Portal](#)
 - ▶ [Converting a Legacy NVIDIA vGPU Software License Server to an NLS License Server](#)
3. Instructions for registering the DLS instance:
 - ▶ [Registering an on-Premises DLS Instance with the NVIDIA Licensing Portal](#)
 - ▶ [Registering a DLS Instance on an Air-Gapped Network with the NVIDIA Licensing Portal](#)
4. [Binding a License Server to a Service Instance](#)
5. [Installing a License Server on a DLS Instance](#)

3.1. Roles Required for Configuring a Service Instance

Unless stated otherwise, the role that these tasks require depends on whether they are being performed for an organization or a virtual group.

- ▶ For an organization, these tasks require the [Organization Administrator](#) or the [Organization User](#) role.
- ▶ For a virtual group, these tasks requires the [Virtual Group Administrator](#) or the [Virtual Group User](#) role.



Note: These roles are required **only** for tasks that are performed on the NVIDIA Licensing Portal. They are **not** required for tasks that are performed on the **NVIDIA Licensing** application on the virtual appliance that hosts a DLS instance.

3.2. Proxy Server Requirements and Firewall Rules for a CLS Instance

To enable communication between a licensed client and a CLS instance through a proxy server, the proxy server must meet certain requirements. To enable communication through a firewall, firewall rules that allow traffic on specific URLs through specific ports must be defined.

The processes for configuring a proxy server and defining firewall rules are separate from the process for configuring a CLS instance. Use the standard interfaces of the proxy server and the firewall that you are using to perform these processes.

Proxy Server Requirements for a CLS Instance

NVIDIA License System supports transparent proxy servers and non-transparent proxy servers.

- ▶ **A transparent proxy server** identifies itself to the server and does **not** modify client requests and responses.
- ▶ **A non-transparent proxy server** does not reveal the IP address of the client and modifies client requests and responses.

Any proxy server between a licensed client and a CLS instance must allow programmatic calls to the URL `api.cls.licensing.nvidia.com`.

Non-Transparent Proxy Server Support

NVIDIA License System supports both authenticated and unauthenticated non-transparent proxy servers.

The following authenticated proxy servers are supported:

- ▶ Squid

The following authentication methods are supported for authenticated proxy servers:

- ▶ Basic
- ▶ Microsoft Windows Challenge/Response (Microsoft NTLM) (Windows clients only)
- ▶ Kerberos (only for clients that are a member of an Active Directory domain)

Firewall Rules for a CLS Instance

To enable communication between a licensed client and a CLS instance through a firewall, firewall rules that allow traffic on the URLs through the ports specified in the following table must be defined.

URL	Port	Traffic
api.cls.licensing.nvidia.com	443	<ul style="list-style-type: none"> ▶ Licensing operations, namely, the borrowing, renewal, and return of a license. ▶ Licensed client authentication
api.licensing.nvidia.com	80	License return from a Windows licensed client that has not been shut down cleanly

3.3. Changing the Name and Description of a DLS Instance

By default, a DLS instance is created with the name `DEFAULT_timestamp` and the description `ON_PREM_SERVICE_INSTANCE`. To distinguish a DLS instance on the NVIDIA Licensing Portal when multiple DLS instances are configured, change these defaults to a meaningful name and the description.

Perform this task from the DLS virtual appliance.

1. Log in to the DLS virtual appliance that is hosting the instance whose name and description you want to change.
2. In the left navigation pane of the **NVIDIA Licensing** dashboard, click **SERVICE INSTANCE**.
3. On the **Service Instance** page that opens, click **EDIT**.
4. In the **Edit Service Instance** dialog box that opens, type your choice of name and description for the instance and click **UPDATE**.



Note: The instance name cannot contain special characters.

The name and description of the instance are updated on the **Service Instance** page.

After changing the name of a DLS instance, follow the instructions in [Creating a License Server on the NVIDIA Licensing Portal](#).

3.4. Creating a License Server on the NVIDIA Licensing Portal

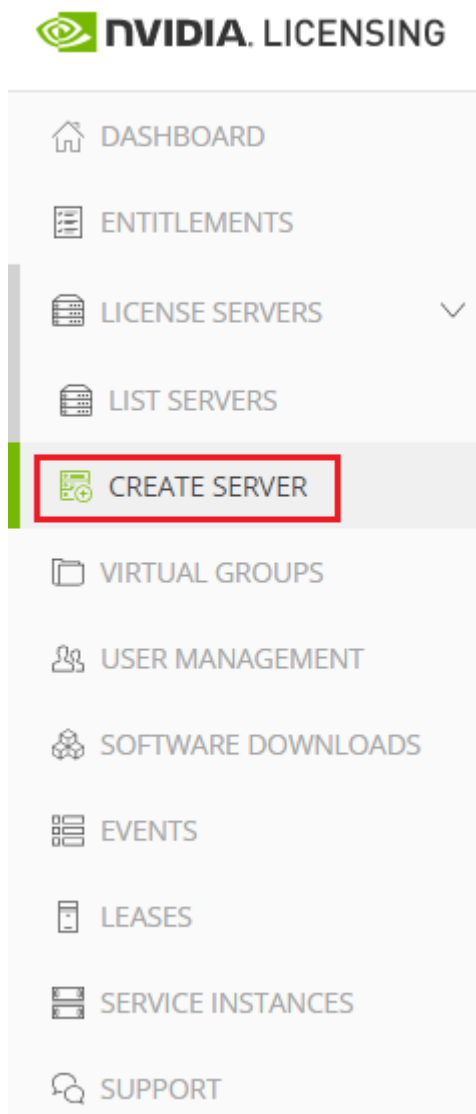
To be able to allot licenses to an NVIDIA License System instance, you must create at least one license server on the NVIDIA Licensing Portal. Creating a license server defines the set of licenses to be allotted.

You can also create multiple servers on the NVIDIA Licensing Portal and distribute your licenses across them as necessary, for example to group licenses functionally or geographically.

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which you want to create the license server.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.

If no license servers have been created for your organization or virtual group, the NVIDIA Licensing Portal dashboard displays a message asking if you want to create a license server.

2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVER** and click **CREATE SERVER**. The Create License Server wizard is started.



If you are adding a license server to an organization or virtual group for which a license server has already been created, click **CREATE SERVER**.

The **Create License Server** wizard opens.

3. On the Create License Server page of the wizard, step through the configuration requirements to provide the details of your license server.
 - a). **Step 1 – Identification:** In the **Name** field, enter your choice of name for the license server and in the **Description** field, enter a text description of the license server.
The description is required and will be displayed on the details page for the license server that you are creating.
 - b). **Step 2 – Features:** Select one or more available features from your entitlements to allot to this license server.
 - c). **Step 3 - Environment:** Select **Cloud (CLS)** or **On-Premises (DLS)** to install this license server.
To make the selection after the license server has been created, select the **Deferred** option.
 - d). **Step 4 – Configuration:** From the **Leasing mode** drop-down list, select one of the following leasing modes:
 - Standard Networked Licensing**
Select this mode to simplify the management of licenses on a license server that supports networked licensing. In this mode, no additional configuration of the licenses on the server is required.

Advanced Networked Licensing

Select this mode if you require control over the management of licenses on a license server that supports networked licensing. This mode requires additional configuration to create license pools and fulfillment conditions on the server. For more information, refer to [Managing License Pools](#) and [Managing Fulfillment Conditions](#).

Node-Locked Licensing

Select this mode **only** if the license server will serve clients that cannot obtain a license from a remote license server over a network connection. In this mode, the clients obtain a node-locked license from a file installed locally on the client system. For more details, refer to [Generating Node-Locked Licenses](#).



CAUTION: This mode requires additional work to create the license file to be installed locally and to return licenses when the client is shut down. If this mode is set, the mode of the license server **cannot** be changed.

- e). Click **REVIEW SUMMARY** to review the configuration summary before creating the license server.
4. On the Create License Server page, from the **Step 4 – Configuration** menu, click the **CREATE SERVER** option to create this license server.
Alternatively, you can click **CREATE SERVER** on the Server Summary page.

After creating a license server on the NVIDIA Licensing Portal, follow the instructions in the topic for the type of service instance that you are configuring:

- ▶ **CLS instance:** [Creating a CLS Instance on the NVIDIA Licensing Portal](#)
- ▶ **DLS instance:** [Registering an on-Premises DLS Instance with the NVIDIA Licensing Portal](#)

3.5. Performing an Express CLS Installation

Performing an express CLS installation creates a license server that NVIDIA License System automatically binds to and installs on the default CLS instance. The license server that you create defines the set of licenses to be allotted to an NVIDIA License System instance.

If no default CLS instance exists, NVIDIA License System creates a default instance for you. After you perform an express installation, no further action is required to complete the initial configuration of the CLS instance. The instance is ready to serve licenses to clients.

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which you want to perform an express CLS installation.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.

- b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.

If no license servers have been created for your organization or virtual group, the NVIDIA Licensing Portal dashboard displays a message asking if you want to create a license server.

2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVER** and click **CREATE SERVER**.

If you are adding a license server to an organization or virtual group for which a license server has already been created, click **CREATE SERVER**.

The **Create License Server** wizard opens.

3. On the **Step 1 - Identification** page of the wizard, provide the details of your license server.
 - a). In the **Name** field, enter your choice of name for the license server.
 - b). In the **Description** field, enter a text description of the license server.

This description is required and will be displayed on the details page for the license server that you are creating.
 - c). Click **NEXT STEP**.
4. On the **Step 2 - Features** page of the wizard, add the licenses for the products that you want to allot to this license server.

For each product, add the licenses as follows:

 - a). In the list of products, select the product for which you want to add licenses.
 - b). In the text-entry field in the **ADDED** column, enter the number of licenses for the product that you want to add.
 - c). Click **NEXT STEP**.
5. On the **Step 3 - Environment** page, select **Cloud (CLS)**, select the **Express installation** option that is added to the page, and click **NEXT STEP**.
6. On the **Step 4-Configuration** page, select the leasing mode that you require.

If the license server is to be used for networked licensing, you can simplify the management of licensed products on the server by selecting the **Standard Networked Licensing** mode.
7. Click **CREATE SERVER**.

3.6. Converting a Legacy NVIDIA vGPU Software License Server to an NLS License Server

To simplify the process of migrating to NLS and to preserve the distribution of licenses on your existing fleet of legacy NVIDIA vGPU software license servers, you can convert each of your existing license servers to an NLS license server. When a legacy NVIDIA

vGPU software license server is converted, the name of the license server and the licensed products allotted to the server are preserved.

Converting a legacy NVIDIA vGPU software license server to an NLS license server affects the server **only** on the NVIDIA Licensing Portal. The legacy NVIDIA vGPU software license server continues to serve licenses to clients until you reconfigure your clients to use the converted NLS license server.

How to convert a legacy NVIDIA vGPU software license server to an NLS license server depends on the type of service instance on which you want the converted license server to be installed. For instructions, refer to the following topics:

- ▶ [Converting a Legacy License Server to an NLS License Server on an Express CLS Installation](#)
- ▶ [Converting a Legacy License Server to an NLS License Server on a Custom CLS Instance](#)
- ▶ [Converting a Legacy License Server to an NLS License Server on a DLS Instance](#)

The conversion process sets the licensing mode of the converted license server to **Standard Networked Licensing**. If you want a different mode for the converted license server, change it as explained in [Changing the Leasing Mode of a License Server](#).

3.6.1. Converting a Legacy License Server to an NLS License Server on an Express CLS Installation

An express CLS installation simplifies the conversion of a legacy NVIDIA vGPU software license server. As part of the conversion, NVIDIA License System automatically binds the license server to and installs the license server on the default CLS instance. If no default CLS instance exists, NVIDIA License System creates a default instance for you.

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group to which the legacy license server belongs.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVERS** and click **LIST SERVERS**.
3. In the list of license servers on the **License Servers** page that opens, from the **Actions** menu for the legacy license server, choose **Migrate to NLS**.
4. In the **Migrate Legacy License Server to NVIDIA License System** window that opens, select **Cloud (CLS)**, leave the **Express installation** option that is added to the page selected, and click **MIGRATE LICENSE SERVER**.

The license server that you selected is converted and is automatically bound to and installed on the default CLS instance. In the list on the **License Servers** page, the **Legacy** label against the name of the converted server is removed.

3.6.2. Converting a Legacy License Server to an NLS License Server on a Custom CLS Instance

If the default CLS instance does not meet your needs, you can use a custom CLS instance for a converted legacy NVIDIA vGPU software license server. You can use an existing instance or create the instance during the conversion.

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group to which the legacy license server belongs.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVERS** and click **LIST SERVERS**.
3. In the list of license servers on the **License Servers** page that opens, from the **Actions** menu for the legacy license server, choose **Migrate to NLS**.
4. In the **Migrate Legacy License Server to NVIDIA License System** window that opens, select **Cloud (CLS)** and **deselect** the **Express installation** option that is added to the page.
5. Select or create the custom CLS instance on which you want the converted license server to be installed.
 - ▶ Select an existing CLS instance from the drop-down list.
 - ▶ Create a CLS as follows:
 - a). In the **Name** field, enter your choice of name for the license server.
 - b). In the **Description** field, enter a text description of the license server.

This description is required and will be displayed on the details page for the license server that you are creating.
 - c). Click **CREATE SERVICE INSTANCE**.
6. Click **MIGRATE LICENSE SERVER**.

The license server that you selected is converted and is automatically bound to and installed on the CLS instance that you selected or created. In the list on the **License Servers** page, the **Legacy** label against the name of the converted server is removed.

3.6.3. Converting a Legacy License Server to an NLS License Server on a DLS Instance

If you want the converted legacy NVIDIA vGPU software to be hosted on-premises at a location that is accessible from your private network, such as inside your data center, you can use a custom DLS instance. You can use an instance that is already registered or register the instance during the conversion.

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group to which the legacy license server belongs.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVERS** and click **LIST SERVERS**.
3. In the list of license servers on the **License Servers** page that opens, from the **Actions** menu for the legacy license server, choose **Migrate to NLS**.
4. In the **Migrate Legacy License Server to NVIDIA License System** window that opens, select **On-Premises (DLS)**.
5. Select or register the DLS instance to which you want the converted license server to be bound.
 - ▶ Select a registered DLS instance from the drop-down list.
 - ▶ Register an unregistered DLS instance as follows:
 - a). Click **SELECT INSTANCE TOKEN**.
 - b). In the file browser that opens, navigate to the folder that contains the downloaded DLS instance token file and select the file.

The file is named `dls_instance_token_mm-dd-yyyy-hh-mm-ss.tok`.
 - c). Back in the **Migrate Legacy License Server to NVIDIA License System** window, click **UPLOAD TOKEN**.
6. Click **MIGRATE LICENSE SERVER**.

The license server that you selected is converted and is automatically bound to the DLS instance that you selected or registered. In the list on the **License Servers** page, the **Legacy** label against the name of the converted server is removed.

After the conversion is complete, install the converted license server on the DLS instance that you selected or registered as explained in [Installing a License Server on a DLS Instance](#).

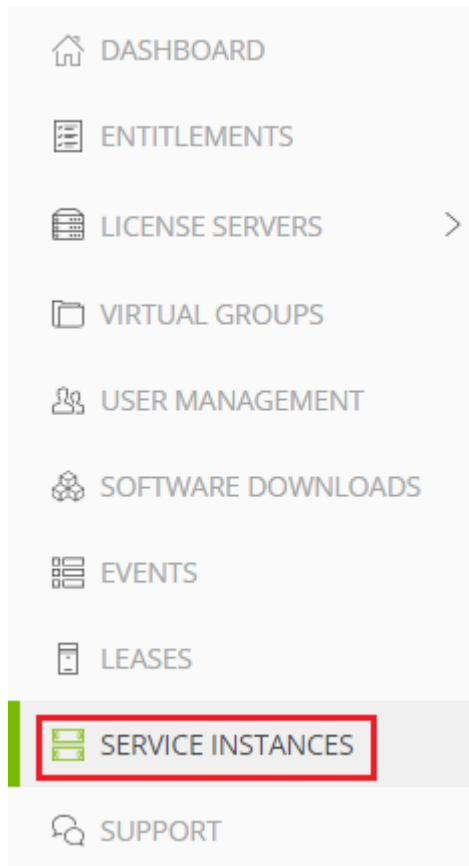
3.7. Creating or Registering a Service Instance

If you are hosting your service instance in the cloud on the NVIDIA Licensing Portal, you must create a CLS instance. If you are hosting your service instance at a location that is accessible from your organization's private network, you must register a DLS instance. If you are hosting your service instance in the cloud on the NVIDIA Licensing Portal, you must create a CLS instance only if you are not using the default CLS instance.

3.7.1. Creating a CLS Instance on the NVIDIA Licensing Portal

When you create a CLS instance, the instance is automatically registered with the NVIDIA Licensing Portal. This task is only necessary if you are not using the default CLS instance. Service instances belong to an organization. Therefore, this task requires the [Organization Administrator](#) role.

1. If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **SERVICE INSTANCES**.



3. On the Service Instances page, from the **Actions** menu, choose **Create cloud (CLS) instance**.

The **Create cloud (CLS) instance** pop-up window opens.

4. Provide the details of your cloud service instance.
 - a). In the **Name** field, enter your choice of name for the service instance.
 - b). In the **Description** field, enter a text description of the service instance.

This description is required and will be displayed on the **Service Instances** page when the entry for service instance that you are creating is expanding.
5. Click **CREATE CLS INSTANCE**.

After creating a CLS instance on the NVIDIA Licensing Portal, follow the instructions in [Binding a License Server to a Service Instance](#).

3.7.2. Registering an on-Premises DLS Instance with the NVIDIA Licensing Portal

A DLS instance is created automatically when the virtual appliance on which the instance resides is installed. However, to enable the instance to be bound to a license server, you must register the instance with the NVIDIA Licensing Portal.

Registering an on-premises DLS instance with the NVIDIA Licensing Portal involves the exchange of a **DLS instance token** between the instance and the NVIDIA Licensing Portal.

A DLS instance token is created by a DLS instance. It identifies the DLS instance to the NVIDIA Licensing Portal and enables it to locate the NVIDIA Licensing Portal. After downloading the token from the DLS instance, you must upload it to the NVIDIA Licensing Portal to complete the registration of the service instance.

1. If you are not already logged in, log in to the **NVIDIA Licensing** application at the IP address of the VM on which the DLS virtual appliance is installed.
2. In the left navigation pane of the **NVIDIA Licensing** dashboard, click **SERVICE INSTANCE**.
3. On the **Service Instance Details** page that opens, from the **ACTIONS** menu, choose **Download DLS Instance Token**.
A DLS instance token file that is named `dls_instance_token_mm-dd-yyyy-hh-mm-ss.tok` is downloaded.
4. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which you are registering the service instance.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
5. On **Service Instances** page that opens, from the **Actions** menu, choose **Register DLS Instance**.
6. In the **Register DLS Instance** window that opens, select the **New installation** option and click **SELECT INSTANCE TOKEN**.

Register DLS Instance ×

Upload a DLS instance token for initial registration or for a DLS upgrade

Are you registering a new DLS instance or performing an upgrade on an existing DLS instance?

New installation Upgrade existing

SELECT INSTANCE TOKEN UPLOAD TOKEN

7. In the file browser that opens, navigate to the folder that contains the DLS instance token file that is named `dls_instance_token_mm-dd-yyyy-hh-mm-ss.tok` that you downloaded and select the file.
8. Back in the **Register DLS Instance** window, click **UPLOAD TOKEN**.
The service instance is added to the list of registered service instances.

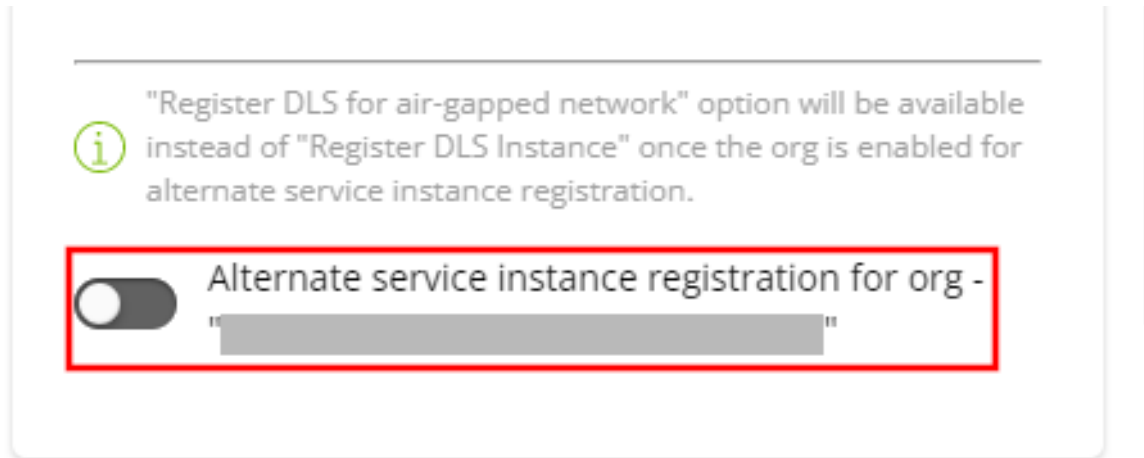
After registering an on-premises DLS instance with the NVIDIA Licensing Portal, follow the instructions in [Binding a License Server to a Service Instance](#).

3.7.3. Enabling your Organization to Register DLS Instances Manually

By default, users in your organization must register DLS instances by downloading DLS instance tokens to upload to the NVIDIA Licensing Portal. If you want users in your organization to be able to register DLS instances by entering their details manually on the NVIDIA Licensing Portal, you must enable this option explicitly.

This task requires the [Organization Administrator](#) role.

1. If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
2. Click **View settings** at the top right of the page.
3. In the **My Info** window that opens, select the **Alternate service instance registration** option.



4. Close the **My Info** window.

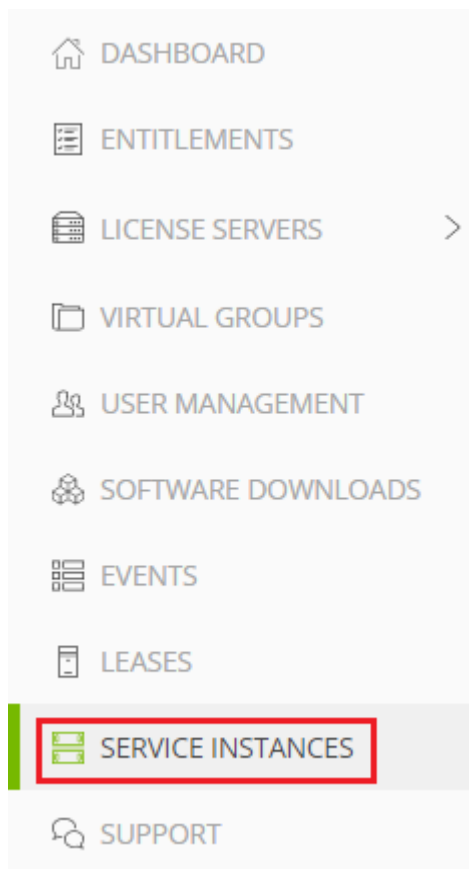
If you want users in your organization to be able again to register DLS instances by downloading DLS instance tokens to upload to the NVIDIA Licensing Portal, deselect the **Alternate service instance registration** option.

3.7.4. Registering a DLS Instance on an Air-Gapped Network with the NVIDIA Licensing Portal

If your DLS instance is on an air-gapped network, you cannot download a DLS instance token to upload to the NVIDIA Licensing Portal to register the instance. Instead, you must register the instance by entering its details manually on the NVIDIA Licensing Portal. Before you begin, ensure that your organization administrator has enabled your organization to register DLS instances manually. For more information, refer to [Enabling your Organization to Register DLS Instances Manually](#).

1. If you are not already logged in, log in to the **NVIDIA Licensing** application at the IP address of the VM on which the DLS virtual appliance is installed.
2. In the left navigation pane of the **NVIDIA Licensing** dashboard, click **SERVICE INSTANCE**.
3. On the **Service Instance Details** page that opens, from the **ACTIONS** menu, choose **Pre-Register Service Instance**.
4. In the **Pre-Register DLS Service Instance** window that opens, click **PRE-REGISTER SERVICE INSTANCE**.
5. On the **Service Instance** page, note the information about the DLS instance that you will need when you register it by entering its details manually on the NVIDIA Licensing Portal.
 - ▶ Service Instance ID
 - ▶ Name

- Description
6. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which you are registering the service instance.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
 7. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **SERVICE INSTANCES**.



8. On **Service Instances** page that opens, from the **ACTIONS** menu, choose **Register DLS for air-gapped network**.
9. In the **Register DLS Instance for Air-Gapped Network** window that opens, enter the name, service instance ID, and description that you obtained in Step 5, and click **REGISTER**.

Register DLS Instance For Air-Gapped Network



Service Instance ID

Service instance ID of a DLS instance hosted on an air-g

Name

Enter a name for this service instance

Description

Enter a description for this service instance

 REGISTER

After registering the DLS instance, perform the following sequence of tasks to complete the configuration of the instance:

1. Bind a license server to the DLS instance that you registered as explained in [Binding a License Server to a Service Instance](#).
2. Download and install the license server on the virtual appliance that hosts DLS instance as explained in [Installing a License Server on a DLS Instance](#).

3.8. Deleting a Service Instance

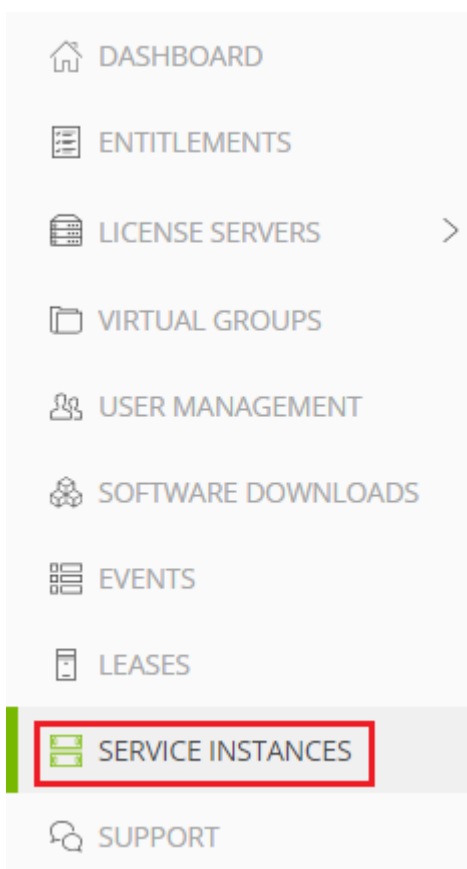
When a service instance is deleted, any license servers that are bound to and installed on the service instance are uninstalled and freed from it. Deleting a **DLS** instance on which license servers are installed forcibly removes all licensed products from the servers and returns the licensed products to their entitlements. This behavior enables you to recover licenses from a failed DLS instance.

This task requires the [Organization Administrator](#) role.

If you are deleting a CLS instance on which license servers are installed, remove all licensed products from the servers as explained in [Managing Licenses and Licensed Products on a License Server](#).

Perform this task on the NVIDIA Licensing Portal. The procedure for deleting a service instance is the same for CLS instances and DLS instances.

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group to which the service instance belongs.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the My Info window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the left navigation pane, click **SERVICE INSTANCE**.



3. In the list of service instances on the Service Instances page that opens, from the **Actions** menu for the service instance, choose **Delete**.
4. When asked to confirm that you want to delete the service instance, click **DELETE**.

3.9. Binding a License Server to a Service Instance

Binding a license server to a service instance ensures that licenses on the server are available only from that service instance. As a result, the licenses are available only to the licensed clients that are served by the service instance to which the license server is bound.

You can bind multiple license servers to the same CLS instance but only one license server to the same DLS instance. If you want to use a different license server than the license server that was originally bound to a DLS instance, free the license sever as explained in [Freeing a License Server from a Service Instance](#).

This task is necessary only if you are not using the default CLS instance.

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group to which the **license server** belongs.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVERS** and click **LIST SERVERS**.
3. In the list of license servers on the **License Servers** page that opens, from the **Actions** menu for the license server, choose **Bind**.
4. In the **Bind Service Instance** pop-up window that opens, select the service instance to which you want to bind the license server and click **BIND**.
The **Bind Service Instance** pop-up window confirms that the license server has been bound to the service instance.

After a license server has been bound to a service instance, the license server is freed from the service instance when the service instance is deleted. You can also free a license sever as explained in [Freeing a License Server from a Service Instance](#).

After binding a license server to a service instance, follow the instructions in the topic for the type of service instance that you are configuring:

- ▶ **CLS instance:** [Installing a License Server on a CLS Instance](#)
- ▶ **DLS instance:** [Installing a License Server on a DLS Instance](#)

3.10. Freeing a License Server from a Service Instance

If you want to move a license server to a different service instance, free the license server before binding it to and installing it on the new service instance. If you want to use a different license server than the license server that was originally bound to a DLS instance, free the license sever from the DLS instance first.

Ensure that **one** of the following prerequisites is met:

- ▶ The license server has not been installed on the service instance.
- ▶ No licenses on a license server that is already installed are checked out by licensed clients.

What happens if this prerequisite is not met depends on the type of the service instance:

- ▶ **CLS instance:** The NVIDIA Licensing Portal prevents the license server from being freed from the service instance.
- ▶ **DLS instance:** Because the instance is not connected to the NVIDIA Licensing Portal, the NVIDIA Licensing Portal allows the license server to be freed from the service instance. However, any attempt to install a license server on the service instance fails.

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group to which the **license server** belongs.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVERS** and click **LIST SERVERS**.
3. In the list of license servers on the **License Servers** page that opens, from the **Actions** menu for the license server, choose **Unbind**.
4. When prompted to confirm that you want to unbind the license server, click **UNBIND**.

After freeing the license server, you can reuse it by following this sequence of instructions:

1. [Binding a License Server to a Service Instance](#)
2. [Installing a License Server on a Service Instance](#)

3.11. Installing a License Server on a Service Instance

After binding a license server to a service instance, you must install the license server on the service instance to make the licenses on the server available to the instance. If you change the licenses or licensed products on a license server that is bound to a DLS instance, you must update the instance with the latest version of the license server. This task is necessary only if you are not using the default CLS instance.

How to install a license server on a service instance depends on whether you are installing the license server on a CLS instance or a DLS instance. For detailed instructions, see:

- ▶ [Installing a License Server on a CLS Instance](#)
- ▶ [Installing a License Server on a DLS Instance](#)

After creating and installing a license server on a service instance, manage the licenses on the server by creating a client configuration token and, optionally, creating license pools and fulfillment conditions. For more information, see [Managing Licenses on a License Server](#).

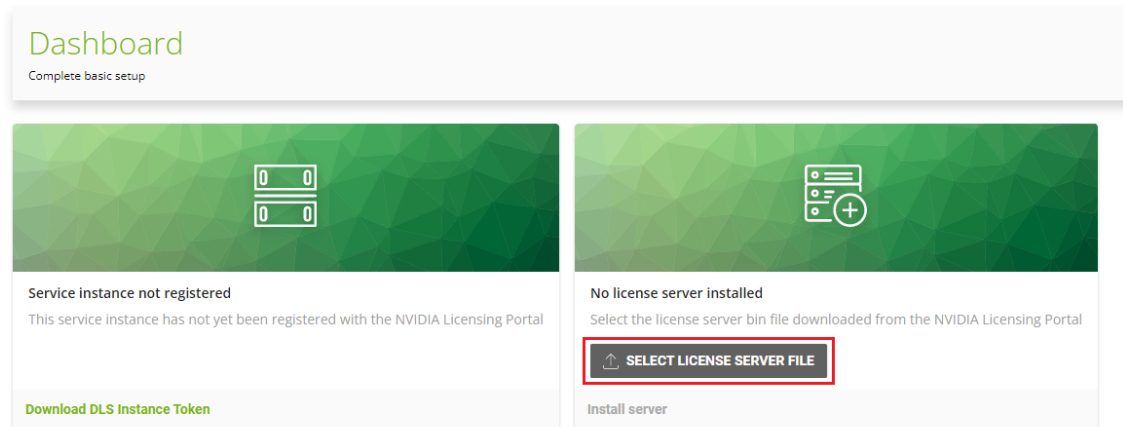
3.11.1. Installing a License Server on a CLS Instance

This task is necessary only if you are not using the default CLS instance.

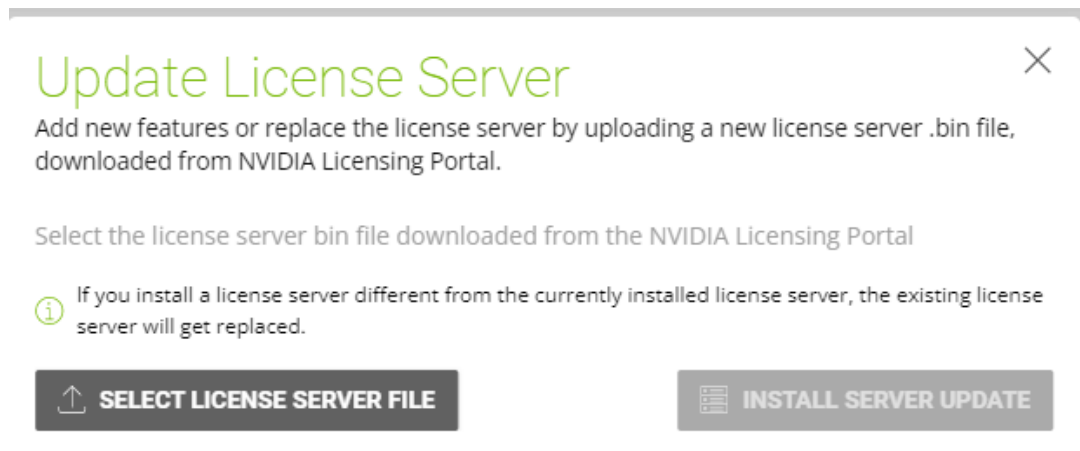
1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which you want to install the license server.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the My Info window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVER** and click **LIST SERVERS**.
3. In the list of license servers on the **License Servers** page that opens, click the name of the license server that you want to install.
4. In the **License Server Details** page that opens, from the **Actions** menu, choose **Install**.
5. In the **Install License Server** pop-up window that opens, click **INSTALL SERVER**.

3.11.2. Installing a License Server on a DLS Instance

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which the license server was created.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the My Info window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVER** and click **LIST SERVERS**.
3. In the list of license servers on the License Servers page that opens, click the name of the license server that you want to install.
4. In the **License Server Details** page that opens, from the **Actions** menu, choose **Download**.
5. In the **Download License File** window that opens, click **Download**.
A license server file that is named `license_mm-dd-yyyy-hh-mm-ss.bin` is downloaded.
6. If you are not already logged in, log in to the **NVIDIA Licensing** application at the IP address of the VM on which the DLS virtual appliance is installed.
After you log in, the information that is displayed on the **NVIDIA Licensing** dashboard depends on whether a license server has already been installed on the DLS virtual appliance.
 - ▶ If a license server has **not** been installed on the DLS virtual appliance, the **NVIDIA Licensing** dashboard displays a message asking if you want to install a license server.
 - ▶ Otherwise, the **NVIDIA Licensing** dashboard displays the **License Server Details** page for the installed license server.
7. Install or update the license server on the DLS virtual appliance.
Whether you install or update the license server depends on whether a license server has already been installed on the DLS virtual appliance.
 - ▶ If a license server has **not** been installed on the DLS virtual appliance, on the **NVIDIA Licensing** dashboard, click **SELECT LICENSE SERVER FILE**.



- ▶ If a license server has already been installed on the DLS virtual appliance, update the license server.
 - a). From the **ACTIONS** menu for the license server, choose **Update server from NLP**.
 - b). In the **Update License Server** pop-up window that opens, click **SELECT LICENSE SERVER FILE**.



8. In the file browser that opens, navigate to the folder that contains the license server file named `license_mm-dd-yyyy-hh-mm-ss.bin` that you downloaded and select the file.
9. When asked if you want to install the selected file, click **INSTALL**.

NVIDIA Licensing dashboard is updated with the details of the license server that you installed.



Note: If you updated an existing license server, the licenses on the new server are **not** allotted to any existing license pools. You must allot these licenses as explained in [Managing License Pools](#)

3.12. Changing the Leasing Mode of a License Server

You can change the leasing mode of a license server after it has been created.



Note: You cannot change the leasing mode of a license server that is bound to and installed on a DLS instance. You must first free the license server from the DLS instance. For instructions, see [Freeing a License Server from a Service Instance](#).

1. Navigate to the License Server Details page of the license server on which you want to change the leasing mode.
For instructions, see [Navigating to the License Server Details Page for a License Server](#).
2. In the License Server Details page that opens, from the **ACTIONS** menu, choose **Change leasing mode**.
3. In the Leasing Mode pop-up window, select a desired leasing mode and then click **CHANGE LEASING MODE**.



Note: If you select the **Node Locked** option, you cannot revert this leasing mode change on the server.

Chapter 4. Managing Licenses on a License Server

After installing a license server on a service instance, you can manage the licenses to be served from the server by distributing them among a number of license pools and by defining fulfillment conditions for requests from licensed clients. In this way, you can reserve licenses for specific types of users.

For example:

- ▶ An organization in which some users are using graphics-intensive computer-aided design (CAD) tools, while other users are using only office productivity tools can create a pool of NVIDIA RTX Virtual Workstation licenses for the CAD tool users and a pool of GRID Virtual PC licenses for the users of office productivity tools.
- ▶ An organization in which some users are performing mission-critical tasks can create a reserve pool of licenses available only to these users and a pool of licenses available to all users. By setting suitable fulfillment conditions, the organization can ensure that when the pool of licenses available to all users is exhausted, only license requests from users performing mission-critical tasks are fulfilled from the reserve pool.

When a license server is installed on a service instance, a single default license pool and a single default fulfillment condition are created on the server. The default license pool initially contains all licenses allotted to the server. The default fulfillment condition allows any client to be served from the default license pool. If you want all your licensed clients to be served licenses from the same license pool under the same conditions, you can generate a client configuration token without creating any license pools or fulfillment conditions. For more information about client configuration tokens, see [Generating a Client Configuration Token](#).

4.1. Where to Perform Tasks for Managing Licenses

Where to perform the tasks for managing licenses on a license server depends on the type of service instance on which the license server is installed.

- ▶ On a CLS instance, perform the tasks on the NVIDIA Licensing Portal.
- ▶ On a DLS instance, perform the tasks on the **NVIDIA Licensing** application on the virtual appliance that hosts the DLS instance.

4.2. Roles Required for Managing Licenses on a CLS Instance

The role that these tasks require depends on whether they are being performed for an organization or a virtual group.

- ▶ For an organization, these tasks require the [Organization Administrator](#) or the [Organization User](#) role.
- ▶ For a virtual group, these tasks require the [Virtual Group Administrator](#) or the [Virtual Group User](#) role.



Note: These roles are required **only** for tasks that are performed for a CLS instance on the NVIDIA Licensing Portal. They are **not** required for tasks that are performed for a DLS instance on the **NVIDIA Licensing** application on the virtual appliance that hosts the DLS instance.

4.3. Navigating to the License Server Details Page for a License Server

How to navigate to the **License Server Details** page for a license server depends on whether you are performing the task on the NVIDIA Licensing Portal on a DLS instance.

1. If you are not already logged in, log in to the web user interface for administering the license server.
 - ▶ **On the NVIDIA Licensing Portal**, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
The NVIDIA Licensing Portal dashboard opens.
 - ▶ **On a DLS instance**, log in to the **NVIDIA Licensing** application at the IP address of the VM on which the DLS virtual appliance is installed.
The **License Server Details** page for the license server on the DLS virtual appliance opens. No further action is required.

The remaining steps are required for **the NVIDIA Licensing Portal only**.

2. If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the My Info window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
3. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVER** and click **LIST SERVERS**.
4. In the list of license servers on the License Servers page that opens, click the name of the license server for which you want to manage licenses.

4.4. Managing License Pools

License pools enable you to divide the product features on a license server so that different categories of users are served licenses from different license pools. All licenses are served from a license pool. Only licenses on a license server that belong to a license pool are available to be served to license clients.

When a license server is installed on a service instance, a single default license pool is created on the server. The default license pool initially contains all licenses allotted to the server. You can subdivide the licenses on license server into any number of license pools. However, if you want to serve all licenses on a server from a single pool, you can use the default license pool without creating any additional license pools.

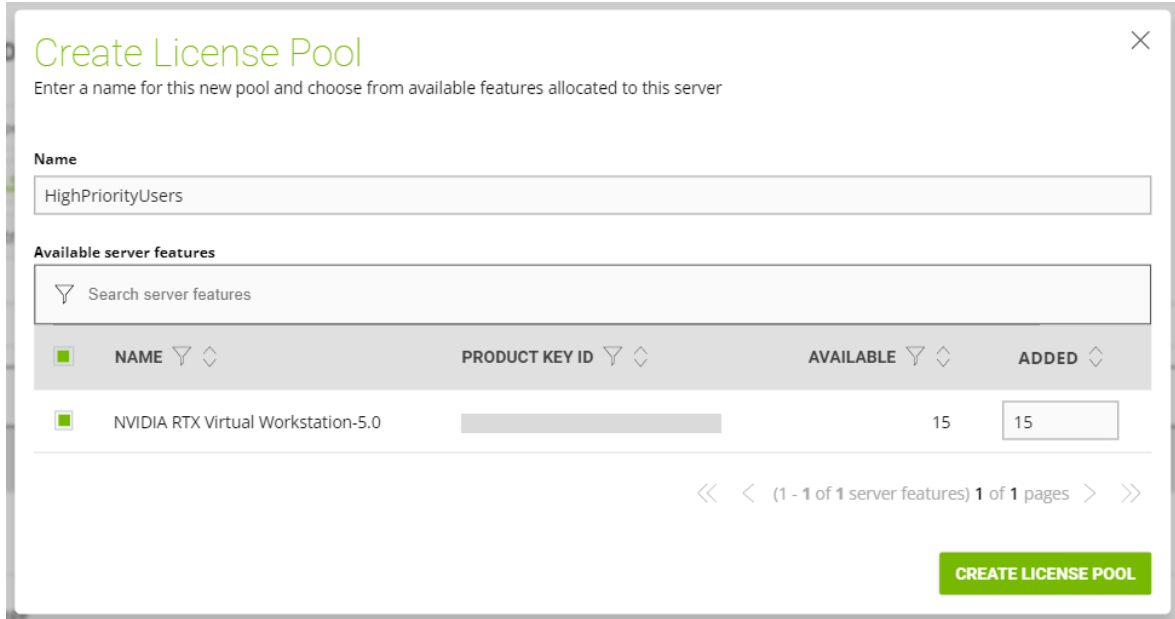


Note: A license server for which the **Node-locked licensing mode?** option is set does not support multiple license pools. All licenses on the server remain in the default license pool.

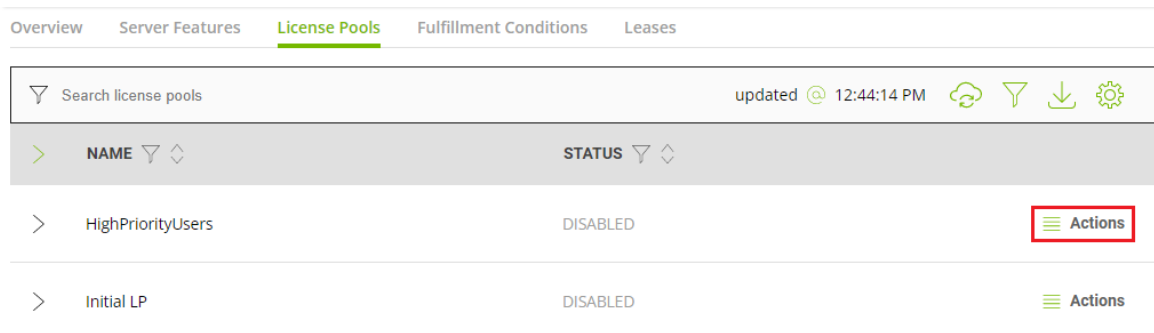
4.4.1. Creating a License Pool

If the only license pool on the license server is the initial default pool, return any licenses that you want to allocate to the license pool to the license server as explained in [Managing Licenses and Licensed Products in a License Pool](#). When a license server is created, a default license pool that contains all licenses on the server is created. When the default license pool contains all licenses on the server, the **CREATE LICENSE POOL** button is inactive.

1. Navigate to the **License Server Details** page of the license server on which you want to create the license pool.
For instructions, see [Navigating to the License Server Details Page for a License Server](#).
2. In the **License Server Details** page that opens, disable the license server by clicking **DISABLE SERVER** and, when prompted, confirm that you want to disable the license server.
When the license server is disabled, it cannot serve licenses to licensed clients.
3. From the **ACTIONS** menu, choose **Create Pool**.
The **Create License Pool** pop-up window opens.



4. In the **License Pool Name** field, enter your choice of name for the license pool.
The name must be a list four characters long, may contain only letters and numbers, and must not contain any spaces or special characters.
5. Add the licenses for the products that you want to allot to this license pool.
For each product, add the licenses as follows:
 - a). From the list of available features, select the product for which you want to add licenses.
In the list of available features, only products on the license server that do not belong to another license pool are listed.
 - b). In the text-entry field in the **ADDED** column, enter the number of licenses for the product that you want to add.
6. Click **CREATE LICENSE POOL**.
The license pool is added to the list of license pools on the **License Pools** tab, in the **License Server Details** page.



- On the Overview tab of the License Server Details page, enable the license server by clicking **ENABLE SERVER** and, when prompted, confirm that you want to enable the license server.

The license server can now serve licenses to licensed clients.

After you create a license pool you can change the set of licenses in the pool as explained in the following topics:

- ▶ [Managing Licenses and Licensed Products in a License Pool](#)
- ▶ [Migrating Licenses Between License Pools](#)

4.4.2. Deleting a License Pool

When a license pool is deleted, all product features in the pool are returned to the license server to which the license pool belongs.

Ensure that no licenses in the pool are checked out by licensed clients. A license pool cannot be deleted while any of the licenses in the pool are checked out by a client.

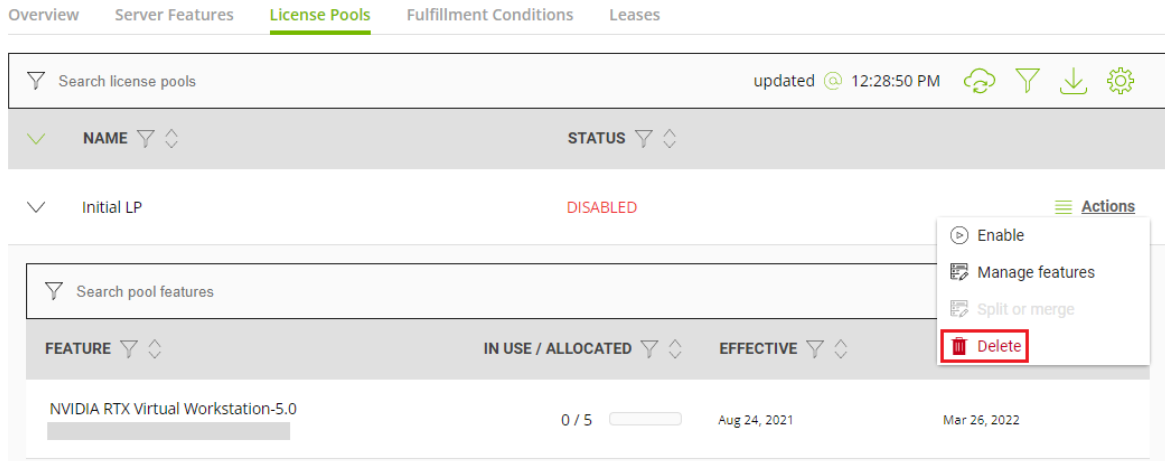
- Navigate to the **License Server Details** page of the license server to which the license pool belongs.

For instructions, see [Navigating to the License Server Details Page for a License Server](#).

- In the **License Server Details** page that opens, click the **License Pools** tab.

The screenshot shows the 'License Server Details' page. At the top, there is a header with 'License Server Details' and a 'Help?' link. Below the header, there are 'REFRESH' and 'ACTIONS' buttons. The main content area displays a card for a license pool named 'Example_CLS' which is 'DISABLED'. The card includes the following information: Status: DISABLED, Type: NVIDIA, Created: Jan 20, 2022 3:27 PM, Modified: Jan 21, 2022 12:43 PM, Service Instance: Example_CLS, and Install Status: INSTALLED. The description is 'CLS to demonstrate a normal installation procedure.' At the bottom of the card, there are five tabs: Overview, Server Features, License Pools (which is highlighted with a red box), Fulfillment Conditions, and Leases.

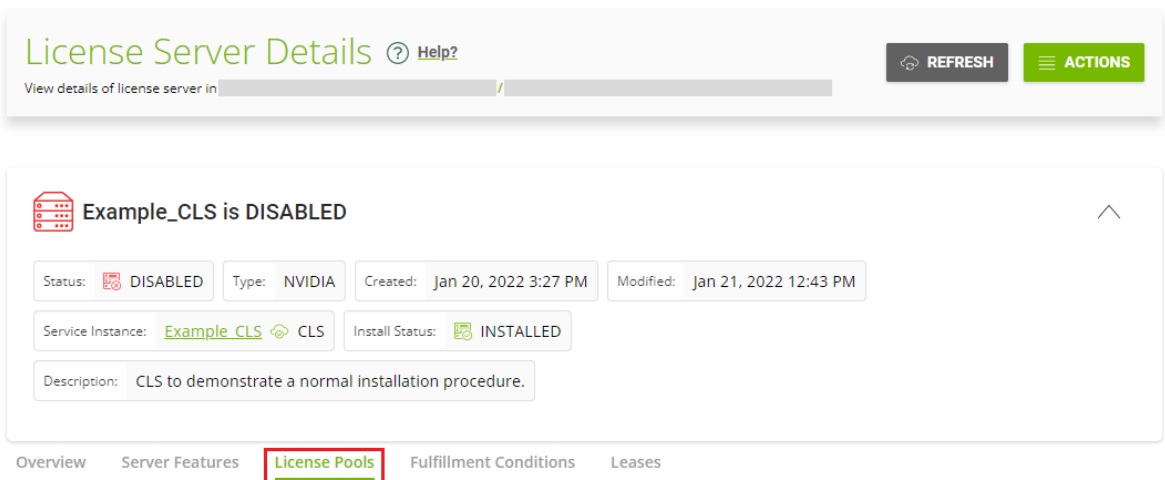
- In the list of license pools on the License Pools tab, from the **Actions** menu for the license pool, choose **Disable**.
When the license pool is disabled, licenses cannot be served to licensed clients from the pool.
- When prompted, confirm that you want to disable the license pool.
- From the **Actions** menu for the license pool, choose **Delete**.
- When asked if you want to delete the license pool, click **DELETE LICENSE POOL**.



4.4.3. Managing Licenses and Licensed Products in a License Pool

Manage licenses in a license pool if you need to add or remove licenses for a specific product in the pool. You can also add and remove licensed products from a license pool. When a licensed product is removed from a license pool, all licenses are returned to the license server.

1. Navigate to the **License Server Details** page of the license server to which the license pool belongs.
For instructions, see [Navigating to the License Server Details Page for a License Server](#).
2. In the License Server Details page that opens, click the **License Pools** tab.



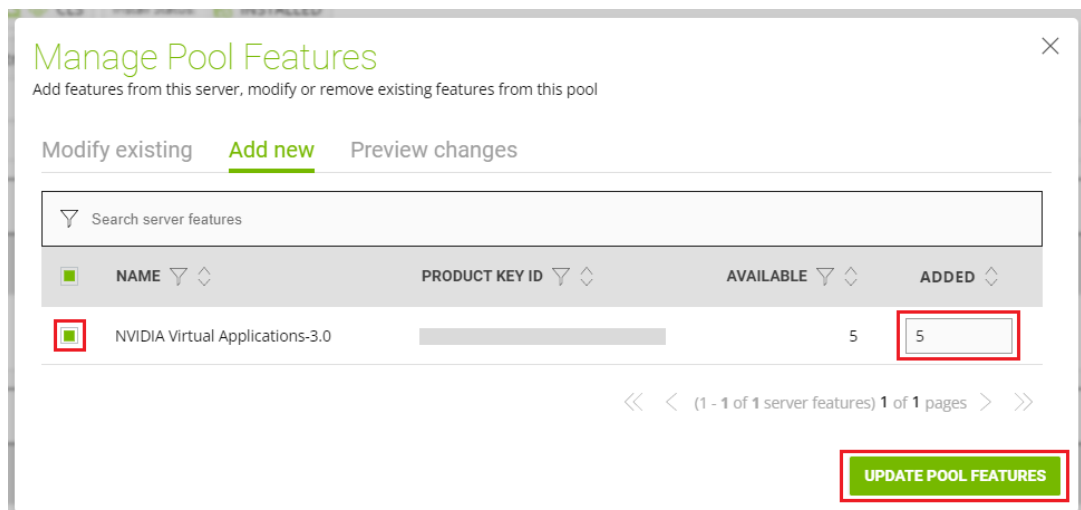
3. In the list of license pools on the License Pools tab, from the **Actions** menu for the license pool, choose **Disable**.

When the license pool is disabled, licenses cannot be served to licensed clients from the pool.

4. When prompted, confirm that you want to disable the license pool.
5. Expand the license pool that contains the licenses you want to manage.
6. From the **Actions** menu for the license pool, choose **Manage features**.
7. In the **Manage Licenses** pop-up window that opens, add or remove licenses for the licensed products that you are interested in.

Add or remove licenses for each licensed product as follows:

- a). In the text-entry field in the **ADDED** column, enter the number of licenses for the product that you want to **remain in the pool after updating licenses**.
 - ▶ To **add** licenses to the pool, enter a number **greater** than the number already in the pool, but less than or equal to the total number of licenses available on the license server.



If you enter a number greater than the total number of licenses available, an error occurs.

- ▶ To **remove** licenses from the pool, enter a number **less** than the number already allocated to the server but greater than 0.

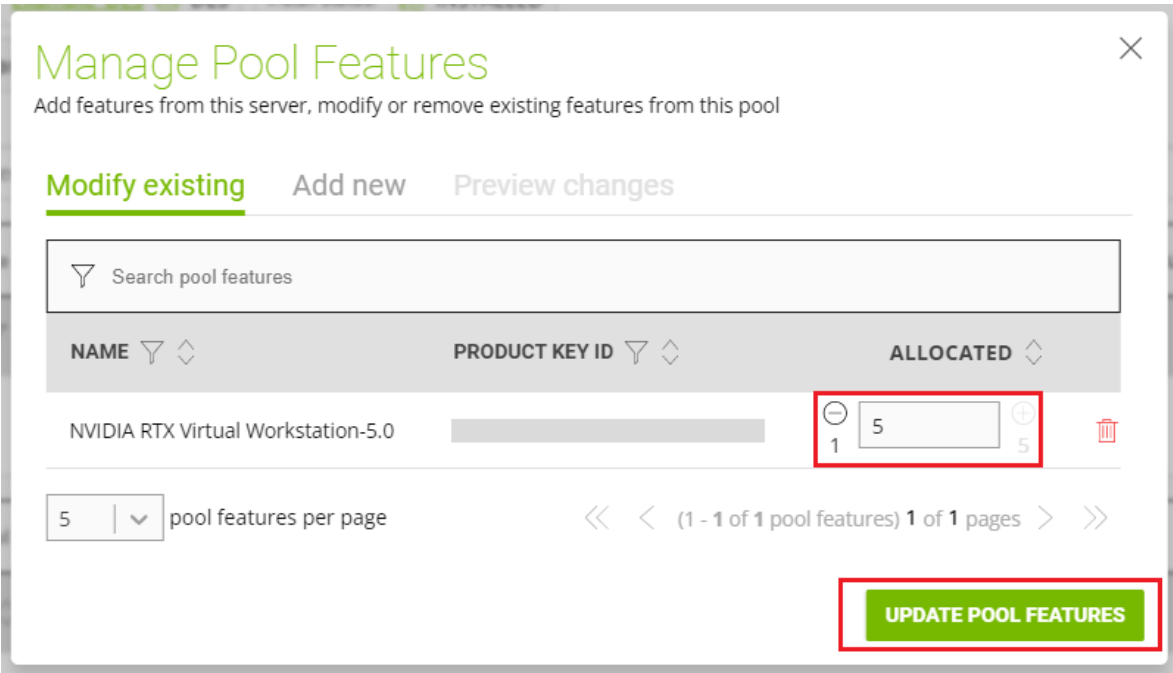
For example, to remove 4 licenses from a pool that contains 10 licenses, leaving 6 licenses on the license server, enter the **Licenses** field to **6**.

You cannot set the **Licenses** field to **0**. You must leave at least 1 license in the license pool. If you want to remove all licenses for a product from the license pool, you must remove the product from the pool by clicking the trash can icon.

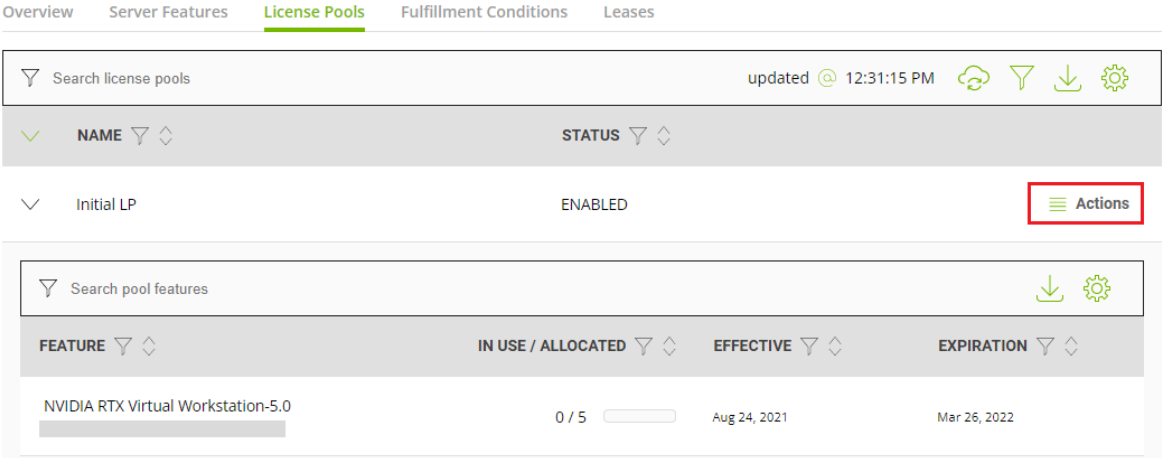
- b). Click **ADD**.

The product and number of licenses are added to the **Features to Modify** list.

8. After adding or removing all the licenses and licensed products that you are interested in, click **UPDATE POOL FEATURES**.



9. From the **Actions** menu for the license pool, choose **Enable** and, when prompted, confirm that you want to enable the license pool.



Licenses can now be served to licensed clients from the pool.

4.4.4. Merging Two License Pools

If you need to consolidate licenses in two pools into a single pool, you can merge the two pools. When you merge two license pools, all licenses in the pool that you select as the

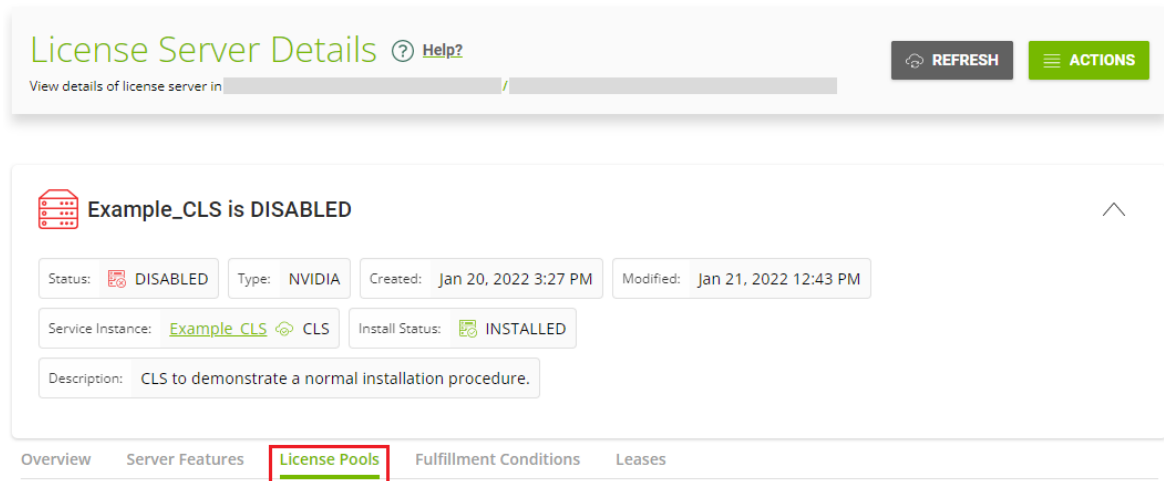
origin pool are migrated to the pool you select as the destination pool and the origin pool is deleted.

Ensure that no licenses in the origin pool are checked out by licensed clients. A license pool cannot be merged while any of the licenses in the pool are checked out by a client. If a license in the pool is checked out, the **Split or merge** command for the pool is dimmed and inactive.

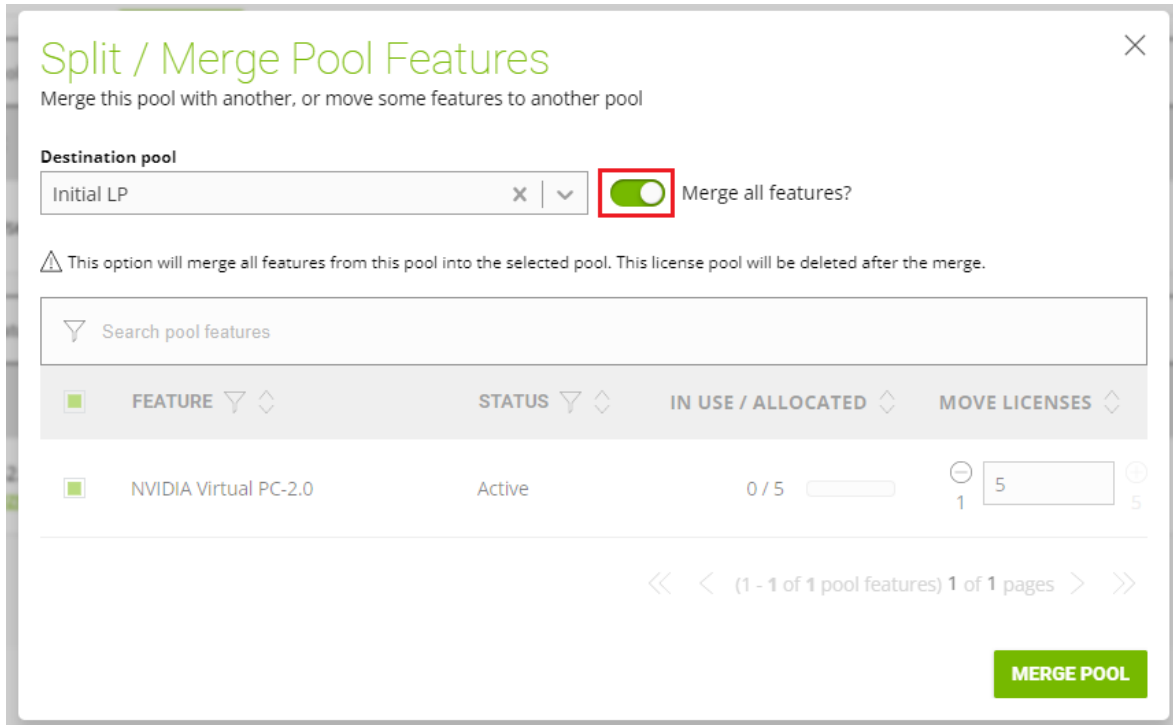
1. Navigate to the **License Server Details** page of the license server to which both the license pools belong.

For instructions, see [Navigating to the License Server Details Page for a License Server](#).

2. In the License Server Details page that opens, click the **License Pools** tab.



3. In the list of license pools on the License Pools tab, from the **Actions** menu for the license pool, choose **Disable**.
When the license pool is disabled, licenses cannot be served to licensed clients from the pool.
4. When prompted, confirm that you want to disable the license pool.
5. From the **Actions** menu for the origin license pool, choose **Split or merge**.
6. In the **Split / Merge Pool Features** pop-up window that opens, select the destination license pool, select the **Merge all features?** option and click **MERGE POOL**.



All licenses in the pool that you select as the origin pool are migrated to the pool you select as the destination pool and the origin pool is deleted.

4.4.5. Migrating Licenses Between License Pools

If demand for licenses from different pools changes, you can migrate licenses between pools to meet the changed demand.

Ensure that the following prerequisites are met:

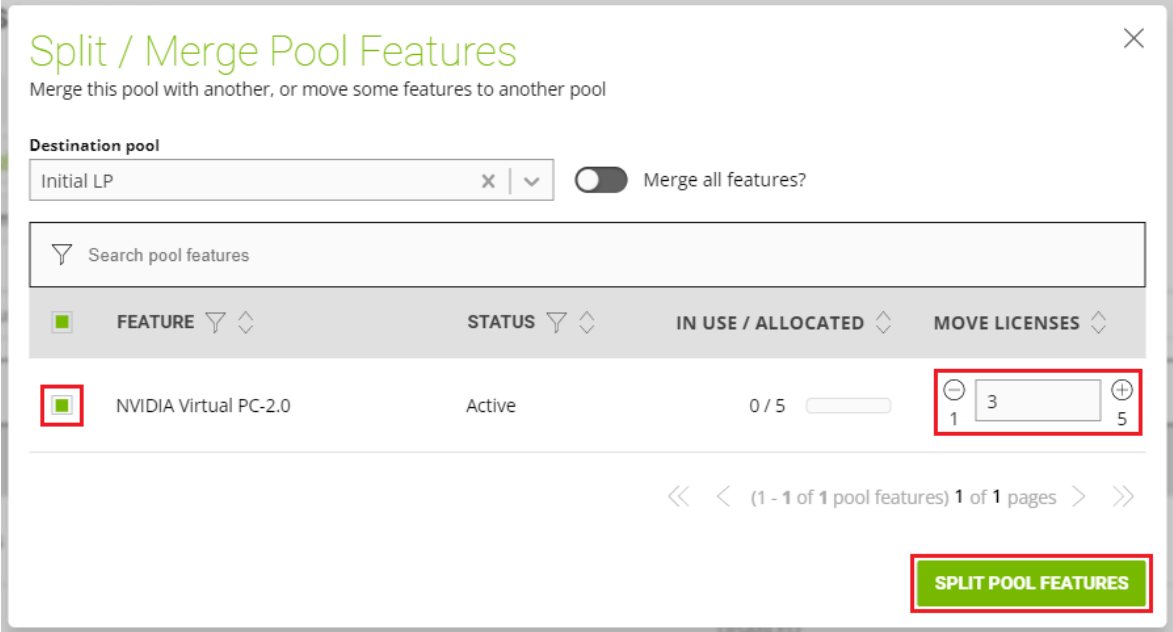
- ▶ Both pools between which you will migrate licenses already exist. If you want to migrate licenses to a new pool, create the pool first.
 - ▶ No licenses that want to migrate are checked out by licensed clients.
1. Navigate to the **License Server Details** page of the license server to which both the license pools belong.
For instructions, see [Navigating to the License Server Details Page for a License Server](#).
 2. In the License Server Details page that opens, click the **License Pools** tab.

The screenshot shows the 'License Server Details' interface. At the top, there is a header with the title 'License Server Details' and a 'Help?' link. Below the header, there is a progress bar and two buttons: 'REFRESH' and 'ACTIONS'. The main content area displays the status of a license pool: 'Example_CLS is DISABLED'. Below this, there are several metadata fields: 'Status: DISABLED', 'Type: NVIDIA', 'Created: Jan 20, 2022 3:27 PM', and 'Modified: Jan 21, 2022 12:43 PM'. Further down, there are 'Service Instance: Example_CLS' and 'Install Status: INSTALLED'. A description field contains the text: 'CLS to demonstrate a normal installation procedure.' At the bottom of the main content area, there is a navigation bar with tabs: 'Overview', 'Server Features', 'License Pools' (which is highlighted with a red box), 'Fulfillment Conditions', and 'Leases'.

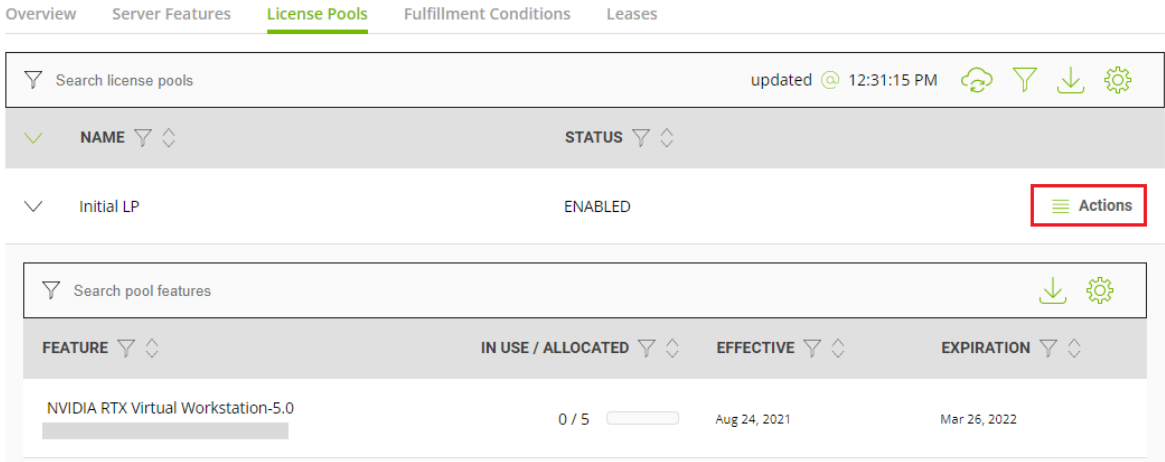
3. In the list of license pools on the License Pools tab, from the **Actions** menu for the license pool, choose **Disable**.
When the license pool is disabled, licenses cannot be served to licensed clients from the pool.
4. When prompted, confirm that you want to disable the license pool.
5. From the **Actions** menu for the license pool that you disabled, choose **Split or merge**.
6. From the **Destination Pool** drop-down list, select the license pool to which you want to migrate licenses.
7. In the **Split / Merge Pool Features** pop-up window that opens, select the licenses that you want migrate.

Select the licenses for each licensed product as follows:

- a). In the list of products, select the licensed product for which you want to migrate licenses.
 - b). In the text-entry field in the **MOVE LICENSES** column, enter the number of licenses for the product that you want to migrate.
You must leave at least one license in the license pool from which you want to migrate licenses. If you want to remove all licenses for a product from the license pool, you must follow the instructions in [Managing Licenses and Licensed Products in a License Pool](#).
 - c). Click **ADD**.
The product and number of licenses are added to the **Features to Move** list.
8. Click **SPLIT POOL FEATURES**.



9. From the **Actions** menu for the license pool, choose **Enable** and, when prompted, confirm that you want to enable the license pool.



Licenses can now be served to licensed clients from the pool.

4.5. Managing Fulfillment Conditions

A fulfillment condition selects the license pools from which a license requested by a licensed client is served. It is a test that is applied to any request from a licensed client to determine if the request may be fulfilled from a specified set of license pools.

A fulfillment condition is bound to an ordered list of license pools. If a request satisfies the conditions of the test, the bound license pools are evaluated, in order, to determine if the request can be served from the pool.

A fulfillment condition may belong to only one license server. However, a license server may contain any number of fulfillment conditions. If a license server contains more than one fulfillment condition, the conditions are ordered. Every request from a licensed client is tested against each fulfillment condition in order either until the request can be fulfilled or has been tested against all the fulfillment conditions.



Note: A license server for which the **Node-locked licensing mode?** option is set does not support multiple fulfillment conditions. Only the default fulfillment condition is available.

4.5.1. About Match Conditions

A match condition determines whether a request from a licensed client may be fulfilled from the license pools bound to a fulfillment condition. You must specify a match condition when you create or edit a fulfillment condition.

Reference Match

The Reference Match condition allows only clients that have been provisioned with the client configuration token associated with a fulfillment condition to be served. The client configuration token contains a unique identifier for the fulfillment condition. The client provides this unique identifier to the server whenever the client requests a license from the server.

For information about how to provision a licensed client with a condition match token, see:

- ▶ [Generating a Client Configuration Token](#)
- ▶ [Configuring a Licensed Client with a Networked License](#)

Universal Match

The Universal Match condition allows any client to be served. It is the default fulfillment condition and is applied if more specific conditions are not met or they were unable to fulfill a request. Because this condition is the most general condition, it is the last condition to be evaluated.

Only one fulfillment condition for a license server may specify the Universal Match condition. If another fulfillment condition for the server specifies this match condition, it is absent from the **Match Condition** drop-down list.

4.5.2. Creating a Fulfillment Condition

1. Navigate to the **License Server Details** page of the license server to which the service instance is bound.

For instructions, see [Navigating to the License Server Details Page for a License Server](#).

2. In the **License Server Details** page that opens, disable the license server by clicking **DISABLE SERVER** and, when prompted, confirm that you want to disable the license server.

When the license server is disabled, it cannot serve licenses to licensed clients.

3. From the **Actions** menu, choose **Create Condition**.

The **Create Fulfillment Condition** wizard opens.

Create Fulfillment Condition ✕

Enter a name for this new fulfillment condition, select a match condition and choose the license pools to bind

1 **Name and match condition** → 2 **Select license pools** → 3 **Preview condition creation**

i Enter a name, an optional description, and select the match condition for this fulfillment condition

Name

HighPriority

Description

Test Fulfillment Condition to demonstrate the creation process.

Match condition

Reference Match

Next: Select license pools →

4. In the **Name** field, enter your choice of name for the fulfillment condition.
5. **Optional:** In the **Description** field, enter a text description of the fulfillment condition.
6. Under **Match Condition**, select a match condition to determine which clients may be served licenses from the license pools bound to this fulfillment condition.

The following match conditions are defined:

- ▶ [Reference Match](#)
- ▶ [Universal Match](#)

7. Specify the sequence of license pools from which licenses will be served to clients.

Create Fulfillment Condition ✕

Enter a name for this new fulfillment condition, select a match condition and choose the license pools to bind

1 Name and match condition → | 2 **Select license pools** → | 3 Preview condition creation

ⓘ Select one or more license pools to bind to the new fulfillment condition

License pools

Initial LP →

Bound license pools

← 0 HighPriorityUsers ↑ ↓

← **Previous: Name and match condition** **Next: Preview condition creation** →

Licenses are served in the order in which they appear in the **Bound license pools** list.



Note: You can specify a license pool in the license pool bindings for any number of fulfillment conditions.

- a). In the **License Pools** list, select the license pools from which you want licenses to be served and click the right arrow icon.
The license pools are moved to the **Bound license pools** list.

- b). In the **Bound license pools** list, adjust the order of the license pools as necessary by selecting each license pool that you need to move and clicking the up arrow and down arrow icons to move the license pool to its required position in the sequence.
8. Click **CREATE FULFILLMENT CONDITION**.

Create Fulfillment Condition ×

Enter a name for this new fulfillment condition, select a match condition and choose the license pools to bind

Name and match
condition

→|

Select license
pools

→|

3 Preview condition
creation

i Review your selections for this fulfillment condition

Name	HighPriority
Description	Test Fulfillment Condition to demonstrate the creation process.
Match condition	Reference Match

Reminder! Feature requests will be evaluated in the order of the bound license pools

Bound pools (1)

HighPriorityUsers

CREATE FULFILLMENT CONDITION

|← **Previous: Select license pools**

The fulfillment condition is added to the list of fulfillment conditions on the **Fulfillment Conditions** tab of the License Server Details page.

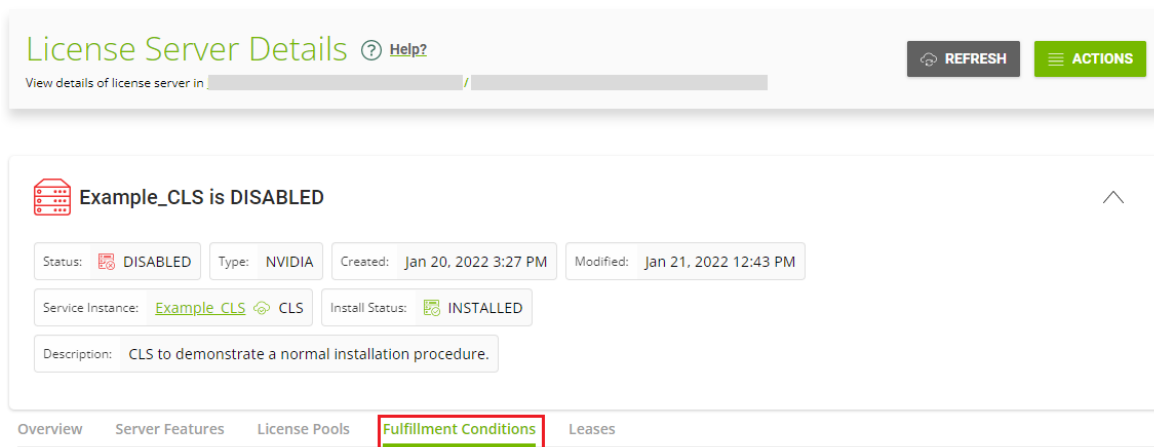
9. On the Overview tab of the License Server Details page, enable the license server by clicking **ENABLE SERVER** and, when prompted, confirm that you want to enable the license server.
- The license server can now serve licenses to licensed clients.

4.5.3. Deleting a Fulfillment Condition

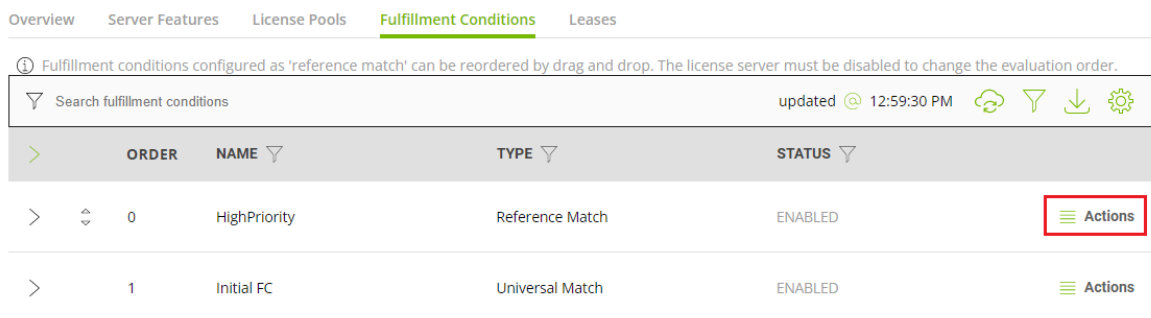
To be able to serve licenses, a license server must have at least one fulfillment condition. If you delete all the fulfillment conditions that belong to a license server, the license server is no longer able to serve licenses to clients.

Note: Deleting a fulfillment condition does not delete the license pools that are bound to the fulfillment condition. If you delete all fulfillment conditions to which a license pool is bound, licenses in the pool can no longer be served to clients.

1. Navigate to the **License Server Details** page of the license server to which the service instance is bound.
For instructions, see [Navigating to the License Server Details Page for a License Server](#).
2. On the License Server Details page that opens, click the **Fulfillment conditions** tab.



3. In the list of fulfillment conditions on the **Fulfillment conditions** tab, from the **Actions** menu for fulfillment condition that you want to delete, choose **Disable**.



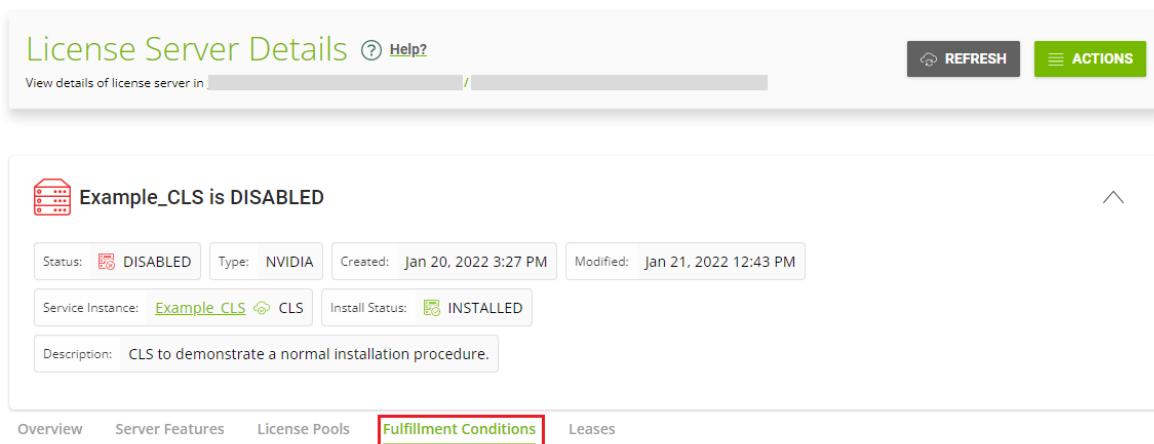
When the fulfillment condition is disabled, it cannot be used to fulfill requests for licenses from licensed clients.

4. From the **Actions** menu for the fulfillment condition that you want to delete, choose **Delete**.
5. When asked if you want to delete the fulfillment condition, click **DELETE FULFILLMENT CONDITION**.

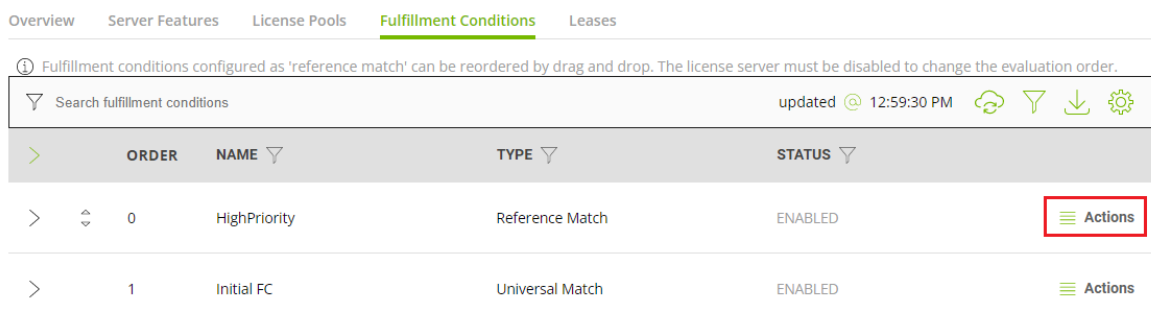
After a fulfillment condition is deleted, it is ignored in requests that specify the condition. Furthermore, if a request specifies only deleted fulfillment conditions, the request won't be satisfied.

4.5.4. Editing a Fulfillment Condition

1. Navigate to the **License Server Details** page of the license server to which the service instance is bound.
For instructions, see [Navigating to the License Server Details Page for a License Server](#).
2. On the License Server Details page that opens, click the **Fulfillment conditions** tab.



3. In the list of fulfillment conditions on the **Fulfillment conditions** tab, from the **Actions** menu for fulfillment condition that you want to delete, choose **Disable**.



When the fulfillment condition is disabled, it cannot be used to fulfill requests for licenses from licensed clients.

4. From the **Actions** menu for the fulfillment condition that you want to edit, choose **Edit**.
The **Edit Fulfillment Condition** wizard is started.
5. Use the **Edit Fulfillment Condition** wizard make the changes that you require.

- a). In the **Name** field, edit the name for the fulfillment condition.
- b). In the **Description** field, edit the description of the fulfillment condition.
- c). Under **Match Condition**, select a new match condition to determine which clients may be served licenses from the license pools bound to this fulfillment condition.



Note: If you change the match condition, you must regenerate all client configuration tokens that specified the fulfillment condition and provision all affected licensed clients with the new token.

The following match conditions are defined:

▶ [Reference Match](#)

▶ [Universal Match](#)

d). Click **Next: Select license pools**.

e). Modify the sequence of license pools from which licenses will be served to clients.

Use the left and right arrow icons to move selected license pools between the **License Pools** list and the **Bound license pools** list. Use the up and down arrow icons to adjust the order of the license pools in the **Bound license pools** list.

Licenses are served in the order in which they appear in the **Bound license pools** list.

f). Click **Preview condition update**.

6. Click **EDIT FULFILLMENT CONDITION**.

Edit Fulfillment Condition ✕

Update the name, description, match type, or bound pools of this fulfillment condition

1 Name and match condition

→

2 Select license pools

→

3 Preview condition update

(i) Review your selections for this fulfillment condition

Name	HighPriorityUpdate
Description	Test Fulfillment Condition to demonstrate the creation process.
Match condition	Reference Match

Reminder! Feature requests will be evaluated in the order of the bound license pools

Bound pools (1)

0
HighPriorityUsers

EDIT FULFILLMENT CONDITION

[← Previous: Select license pools](#)

7. From the **Actions** menu for fulfillment condition that you edited, choose **Enable**. The fulfillment condition can now be used to fulfill requests for licenses from licensed clients.

4.5.5. Changing the Order of Fulfillment Conditions

By default, fulfillment conditions that are configured with the Reference Match condition are tested in the order in which they were added to a license server. You can change this order if you want the fulfillment conditions to be tested in a specific order.

Ensure that the license server contains at least two fulfillment conditions that are configured with the Reference Match condition.

1. Navigate to the **License Server Details** page of the license server to which the service instance is bound.
For instructions, see [Navigating to the License Server Details Page for a License Server](#).

- In the **License Server Details** page that opens, disable the license server by clicking **DISABLE SERVER** and, when prompted, confirm that you want to disable the license server.

When the license server is disabled, it cannot serve licenses to licensed clients.

- Click **REINDEX**.

The **Reindex Fulfillment Conditions** window opens.

- Rearrange your fulfillment conditions in the order that you want.

Move each fulfillment condition that you want to move up or down in the processing order as follows:

- In the **Fulfillment Conditions** list, select the condition that you want to move.



Note: A fulfillment condition that is configured with the Universal Match condition is not displayed in this list because it is processed last.

- When hovering your cursor over the up and down arrows, your cursor will change into a hand. Use the hand cursor to drag the condition to the position in the processing order that you want.

ORDER	NAME	TYPE	STATUS
0	HighPriority	Reference Match	ENABLED
1	LowPriority	Reference Match	ENABLED
2	Initial FC	Universal Match	DISABLED

- After re-ordering the conditions, you will see the following alert: **You have unsaved changes in the ordering**. Click the alert to reindex the order of your fulfillment conditions.

The order in which the fulfillment conditions are listed on the **License Server Details** page is updated to match the order that you specified.

- On the Overview tab of the License Server Details page, enable the license server by clicking **ENABLE SERVER** and, when prompted, confirm that you want to enable the license server.

The license server can now serve licenses to licensed clients.

4.6. Generating a Client Configuration Token

A client configuration token identifies the service instance, license servers, and fulfillment conditions to be used to serve a license in response to a request from a licensed client. This information must be exchanged between a service instance and a licensed client to enable the service instance to serve licenses to the client.

After generating a client configuration token, you copy it to each licensed client that you want to use the token. Each client then provides data from the token back to the server whenever the client requests a license from the server.

A client configuration token is valid for 12 years after it is generated.

Create **one** client configuration token **for each** combination of license servers and fulfillment conditions that you want to use to serve licenses in response to requests from licensed clients.



Note: You **cannot** generate a client configuration token from a service instance that is bound to a license server for which the **Node-locked licensing mode?** option is set. Instead, generate a node-locked license as explained in [Generating Node-Locked Licenses](#).

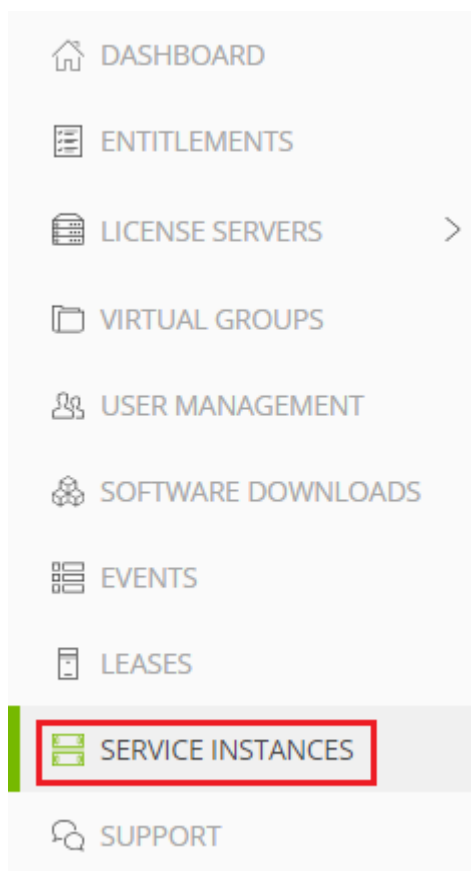
How to generate a client configuration token depends on whether you are generating the token for a CLS or a DLS instance. For detailed instructions, see:

- ▶ [Generating a Client Configuration Token for a CLS Instance](#)
- ▶ [Generating a Client Configuration Token for a DLS Instance](#)

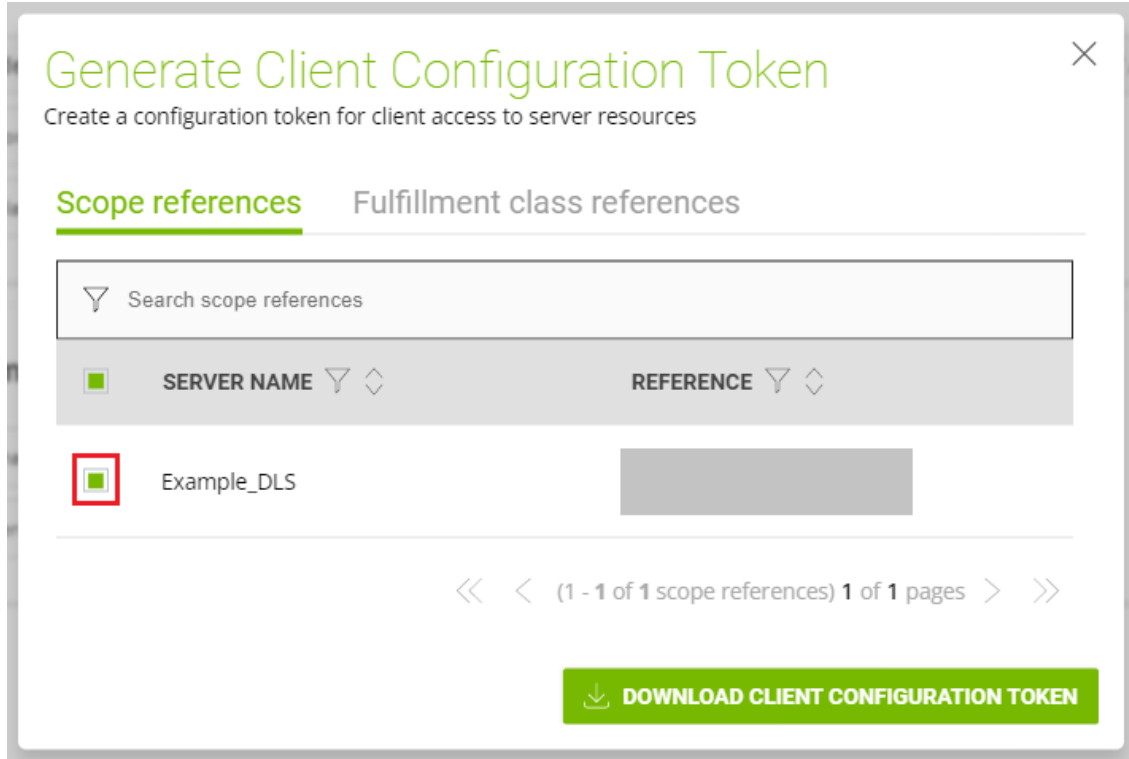
After creating a client configuration token from a service instance, copy the client configuration token to each licensed client that you want to use the combination of license servers and fulfillment conditions specified in the token. For more information, see [Configuring a Licensed Client with a Networked License](#).

4.6.1. Generating a Client Configuration Token for a CLS Instance

1. Log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
2. If your assigned roles give you access to multiple virtual groups, select the virtual group for which you are managing licenses from the list of virtual groups at the top right of the NVIDIA Licensing Portal dashboard.
3. In the left navigation pane, click **SERVICE INSTANCES**.



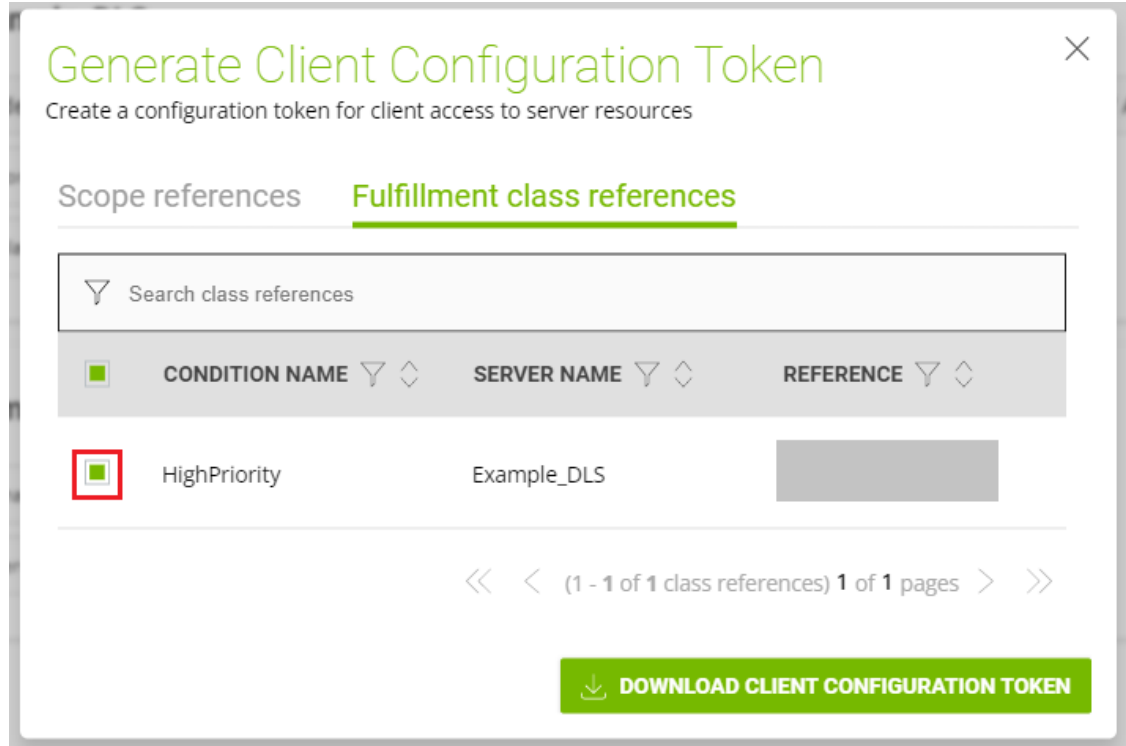
4. On the Service Instances page that opens, from the **Actions** menu for the CLS instance for which you want to generate a client configuration token, choose **Generate client configuration token**.
5. In the **Generate Client Configuration Token** pop-up window that opens, select the references that you want to include in the client configuration token.
 - a). From the list of scope references, select the scope references that you want to include.



You must select **at least one** scope reference.

Each scope reference specifies the license server that will fulfil a license request.

- b). **Optional:** Click the **Fulfillment class references** tab, and from the list of fulfillment class references, select the fulfillment class references that you want to include.



Including fulfillment class references is optional.

- c). **Optional:** In the **Expiration** section, select an expiration date for the client configuration token. If you do not select a date, the default token expiration time is 12 years.
- d). Click **DOWNLOAD CLIENT CONFIGURATION TOKEN**.

A file named `client_configuration_token_mm-dd-yyyy-hh-mm-ss.tok` is saved to your default downloads folder.

4.6.2. Generating a Client Configuration Token for a DLS Instance

1. If you are not already logged in, log in to the **NVIDIA Licensing** application at the IP address of the VM on which the DLS instance resides.
2. In the left navigation pane, click **SERVICE INSTANCE**.
3. On the **Service Instance** page that opens, from the **Actions** menu for the DLS instance for which you want to generate a client configuration token, choose **Generate client configuration token**.

Service Instance Details
Manage the DLS service instance and customize to your environment

Example_DLS

Id: [input] Type: DLS State: Registered Created: Jan 18, 2022 4:57 AM Modified: Jan 20, 2022 3:46 PM

Description: Example DLS to show features and procedures to configuring an on-premise NLS appliance.

High availability: Standalone [Configure high availability](#)

Primary Node Health

Domain name: [input] IPv4 address: [input] IPv6 address: not assigned

Critical services: Active [Restart](#) Other services: Active [Restart](#)

[CONFIGURE IP ADDRESS](#)

[ACTIONS](#)

- 4. In the **Generate Client Configuration Token** pop-up window that opens, select the references that you want to include in the client configuration token.
 - a). Click the **Scope references** tab, and from the list of scope references, select the scope references that you want to include.

Generate Client Configuration Token
Create a configuration token for client access to server resources

Scope references Fulfillment class references

Search scope references

SERVER NAME	REFERENCE
<input checked="" type="checkbox"/> Example_DLS	[input]

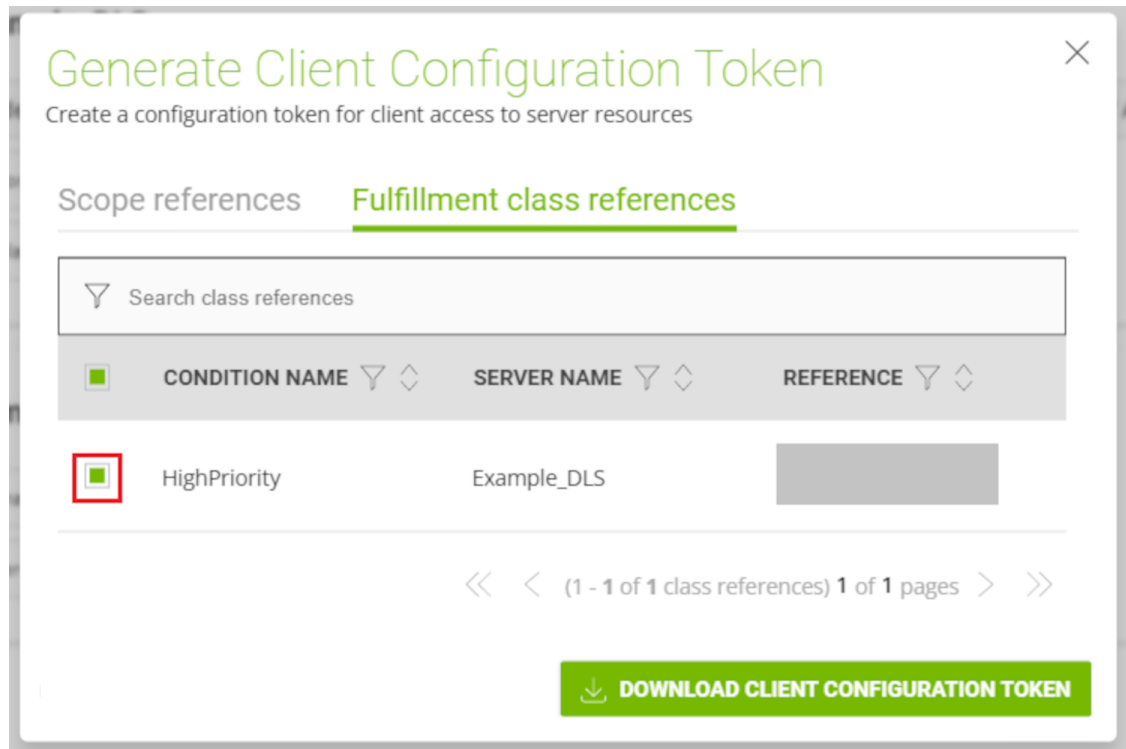
<< < (1 - 1 of 1 scope references) 1 of 1 pages > >>

[DOWNLOAD CLIENT CONFIGURATION TOKEN](#)

You must select **at least one** scope reference.

Each scope reference specifies the license server that will fulfil a license request.

- b). **Optional:** Click the **Fulfillment class references** tab, and from the list of fulfillment class references, select the fulfillment class references that you want to include.



Including fulfillment class references is optional.

- c). **Optional:** If you want the service instance, or each node in an HA cluster of instances, to be identified through its IP address, click the **Server address preferences** tab and select the address for the IP version that you want: **IPv6** or **IPv4**.

By default, a service instance, or each node in an HA cluster of instances, is identified through its fully qualified domain name.

d). **Optional:** In the **Expiration** section, select an expiration date for the client configuration token. If you do not select a date, the default token expiration time is 12 years.

e). Click **DOWNLOAD CLIENT CONFIGURATION TOKEN**.

A file named `client_configuration_token_mm-dd-yyyy-hh-mm-ss.tok` is saved to your default downloads folder.

You can decouple the leasing port from the UI port for auth and lease operations. Once you have done so, you can block the UI port for the client VM.

Note: For backward compatibility, leasing operations will still be supported by the default HTTPS port (443) in the VM version.

- ▶ All UI and leasing operations will be supported on the default HTTPS port, 443.
- ▶ Only leasing operations will be supported on the leasing port, 8082.

4.7. Generating Node-Locked Licenses

Note: Support for node-locked licensing was introduced in NVIDIA vGPU software 15.0. It is **not** supported in earlier NVIDIA vGPU software releases.

1. Navigate to the **License Server Details** page of the license server from which you want to generate node-locked licenses.

For instructions, see [Navigating to the License Server Details Page for a License Server](#).

2. In the **License Server Details** page that opens, ensure that the license server is **enabled**.
3. From the **ACTIONS** menu, choose **Generate node-locked license**.
4. In the **Generate Node-Locked License** pop-up window that opens, configure the node-locked licenses that you want to generate and click **GENERATE**.
 - a). For each client to which you want to apply the license, enter the MAC address of the client in the text entry field and click **Add MAC Address(es)**.
You can specify the MAC address in any of the following formats:
 - ▶ As six two-digit hexadecimal numbers, separated by colons, for example, **00:00:5E:00:53:10**
 - ▶ As six two-digit hexadecimal numbers, separated by hyphens, for example, **00-00-5E-00-53-10**
 - ▶ As three four-digit hexadecimal numbers, separated by periods, for example, **0000.5E00.5310**
 - b). From the **Available features** drop-down list, select each product for which you want to generate a license.



Note: The expiry date of the license generated for each product is the expiry date of the entitlement to which the product belongs. No options are provided for setting the date on which you want the node-locked license to expire.

A ZIP archive named `n11.zip` is created and downloaded to your default downloads folder. This ZIP archive contains the following files:

- ▶ One license file for each client whose MAC address you entered
 - ▶ An index file named `index.txt` to enable you identify the license file that was created for each client from its MAC address
5. When you're ready to configure your licensed clients with the node-locked licenses that you generated, extract the contents of the ZIP archive `n11.zip`.

After extracting the contents of the ZIP archive `n11.zip`, configure each licensed client for which you generated a node-locked license as explained in [Configuring a Licensed Client with an NVIDIA vGPU Software Node-Locked License](#).

4.8. Disabling and Enabling a License Server, License Pool, or Fulfillment Condition

When modifying a license server, license pool, or fulfillment condition, you must disable it before modifying it. To ensure that service instance can serve licenses to licensed clients,

you must ensure that its license servers, licence pools and fulfillment conditions are enabled.

You disable and enable a license server, license pool, or fulfillment condition from the relevant **License Server Details** page. For information about how to navigate to the **License Server Details** page, see [Navigating to the License Server Details Page for a License Server](#).

- # To disable a license server, navigate to the **Overview** tab of the License Server Details page for the license server. Then click **DISABLE SERVER** and, when prompted, confirm that you want to disable the license server.
When the license server is disabled, it cannot serve licenses to licensed clients.
- # To enable a license server, navigate to the **Overview** tab of the License Server Details page for the license server. Then click **ENABLE SERVER** and, when prompted, confirm that you want to enable the license server.
The license server can now serve licenses to licensed clients.
- # To disable a license pool, navigate to the **License Pools** tab of the License Server Details page for the license server to which the license pool belongs. Then from the **Actions** menu for the license pool, choose **Disable**.
When the license pool is disabled, licenses cannot be served to licensed clients from the pool.
- # To enable a license pool, navigate to the **License Pools** tab of the License Server Details page for the license server to which the license pool belongs. Then from the **Actions** menu for the license pool, choose **Enable**.
Licenses can now be served to licensed clients from the pool.
- # To disable a fulfillment condition, navigate to the **Fulfillment conditions** tab of the License Server Details page for the license server to which the fulfillment condition belongs. Then from the **Actions** menu for the fulfillment condition, choose **Disable**.
When the fulfillment condition is disabled, it cannot be used to fulfill requests for licenses from licensed clients.
- # To enable a fulfillment condition, navigate to the **Fulfillment conditions** tab of the License Server Details page for the license server to which the fulfillment condition belongs. Then from the **Actions** menu for the fulfillment condition, choose **Enable**.
The fulfillment condition can now be used to fulfill requests for licenses from licensed clients.

4.9. Editing License Server Settings

License server settings control how a service instance handles licenses that have been served to licensed clients. You can edit the settings for an individual license server or for all license servers that are bound to a service instance. Any setting that you edit for an individual license server overrides the setting for license servers that are bound to the service instance.



Note: These settings do **not** affect license servers that are being used for node-locked licenses, that is, license servers for which the **Node-locked licensing mode?** option is set.

1. Open the **Settings** window of the individual license server or service instance for which you want to edit license server settings.

Scope

Individual license server

Steps

- a). Navigate to the **License Server Details** page of the license server to which the service instance is bound.

For instructions, refer to [Navigating to the License Server Details Page for a License Server](#).

- b). On the **License Server Details** page that opens, disable the license server by clicking **DISABLE SERVER** and, when prompted, confirm that you want to disable the license server.

When the license server is disabled, it cannot serve licenses to licensed clients.

- c). From the **ACTIONS** menu, choose **Settings**.

The **Server Settings** pop-up window opens.

All license servers bound to a CLS instance

- a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.

The NVIDIA Licensing Portal dashboard opens.

- b). If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.

- c). In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **SERVICE INSTANCES**.

- d). In the list of service instances on the **Service Instances** page that

Scope**All license servers bound to a DLS instance****Steps**

opens, from the **Actions** menu for the service instance, choose **Settings**.

e). In the **Service Instance Settings** pop-up window that opens, ensure that the **License Server** tab is selected.

a). If you are not already logged in, log in to the **NVIDIA Licensing** application on the VM or container in which the DLS appliance is installed or deployed.

b). In the left navigation pane, click **SETTINGS**.

c). On the **Service Instance Settings** page that opens, expand the **Basic Settings** section.

d). In the **Basic Settings** section, ensure that the **Lease** tab is selected.

2. Edit the settings that you want to change and click **SAVE SETTINGS**.

a). View the **Feature Overage** setting.

Maximum Allowed Feature Overage

This setting cannot be changed from its preset value and is displayed for information only. For counted licenses, it is the maximum percentage of the number of licenses available that can be leased to license clients for a limited period of time when all available licenses are checked out. For example, if you have 100 concurrent user licenses and all licenses are checked out, up to an additional 10 licenses can be leased to license clients for a limited period of time. During periods when overage allowances are being used, administrative warnings may be generated.

b). In the left navigation pane, click **Lease Duration Settings** and edit the following settings.

The effect of these settings depends on whether the request for a license from a client specifies how long the license shall remain valid. A request from a client specifies how long the license shall remain valid only if the `LicenseInterval` Windows registry value or Linux configuration parameter is set. For more information, refer to [Virtual GPU Client Licensing User Guide](#).

Lease Duration

The period of time for which a license remains valid after the license has been served if this period is not specified in the request for a license. At the end of the lease period, the license becomes invalid at the client and becomes available to be served to other licensed clients.

This setting applies only if the request for a license from the client does not specify how long the license shall remain valid. Otherwise, this setting is ignored.

Maximum Lease Duration

The maximum period of time for which a license remains valid at a licensed client after the license has been served to the client.

If the request for a license from the client specifies that the license shall remain valid for a longer period, this setting overrides the period specified in the request.

Minimum Lease Duration

The minimum period of time for which a license remains valid at a licensed client after the license has been served to the client.

If the request for a license from the client specifies that the license shall remain valid for a shorter period, this setting overrides the period specified in the request.

- c). In the left navigation pane, click **Other Settings** and edit the following settings.

Default Renewal Period

The percentage of the lease period that must elapse before a licensed client can renew a license. By renewing a license before the lease period has elapsed, a licensed client can extend its license beyond the original expiration time of the license. Extending a license ensures that if a licensed client temporarily loses network connectivity to the licensing service, there is enough time for connectivity to be restored before the license expires.

For example, if the lease period is one day and the renewal period is 20%, the client attempts to renew its license every 4.8 hours. If network connectivity is lost, the loss of connectivity is detected during license renewal and the client has 19.2 hours in which to re-establish connectivity before its license expires.

Offline Lease

Enable or disable offline lease of licenses to clients. When offline lease of licenses to clients is enabled, clients can keep their licenses even when powered off.

3. **Individual license server only:** On the **License Server Details** page, enable the license server by clicking **ENABLE SERVER** and, when prompted, confirm that you want to enable the license server.
The license server can now serve licenses to licensed clients.

4.10. Manually Releasing Leases from a Server

This section will describe options to manually release licenses using the License Server GUI if immediate license freeing is needed.

In the example where a License Client VM has been un-gracefully stopped and deleted from existence, the license will remain in-use on the server and will not be freed **until the**

lease has reached expiration. Because of this, manual admin release from the server is useful and these steps will describe the procedure.

4.10.1. Manual Release of Specific Clients Licensed to an NLS Service Instance

This section will describe how to locate and manually release specific VMs from the server.



Note:

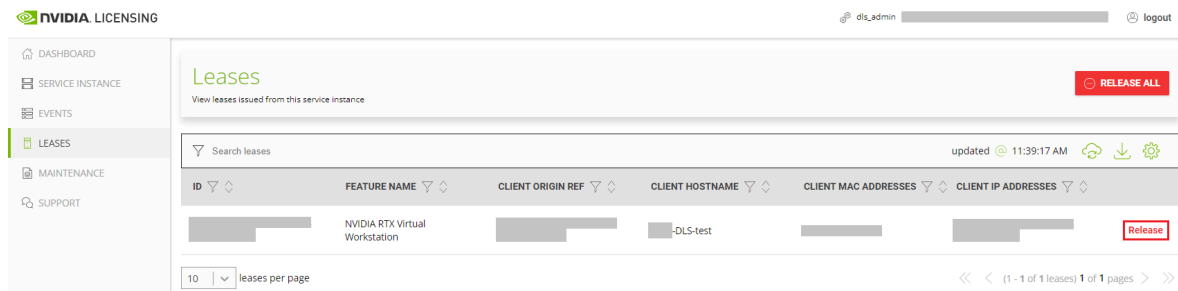
There is a daily 10% rolling limit of client VMs that can be released manually. This 10% is based on total allocated license amount onto the server. (Example: 100 licenses allocated to the server, 10 Leases can be specifically manually released).

For DLS:

1. Navigate to the DLS GUI and login with `dls_admin`.
2. Navigate to the **Leases** tab from the left navigation pane.

Optional: Use the Search Bar to search for specific License Clients to release for. To filter, you can search by: ID, Feature Name, Client Origin Ref, Client Hostname, Client MAC Addresses, or Client IP Addresses.

3. Click the red **Release** button to manually release the Virtual Machine.



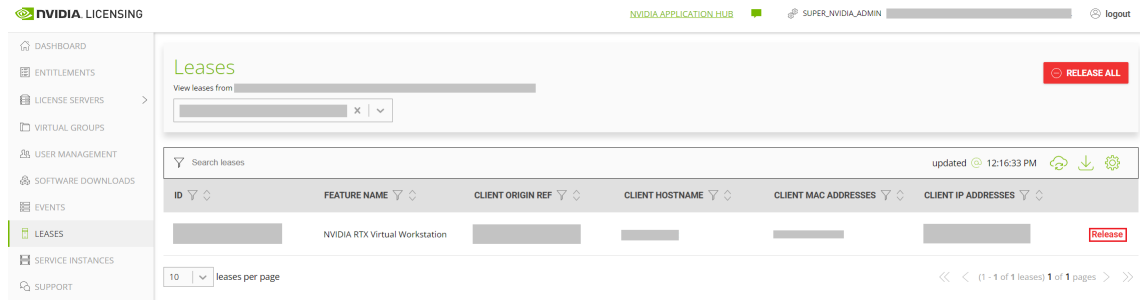
4. Click **Force Release** on the dialog that pops up.

For CLS:

1. Navigate to the NVIDIA Licensing Portal and login.
2. Navigate to the **Leases** tab from the left navigation pane.
3. From the drop-down in the header, select the Service Instance from which held leases will be force released.

Optional: Use the Search Bar to search for specific License Clients to release for. To filter, you can search by: ID, Feature Name, Client Origin Ref, Client Hostname, Client MAC Addresses, or Client IP Addresses.

4. Click the red **Release** button to manually release the Virtual Machine.



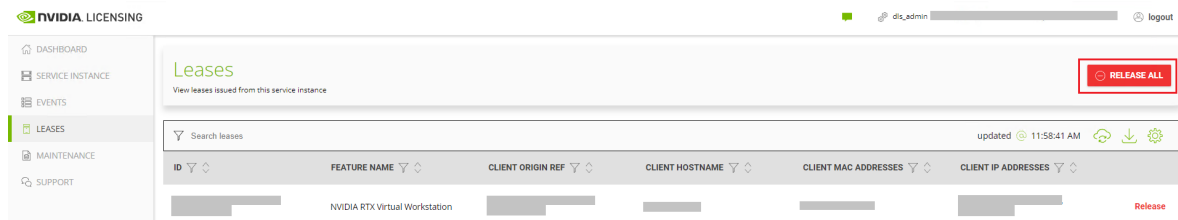
5. Click **Force Release** on the dialog that pops up.

4.10.2. Manual Force Bulk Release of All Clients Licensed to an NLS Service Instance

This section will describe how to locate and force release all leases that are in-use on a given NVIDIA License System Service Instance.

For DLS:

1. Navigate to the DLS GUI and login with `dls_admin`.
2. Navigate to the **Leases** tab from the left navigation pane.
3. Click the red **Release All** button in the top-right corner.



4. Click **Force Release** on the dialog that pops up.

For CLS:

1. Navigate to the NVIDIA Licensing Portal and login.
2. Navigate to the **Leases** tab from the left navigation pane.

3. From the drop-down in the header, select the Service Instance from which held leases will be force released.
4. Click the red **Release All** button in the top-right corner.
5. Click **Force Release** on the dialog that pops up.

Reset the Bulk Release Limit

A total of 5 licenses can be bulk released from the server. Once this limit has been reached, users can install the reset bulk release limit token.

For DLS:

1. Navigate to the NVIDIA Licensing Portal and login.
2. Click on the **Service Instance** page.
3. From the **Actions** menu on right, select **Download Force Release All Token**.
4. Once the token is downloaded, it must be uploaded on the DLS VM.
5. If there are active leases and the user has exhausted the bulk force release limit on DLS, an option will be shown to upload this token to reset the limit.

For CLS:

1. Navigate to the NVIDIA Licensing Portal and login.
2. Click on the **Service Instance** page.
3. From the **Actions** menu on right, select **Reset Force Release All Limit**. This will reset the limit back to the default value of 5.
4. If the user performs this action before exhausting the current limit, the operation will not be allowed to execute.

4.10.3. Forcibly Releasing a Node-Locked License

If a client that is licensed with a node-locked license fails or is shut down abruptly, the license remains in use and is not released **until it expires**. If you can't release the license by restarting and gracefully shutting down the client, you can use the license server to which the license is allotted to forcibly release it.



Note: NVIDIA License System does not limit the number of node-locked license that you can forcibly release in a given period.

1. Navigate to the **License Server Details** page of the license server to which the license is allotted.
For instructions, see [Navigating to the License Server Details Page for a License Server](#).
2. In the **License Server Details** page that opens, click the **Leases** tab.
3. In the list of lease files on the **Leases** tab, follow the **Release** link for the license that you want to release.

4. When asked if you want to forcibly release the license, click **FORCE RELEASE**.

4.11. Supporting Non-Persistent Desktop Pools

Non-persistent desktop pools are used by VM instances with reused MAC addresses, such as VM instances created by using Citrix Machine Creation Services (MCS) or VMware Instant Clone technology. If you plan to serve licenses to non-persistent desktop pools from a service instance, prevent the service instance from serving multiple licenses to clients that have the same MAC address.

When a service instance supports non-persistent desktop pools, it checks the MAC address of the client in each request to check out a license. If the request specifies the MAC address of a client that has already checked out a license, the existing checkout is cleared before the request is fulfilled.



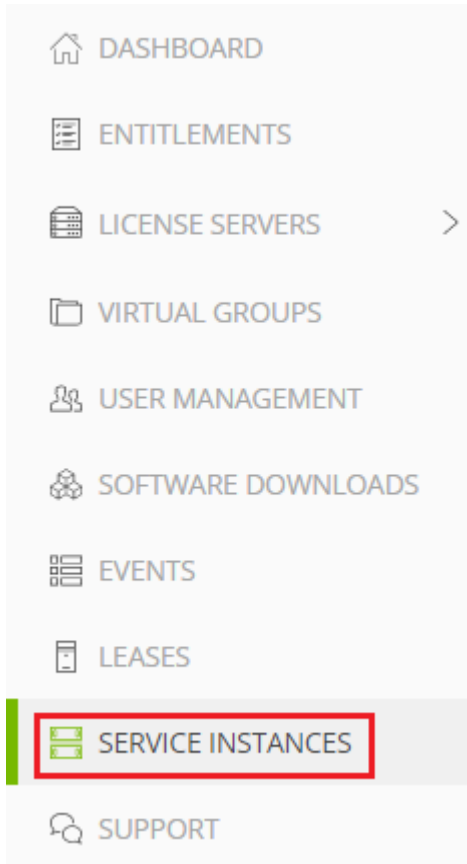
Note: You **cannot** enable support for non-persistent desktop pools for a service instance that is bound to a license server for which the **Node-locked licensing mode?** option is set.

How to support non-persistent desktop pools depends on whether you want to support them from a CLS or a DLS instance. For detailed instructions, see:

- ▶ [Supporting Non-Persistent Desktop Pools from a CLS Instance](#)
- ▶ [Supporting Non-Persistent Desktop Pools from a DLS Instance](#)

4.11.1. Supporting Non-Persistent Desktop Pools from a CLS Instance

1. Log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
2. If your assigned roles give you access to multiple virtual groups, select the virtual group for which you are managing licenses from the list of virtual groups at the top right of the NVIDIA Licensing Portal dashboard.
3. In the left navigation pane, click **SERVICE INSTANCES**.



4. On the **Service Instances** page that opens, from the **Actions** menu for the CLS instance, choose **Toggle Non-Persistent Desktop Pools**.
5. When asked if you want to support non-persistent desktop pools, click **ENABLE NON-PERSISTENT DESKTOP POOLS**.
6. From the **Toggle Non-Persistent Desktop Pools** menu, select one of the following options to detect address conflicts and avoid duplicate leases based on your setup:
 - ▶ **Hostname:** The host name
 - ▶ **Mac Address:** The MAC address of the client
 - ▶ **Hostname/Mac Address:** The host name or the MAC address of the client



Note: The status of non-persistent desktop pools is displayed on the **Service Instances** page and the **Server Details** page.

4.11.2. Supporting Non-Persistent Desktop Pools from a DLS Instance

1. If you are not already logged in, log in to the **NVIDIA Licensing** application at the IP address of the VM on which the DLS instance resides.

2. In the left navigation pane, click **SERVICE INSTANCE**.
3. On the **Service Instance** page that opens, from the **ACTIONS** menu, choose **Toggle Non-Persistent Desktop Pools**.
4. When asked if you want to support non-persistent desktop pools, click **ENABLE NON-PERSISTENT DESKTOP POOLS**.
5. From the **Toggle Non-Persistent Desktop Pools** menu, select one of the following options to detect address conflicts and avoid duplicate leases based on your setup:
 - ▶ **Hostname:** The host name
 - ▶ **Mac Address:** The MAC address of the client
 - ▶ **Hostname/Mac Address:** The host name or the MAC address of the client



Note: The status of non-persistent desktop pools is displayed on the **Service Instances** page and the **Server Details** page.

Chapter 5. Configuring a Licensed Client of NVIDIA License System

To use a licensed product, each NVIDIA License System client system must be able to obtain a license for the product either from a service instance of NVIDIA License System or from a license file installed locally on the client system. A client system can be a VM that is configured with NVIDIA vGPU, a VM that is configured for GPU pass through, or a physical host to which a physical GPU is assigned in a bare-metal deployment.

5.1. Configuring a Licensed Client with a Networked License

A client with a network connection obtains a license by leasing it from a NVIDIA License System service instance. The service instance serves the license to the client over the network from a pool of floating licenses obtained from the NVIDIA Licensing Portal. The license is returned to the service instance when the licensed client no longer requires the license.



Note: NVIDIA vGPU software releases earlier than 13.0 do **not** support NVIDIA License System. For full details of NVIDIA vGPU software releases that support NVIDIA License System, refer to [NVIDIA License System Release Notes](#).

Before configuring a licensed client, ensure that the following prerequisites are met:

- ▶ The NVIDIA vGPU software graphics driver is installed on the client.
- ▶ The client configuration token that you want to deploy on the client has been created from the NVIDIA Licensing Portal or the DLS as explained in [Generating a Client Configuration Token](#).

- ▶ Ports 443 and 80 in your firewall or proxy must be open to allow HTTPS traffic between a service instance and its the licensed clients. These ports must be open for both CLS instances and DLS instances.



Note: For DLS releases **before** DLS 1.1, ports 8081 and 8082 were also required to be open to allow HTTPS traffic between a DLS instance and its licensed clients. Although these ports are no longer required, they remain supported for backward compatibility.

The NVIDIA vGPU software graphics driver creates a default location in which to store the client configuration token on the client. If you want to use this location for the client configuration token and, on Windows, are configuring the client with NVIDIA vGPU, you can configure the client with default settings. Otherwise, you must configure the client with custom settings as explained in [Configuring a Licensed Client with a Networked License with Custom Settings](#).

The process for configuring a licensed client is the same for CLS and DLS instances but depends on the OS that is running on the client.

5.1.1. Configuring a Licensed Client with a Networked License on Windows with Default Settings

Perform this task from the client.

1. Copy the client configuration token to the %SystemDrive%:\Program Files\NVIDIA Corporation\vGPU Licensing\ClientConfigToken folder.
2. Restart the NvDisplayContainer service.

The NVIDIA service on the client should now automatically obtain a license from the CLS or DLS instance.

5.1.2. Configuring a Licensed Client with a Networked License on Linux with Default Settings

Perform this task from the client.

1. As root, open the file /etc/nvidia/gridd.conf in a plain-text editor, such as vi.

```
$ sudo vi /etc/nvidia/gridd.conf
```



Note: You can create the /etc/nvidia/gridd.conf file by copying the supplied template file /etc/nvidia/gridd.conf.template.

2. Add the FeatureType configuration parameter to the file /etc/nvidia/gridd.conf on a new line as FeatureType="value".

value depends on the type of the GPU assigned to the licensed client that you are configuring.

GPU Type	Value
NVIDIA vGPU	1. NVIDIA vGPU software automatically selects the correct type of license based on the vGPU type.
Physical GPU	The feature type of a GPU in pass-through mode or a bare-metal deployment: <ul style="list-style-type: none"> ▶ 0: NVIDIA Virtual Applications ▶ 2: NVIDIA RTX Virtual Workstation ▶ 4: NVIDIA Virtual Compute Server

This example shows how to configure a licensed Linux client for NVIDIA RTX Virtual Workstation.

```
# /etc/nvidia/gridd.conf.template - Configuration file for NVIDIA Grid Daemon
...
# Description: Set Feature to be enabled
# Data type: integer
# Possible values:
# 0 => for unlicensed state
# 1 => for NVIDIA vGPU
# 2 => for NVIDIA RTX Virtual Workstation
# 4 => for NVIDIA Virtual Compute Server
FeatureType=2
...
```

3. Copy the client configuration token to the `/etc/nvidia/ClientConfigToken` directory.
4. Ensure that the file access modes of the client configuration token allow the owner to read, write, and execute the token, and the group and others only to read the token.
 - a). Determine the current file access modes of the client configuration token.

```
# ls -l client-configuration-token-directory
```

- b). If necessary, change the mode of the client configuration token to 744.

```
# chmod 744 client-configuration-token-directory/client_configuration_token_*.tok
```

client-configuration-token-directory

The directory to which you copied the client configuration token in the previous step.

5. Save your changes to the `/etc/nvidia/gridd.conf` file and close the file.
6. Restart the `nvidia-gridd` service.

The NVIDIA service on the client should now automatically obtain a license from the CLS or DLS instance.

5.1.3. Configuring a Licensed Client with a Networked License with Custom Settings

NVIDIA License System provides custom settings for the following configuration properties of a licensed client:

- ▶ The feature type of a physical GPU
- ▶ The directory in which to store the client configuration token
- ▶ Details for a proxy server between a licensed client and a CLS instance

If you want to use the default directory in which to store the client configuration token and, on Windows, are configuring the client with NVIDIA vGPU, follow the instructions for a simplified configuration in [Configuring a Licensed Client with a Networked License](#).

5.1.3.1. Configuring a Licensed Client with a Networked License on Windows with Custom Settings

Perform this task from the client.

1. **Physical GPUs only:** Add the `FeatureType` DWord (REG_DWORD) registry value to the Windows registry key `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global\GridLicensing`.



Note:

- ▶ If you're licensing an NVIDIA vGPU, the `FeatureType` DWord (REG_DWORD) registry value is **not** required. NVIDIA vGPU software automatically selects the correct type of license based on the vGPU type.
- ▶ If you are upgrading an existing driver, this value is already set.
- ▶ For NVIDIA vGPU software releases before 15.0, add the registry value to the Windows registry key `HKEY_LOCAL_MACHINE\SOFTWARE\NVIDIA Corporation\Global\GridLicensing`.

Set this value to the feature type of a GPU in pass-through mode or a bare-metal deployment:

- ▶ 0: NVIDIA Virtual Applications
 - ▶ 2: NVIDIA RTX Virtual Workstation
 - ▶ 4: NVIDIA Virtual Compute Server
2. **Optional:** If you want store the client configuration token in a custom location, add the `ClientConfigTokenPath` String (REG_SZ) registry value to the Windows registry key `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global\GridLicensing`.



Note: For NVIDIA vGPU software releases before 15.0, add the registry value to the Windows registry key `HKEY_LOCAL_MACHINE\SOFTWARE\NVIDIA Corporation\Global\GridLicensing`.

Set the value to the full path to the folder in which you want to store the client configuration token for the client. You can use the syntax `\\fully-qualified-domain-name\share-name` for the path to the folder. By default, the client searches for the client configuration token in the `%SystemDrive%\Program Files\NVIDIA Corporation\vGPU Licensing\ClientConfigToken` folder.

By specifying a shared network drive mapped on the client, you can simplify the deployment of the same client configuration token on multiple clients. Instead of copying the client configuration token to each client individually, you can keep only one copy in the shared network drive.

3. If you are storing the client configuration token in a custom location, create the folder in which you want to store the client configuration token.

If the folder is a shared network drive, ensure that the following conditions are met:

- ▶ The folder is mapped locally on the client to the path specified in the `ClientConfigTokenPath` registry value.
- ▶ The `COMPUTER` object has the rights to access the folder on the shared network drive. The `COMPUTER` object requires these rights because the license service runs before any user logs in.

If you are storing the client configuration token in the default location, omit this step. The default folder in which the client configuration token is stored is created automatically after the NVIDIA vGPU software graphics driver is installed.

4. Copy the client configuration token to the folder in which you want to store the client configuration token.

Ensure that this folder contains only the client configuration token that you want to deploy on the client and no other files or folders. If the folder contains more than one client configuration token, the client uses the newest client configuration token in the folder.

- ▶ If you want to store the client configuration token in the default location, copy the client configuration token to the `%SystemDrive%\Program Files\NVIDIA Corporation\vGPU Licensing\ClientConfigToken` folder.
- ▶ If you want to store the client configuration token in a custom location, copy the token to the folder that you created in the previous step.

5. **Optional:** If you want the licensed client to check out a license when a user logs in to the client, add the `EnableLicenseOnLogin` String (REG_SZ) registry value to the Windows registry key `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global\GridLicensing` and set this registry value to 1.

By default, a licensed client checks out a license when the client is booted.

6. If a non-transparent proxy server is configured between your licensed client and a CLS instance, provide the information about the proxy server that the licensed client requires.



Note: Authenticated non-transparent proxy servers are **not** supported before NVIDIA vGPU software release 15.2.

Provide this information by adding the following registry values to the Windows registry key `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global\GridLicensing`.



Note: For NVIDIA vGPU software releases before 15.0, add the registry value to the Windows registry key `HKEY_LOCAL_MACHINE\SOFTWARE\NVIDIA Corporation\Global\GridLicensing`.

- a). For all non-transparent proxy servers, provide the address and port number of the proxy server in the following Windows registry values.

ProxyServerAddress String (REG_SZ)

The address of the proxy server. The address can be a fully qualified domain name such as `iproxy1.example.com`, or an IP address such as `10.31.20.45`.

ProxyServerPort String (REG_SZ)

The port number of the proxy server.

- b). If necessary, provide the credentials of the user that will log in to the proxy server.

This information is required for proxy servers that use the following authentication methods:

- ▶ Basic authentication
- ▶ Microsoft Windows Challenge/Response (Microsoft NTLM) authentication for a client that is not a member of an Active Directory domain

ProxyUserName String (REG_SZ)

The username of the user that will log in to the proxy server.

ProxyCredentialsFilePath String (REG_SZ)

The full path to the file that contains the encrypted credentials of the user that will log in to the proxy server, for example, `C:\Program Files\NVIDIA Corporation\vGPU Licensing\ProxySettings\proxy-credentials.dat`.

This file is generated as explained in [Generating an Encrypted Credentials File](#).

7. Restart the `NvDisplayContainer` service.

The NVIDIA service on the client should now automatically obtain a license from the CLS or DLS instance.

5.1.3.2. Configuring a Licensed Client with a Networked License on Linux with Custom Settings

Perform this task from the client.

1. As root, open the file `/etc/nvidia/gridd.conf` in a plain-text editor, such as `vi`.

```
$ sudo vi /etc/nvidia/gridd.conf
```



Note: You can create the `/etc/nvidia/gridd.conf` file by copying the supplied template file `/etc/nvidia/gridd.conf.template`.

2. Add the `FeatureType` configuration parameter to the file `/etc/nvidia/gridd.conf` on a new line as `FeatureType="value"`.

`value` depends on the type of the GPU assigned to the licensed client that you are configuring.

GPU Type	Value
NVIDIA vGPU	1. NVIDIA vGPU software automatically selects the correct type of license based on the vGPU type.
Physical GPU	The feature type of a GPU in pass-through mode or a bare-metal deployment: <ul style="list-style-type: none"> ▶ 0: NVIDIA Virtual Applications ▶ 2: NVIDIA RTX Virtual Workstation ▶ 4: NVIDIA Virtual Compute Server

This example shows how to configure a licensed Linux client for NVIDIA RTX Virtual Workstation.

```
# /etc/nvidia/gridd.conf.template - Configuration file for NVIDIA Grid Daemon
...
# Description: Set Feature to be enabled
# Data type: integer
# Possible values:
# 0 => for unlicensed state
# 1 => for NVIDIA vGPU
# 2 => for NVIDIA RTX Virtual Workstation
# 4 => for NVIDIA Virtual Compute Server
FeatureType=2
...
```

3. **Optional:** If you want store the client configuration token in a custom location, add the `ClientConfigTokenPath` configuration parameter to the file `/etc/nvidia/gridd.conf` on a new line as `ClientConfigTokenPath="path"`

path

The full path to the directory in which you want to store the client configuration token for the client. By default, the client searches for the client configuration token in the `/etc/nvidia/ClientConfigToken/` directory.

By specifying a shared network directory that is mounted locally on the client, you can simplify the deployment of the same client configuration token on multiple clients. Instead of copying the client configuration token to each client individually, you can keep only one copy in the shared network directory.

This example shows how to configure a licensed Linux client to search for the client configuration token in the `/mnt/nvidia/ClientConfigToken/` directory. This directory is a mount point on the client for a shared network directory.

```
# /etc/nvidia/gridd.conf.template - Configuration file for NVIDIA Grid Daemon
...
ClientConfigTokenPath=/mnt/nvidia/ClientConfigToken/
...
```

4. If you are storing the client configuration token in a custom location, create the directory in which you want to store the client configuration token.

If the directory is a shared network directory, ensure that it is mounted locally on the client at the path specified in the `ClientConfigTokenPath` configuration parameter.

If you are storing the client configuration token in the default location, omit this step. The default directory in which the client configuration token is stored is created automatically after the NVIDIA vGPU software graphics driver is installed.

5. Copy the client configuration token to the directory in which you want to store the client configuration token.

Ensure that this directory contains only the client configuration token that you want to deploy on the client and no other files or directories. If the directory contains more than one client configuration token, the client uses the newest client configuration token in the directory.

- ▶ If you want to store the client configuration token in the default location, copy the client configuration token to the `/etc/nvidia/ClientConfigToken` directory.
- ▶ If you want to store the client configuration token in a custom location, copy the token to the directory that you created in the previous step.

6. Ensure that the file access modes of the client configuration token allow the owner to read, write, and execute the token, and the group and others only to read the token.

- a). Determine the current file access modes of the client configuration token.

```
# ls -l client-configuration-token-directory
```

- b). If necessary, change the mode of the client configuration token to 744.

```
# chmod 744 client-configuration-token-directory/client_configuration_token_*.tok
```

client-configuration-token-directory

The directory to which you copied the client configuration token in the previous step.

7. **Optional:** If you want the licensed client to check out a license when a user logs in to the client, add the `EnableLicenseOnLogin` configuration parameter to the file `/etc/nvidia/gridd.conf` on a new line as `EnableLicenseOnLogin=1`.

By default, a licensed client checks out a license when the client is booted.

8. If a non-transparent proxy server is configured between your licensed client and a CLS instance, provide the information about the proxy server that the licensed client requires.



Note: Authenticated non-transparent proxy servers are **not** supported before NVIDIA vGPU software release 15.2.

- a). For all non-transparent proxy servers, provide the address and port number of the proxy server.

Provide this information by adding the following configuration parameters to the file `/etc/nvidia/gridd.conf` on separate lines.

```
ProxyServerAddress=address
ProxyServerPort=port
```

address

The address of the proxy server. The address can be a fully qualified domain name such as `iproxy1.example.com`, or an IP address such as `10.31.20.45`.

port

The port number of the proxy server.

This example sets the address of a proxy server to 10.31.20.45 and the port number to 3128.

```
# /etc/nvidia/gridd.conf.template - Configuration file for NVIDIA Grid Daemon
...
ProxyServerAddress=10.31.20.45
ProxyServerPort=3128
...
```

- b). If necessary, provide the credentials of the user that will log in to the proxy server.

This information is required for proxy servers that use basic authentication.

Provide this information by adding the following configuration parameters to the file `/etc/nvidia/gridd.conf` on separate lines.

```
ProxyUserName=domain\username
ProxyCredentialsFilePath=path
```

domain

The domain to which the user belongs, for example, `example.com`.

username

The username of the user that will log in to the proxy server, for example, `clsuser`.

path

The full path to the file that contains the encrypted credentials of the user that will log in to the proxy server, for example, `/etc/nvidia/proxy-credentials.dat`.

This file is generated as explained in [Generating an Encrypted Credentials File](#).

This example sets the domain and username of the user that will log in to the proxy server to `example.com\clsuser` and the path to the file that contains the encrypted credentials of the user to `/etc/nvidia/proxy-credentials.dat`.

```
# /etc/nvidia/gridd.conf.template - Configuration file for NVIDIA Grid Daemon
...
ProxyUserName=example.com\clsuser
ProxyCredentialsFilePath=/etc/nvidia/proxy-credentials.dat
...
```

9. Save your changes to the `/etc/nvidia/gridd.conf` file and close the file.

10. Restart the `nvidia-gridd` service.

5.1.4. Generating an Encrypted Credentials File

Some authentication methods require a licensed client to provide user credentials when the client authenticates with a proxy server. To enable the client to provide these credentials securely without input from a user, you must generate a file that contains these credentials in an encrypted form that the client can read.

The following authentication methods require an encrypted credentials file:

- ▶ Basic authentication
- ▶ Microsoft Windows Challenge/Response (NTLM) authentication for a client that is not a member of an Active Directory domain

How to generate an encrypted credentials file depends on the OS that client is running. For detailed instructions, refer to the following topics:

- ▶ [Generating an Encrypted Credentials File on Windows](#)
- ▶ [Generating an Encrypted Credentials File on Linux](#)

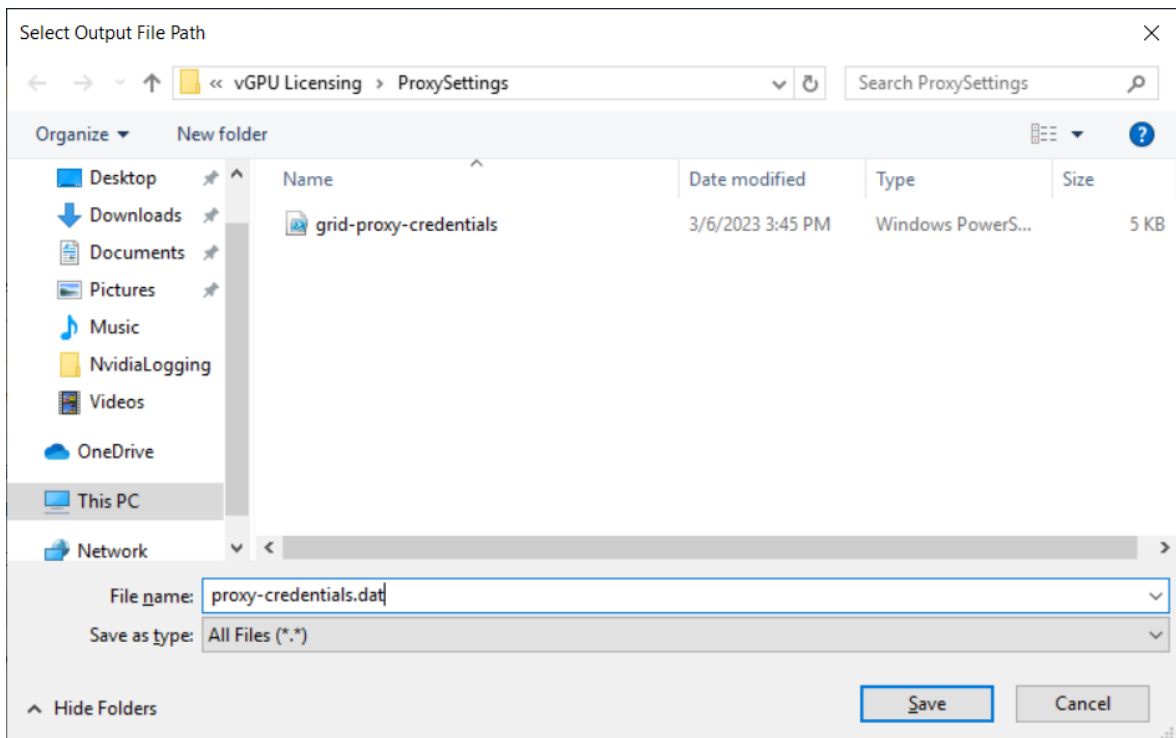
5.1.4.1. Generating an Encrypted Credentials File on Windows

Perform this task in a **Windows PowerShell** window as the Administrator user on the client.

1. Change to the C:\Program Files\NVIDIA Corporation\vGPU Licensing\ProxySettings folder.

```
PS C:\> cd "C:\Program Files\NVIDIA Corporation\vGPU Licensing\ProxySettings"
```
2. Run the grid-proxy-credentials Windows PowerShell script.

```
PS C:\> .\grid-proxy-credentials.ps1
```
3. In the **Select Output File Path** window that opens, navigate to the directory in which you want to generate the credentials file, enter the file name, and click **Save**.



4. When prompted in the **Windows PowerShell** window, specify the password of the user that will log in to the proxy server when the licensed client requests a license.

Provide the path to this file when configuring a licensed client that will use the file as explained in [Configuring a Licensed Client with a Networked License on Windows with Default Settings](#).

5.1.4.2. Generating an Encrypted Credentials File on Linux

Perform this task in a Linux command shell on the client.

1. Run the `grid-proxy-credentials.sh` command.

```
# /usr/lib/nvidia/grid-proxy-credentials.sh -o output-file-path
output-file-path
```

The full path to the credentials file that you are generating. Ensure that the directory in the path exists.



Tip: To get help information for this command, type `/usr/lib/nvidia/grid-proxy-credentials.sh --help`.

This example creates the credentials file `/etc/nvidia/proxy-credentials.dat`.

```
# /usr/lib/nvidia/grid-proxy-credentials.sh -o /etc/nvidia/proxy-credentials.dat
```

2. When prompted, specify the password of the user that will log in to the proxy server when the licensed client requests a license.

Provide the path to this file when configuring a licensed client that will use the file as explained in [Configuring a Licensed Client with a Networked License on Linux with Default Settings](#).

5.2. Configuring a Licensed Client with an NVIDIA vGPU Software Node-Locked License

A client system without a network connection or on an air-gapped network can obtain a node-locked NVIDIA vGPU software license from a file installed locally on the client system.



Note: Support for node-locked licensing was introduced in NVIDIA vGPU software 15.0. It is **not** supported in earlier NVIDIA vGPU software releases.

Before configuring a licensed client with a node-locked license, ensure that the following prerequisites are met:

- ▶ The NVIDIA vGPU software graphics driver is installed on the client.
- ▶ The ZIP archive that contains the node-locked license file with which you want to license the client has been generated and downloaded, and its contents extracted as explained in [Generating Node-Locked Licenses](#).

The NVIDIA vGPU software graphics driver creates a default location in which to store the license file on the client. You can specify a custom location for the license file by adding a registry value on Windows or by setting a configuration parameter on Linux.

The process for configuring a licensed client with a node-locked license depends on the OS that is running on the client.

5.2.1. Configuring a Licensed Client with a Node-Locked License on Windows

Perform this task from the VM or physical host.

1. **Optional:** If you want to store the license file in a custom location, create a local folder on the VM or physical host in which to store the license file.

If you want to store the license file in the default location, omit this step. The default folder in which the license file is stored is created automatically after the NVIDIA vGPU software graphics driver is installed.

2. **Optional:** If you want to store the license file in a custom location, add the `LicenseFilePath` String (REG_SZ) registry value to the Windows registry key `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global\GridLicensing`.



Note: For NVIDIA vGPU software releases before 15.0, add the registry value to the Windows registry key `HKEY_LOCAL_MACHINE\SOFTWARE\NVIDIA Corporation\Global\GridLicensing`.

Set the value to the full path to the folder that you created in the previous step.

If you want to store the license file in the default location, omit this step. By default, the client searches for the license file in the `%SystemDrive%\Program Files\NVIDIA Corporation\vGPU Licensing\License` folder.

3. Use the index file named `index.txt` to identify the license file for the client from its MAC address.

This example shows the `index.txt` file that identifies the license file generated for each client in the following table.

Client MAC Address	License File Name
00:00:5E:00:53:10	1a6e8f90-599e-4459-b2e0-06f67fc183fd.lic
00:00:5E:00:53:27	32894809-3f79-4461-9d53-100b7f6bc338.lic

```
[{"id": "VM0", "pevi": [{"id": "MAC", "val": "00:00:5E:00:53:27", "mod": null}],
"file_name": "32894809-3f79-4461-9d53-100b7f6bc338.lic"},
{"id": "VM1", "pevi": [{"id": "MAC", "val": "00:00:5E:00:53:10", "mod": null}],
"file_name": "1a6e8f90-599e-4459-b2e0-06f67fc183fd.lic"}]
```

4. Copy the NVIDIA license `.lic` file that was generated for the licensed client to a local folder on the client.

Ensure that this folder contains only the license file that you want to deploy on the client and no other files or folders. If the folder contains more than one license file, the client uses the newest license file in the folder.

- ▶ If you want to store the license file in the default location, copy the license file to the %SystemDrive%\Program Files\NVIDIA Corporation\vGPU Licensing\License folder.
- ▶ If you want to store the license file in a custom location, copy the license file to the folder that you specified in the previous step.

5. Reboot the VM or physical host.

The NVIDIA service on the VM or physical host should now automatically obtain a license from the license file.

After a Windows VM or physical host has been configured with a node-locked NVIDIA vGPU software license, options for configuring licensing for a network-based license server are no longer available in **NVIDIA Control Panel**.

5.2.2. Configuring a Licensed Client with a Node-Locked License on Linux

Perform this task from the VM or physical host.

1. **Optional:** If you want to store the license file in a custom location, create a local directory on the VM or physical host in which to store the license file.

If you are storing the license file in the default location, omit this step. The default directory in which the license file is stored is created automatically after the NVIDIA vGPU software graphics driver is installed.

2. **Optional:** If you want to store the license file in a custom location, add the `LicenseFilePath` configuration parameter to the file `/etc/nvidia/gridd.conf`.

If you want to store the license file in the default location, omit this step. By default, the client searches for the license file in the `/etc/nvidia/vGPULicense` directory.

- a). As root, open the file `/etc/nvidia/gridd.conf` in a plain-text editor, such as `vi`.

```
$ sudo vi /etc/nvidia/gridd.conf
```



Note: You can create the `/etc/nvidia/gridd.conf` file by copying the supplied template file `/etc/nvidia/gridd.conf.template`.

- b). Add the following line to the file `/etc/nvidia/gridd.conf`.

```
LicenseFilePath="path"
```

path

The full path to the directory that you created in the previous step.

This example shows how to configure a licensed Linux client to search for the license file in the `/etc/nvidia/nll` directory.

```
# /etc/nvidia/gridd.conf.template - Configuration file for NVIDIA Grid Daemon
...
# Description: Used to specify the directory path which is checked for node-
locked licensing files.
# Data type: string
# Format: "<directory path>"
LicenseFilePath="/etc/nvidia/nll"
...
```

- c). Save your changes to the `/etc/nvidia/gridd.conf` file and close the file.
- Use the index file named `index.txt` to identify the license file for the client from its MAC address.

This example shows the `index.txt` file that identifies the license file generated for each client in the following table.

Client MAC Address	License File Name
00:00:5E:00:53:10	1a6e8f90-599e-4459-b2e0-06f67fc183fd.lic
00:00:5E:00:53:27	32894809-3f79-4461-9d53-100b7f6bc338.lic

```
[{"id": "VM0", "pevi": [{"id": "MAC", "val": "00:00:5E:00:53:27", "mod": null}],
"file_name": "32894809-3f79-4461-9d53-100b7f6bc338.lic"},
{"id": "VM1", "pevi": [{"id": "MAC", "val": "00:00:5E:00:53:10", "mod": null}],
"file_name": "1a6e8f90-599e-4459-b2e0-06f67fc183fd.lic"}]
```

- Copy the NVIDIA license `.lic` file that was generated for the licensed client to a local directory on the client.

Ensure that this directory contains only the license file that you want to deploy on the client and no other files or directories. If the directory contains more than one license file, the client uses the newest license file in the directory.

- ▶ If you want to store the license file in the default location, copy the license file to the `/etc/nvidia/vGPUlicense` directory.
- ▶ If you want to store the license file in a custom location, copy the license file to the directory that you specified in the previous step.

- Reboot the VM or physical host.

The NVIDIA service on the VM or physical host should now automatically obtain a license from the license file.

After a Linux VM or physical host has been configured with a file-based NVIDIA vGPU software license, options for configuring licensing for a networked license server are no longer available in **NVIDIA X Server Settings**.

5.3. Verifying the NVIDIA vGPU Software License Status of a Licensed Client

After configuring a client with an NVIDIA vGPU software license, verify the license status by displaying the licensed product name and status.

To verify the license status of a licensed client, run `nvidia-smi` with the `-q` or `--query` option from the licensed client, **not** the hypervisor host. If the product is licensed, the expiration date is shown in the license status.

```
nvidia-smi -q
=====NVSMI LOG=====
Timestamp                               : Wed Nov 23 10:52:59 2022
```



```

Driver Version           : 525.60.06
CUDA Version            : 12.0

Attached GPUs           : 2
GPU 00000000:02:03.0
  Product Name          :
  Product Brand         : NVIDIA RTX Virtual Workstation
  Product Architecture  : Ampere
  Display Mode          : Enabled
  Display Active        : Disabled
  Persistence Mode     : Enabled
  MIG Mode
    Current              : Disabled
    Pending              : Disabled
  Accounting Mode       : Disabled
  Accounting Mode Buffer Size : 4000
  Driver Model
    Current              : N/A
    Pending              : N/A
  Serial Number         : N/A
  GPU UUID              : GPU-ba5b1e9b-1dd3-11b2-be4f-98ef552f4216
  Minor Number          : 0
  VBIOS Version         : 00.00.00.00.00
  MultiGPU Board        : No
  Board ID              : 0x203
  Board Part Number     : N/A
  GPU Part Number       : 25B6-890-A1
  Module ID             : N/A
  Inforom Version
    Image Version        : N/A
    OEM Object           : N/A
    ECC Object           : N/A
    Power Management Object : N/A
  GPU Operation Mode
    Current              : N/A
    Pending              : N/A
  GSP Firmware Version : N/A
  GPU Virtualization Mode
    Virtualization Mode : VGPU
    Host VGPU Mode      : N/A
vGPU Software Licensed Product
  Product Name       : NVIDIA RTX Virtual Workstation
  License Status    : Licensed (Expiry: 2022-11-23 10:41:16
GMT)
  ...
  ...

```

Chapter 6. Administering a Service Instance

Perform these routine administration tasks as needed during the lifetime of the service instance after completing the initial configuration of the service instance.

6.1. Migrating a DLS Instance

Migrating a DLS instance simplifies the upgrade of a DLS appliance. After installing a new version of the DLS appliance, you can transfer the license servers, user registration, IP address, and service instance from the existing DLS appliance to a new DLS appliance. However, event records on the existing DLS appliance are **not** migrated.

Considerations for Migrating an HA Cluster of DLS Instances

If you are upgrading the DLS appliance for the DLS instances in an HA cluster, migrate **only** the primary instance. During the migration process, all data is removed from the secondary DLS instance and the instance is removed from the cluster. After completing the migration process for the primary instance, you can configure an HA cluster from the new primary instance.

Types of Migrations Supported by NVIDIA License System

The type of migration to perform depends on whether the new version of the DLS virtual appliance is a software update that contains security updates and bug fixes or is a new major version.

In-place upgrade

An in-place upgrade enables you apply a software update to a DLS appliance when NVIDIA provides security updates and bug fixes. Upgrading a DLS appliance in place overinstalls the software updates on the existing appliance without any disruption in service. All data on the appliance is preserved during the upgrade. Because the

updates are overinstalled on the existing appliance, you **cannot** change the platform type of a DLS appliance by performing an in-place upgrade.



Note: In-place upgrades are supported starting with NVIDIA License System 3.0.0. NVIDIA License System versions earlier than 3.0.0 do not support in-place upgrades.

File Based Upgrade

This is a new form of Upgrade that replaces Portal Assisted Upgrade and is available from DLS Virtual Appliance 3.3.0 onwards. In this Upgrade, you need not upload any file to the NVIDIA Licensing Portal.



Note: There could be some downtime associated with performing a File Based Upgrade.

Before You Begin

Ensure that the port mappings for ports 8081 and 8084 are not changed from their defaults.

To prevent loss of data if a migration fails, create a snapshot of your DLS appliance before migrating it. If a migration fails, you can restore your DLS appliance from the snapshot. For information about how to create a snapshot and restore a DLS appliance from the snapshot, refer to the instructions for your appliance type in the following table.

Appliance Type	Creating a Snapshot	Restoring an Appliance
Citrix Hypervisor VM	Create a VM snapshot	Restore a VM to its previous state
Linux Kernel-based Virtual Machine (KVM) VM	How to create snapshots of QEMU/KVM guests (linuxconfig.org)	How to create snapshots of QEMU/KVM guests (linuxconfig.org)
Microsoft Windows Server with Hyper-V VM	Using checkpoints to revert virtual machines to a previous state	Applying checkpoints
Red Hat Enterprise Linux Kernel-based Virtual Machine (KVM) VM	Creating Snapshots	snapshot-revert
Red Hat Virtualization VM	Creating Snapshots	snapshot-revert
Ubuntu Hypervisor VM	How to create snapshots of QEMU/KVM guests (linuxconfig.org)	How to create snapshots of QEMU/KVM guests (linuxconfig.org)
VMware vSphere Hypervisor (ESXi) VM	Take a Snapshot in the VMware Host Client	Restoring Snapshots

Appliance Type	Creating a Snapshot	Restoring an Appliance
Container	Back up the DLS container volume	Stop the DLS container, restore the DLS container volume, and start the container again.

Considerations for Migrating a DLS 1.0.0 or 1.0.1 Instance

If you are migrating a DLS 1.0.0 or 1.0.1 instance, the version shown on the **VA Upgrade Job Progress** page is 1.0.0 to preserve backwards compatibility.

After migrating a DLS 1.0.0 or 1.0.1 instance, you must change the password for the `dls_admin` user to enable or disable the `dls_system` user on the new instance.

Instructions for Upgrading from Earlier Releases

The instructions in this document explain how to upgrade from this release to a later release. For upgrading from an earlier release, follow the instructions specified for that release:

- ▶ [Instructions for upgrading from DLS 3.1.0](#)
- ▶ [Instructions for upgrading from DLS 3.0.0](#)
- ▶ [Instructions for upgrading from DLS 2.1.0](#)
- ▶ [Instructions for upgrading from DLS 2.0.1](#)
- ▶ [Instructions for upgrading from DLS 2.0.0](#)
- ▶ [Instructions for upgrading from DLS 1.1](#)
- ▶ [Instructions for upgrading from DLS 1.0.1](#)
- ▶ [Instructions for upgrading from DLS 1.0](#)

Performing a File-Based Migration of a DLS Instance

File based migration of a DLS instance enables you to perform an upgrade based on a binary migration file of a DLS instance when NVIDIA releases a new major version of the DLS virtual appliance.

1. [Initiating Upgrade on a DLS Instance](#)
2. Instructions for preparing a new appliance from which to generate a DLS Migration Token:
 - ▶ [Preparing a New Appliance for a Standalone DLS Instance](#)
 - ▶ [Preparing a New Appliance for a VM-Based Node in an HA Cluster](#)
3. [Generating a Migration File for the DLS Instance that You are Migrating](#)
4. [Transferring Migration Data to the DLS Instance on an Upgraded Virtual Appliance](#)
5. **For an HA cluster:** [Synchronizing Changes with the Primary Node in an HA Cluster](#)

6. For an HA cluster: [Configuring an HA Cluster of DLS Instances](#)

To perform a file-based upgrade of a DLS instance, follow this sequence of instructions:

6.1.1. Upgrading a DLS Appliance in Place

To provide security updates and bug fixes, NVIDIA periodically releases updates to the DLS appliance software. To apply a software update to a DLS appliance, upgrade the DLS appliance in place. Upgrading a DLS appliance in place overinstalls the software updates on the existing appliance without any disruption in service. All data on the appliance is preserved during the upgrade.

To upgrade a DLS appliance in place, follow this sequence of instructions:

1. [Initiating an In-Place Upgrade to a DLS Appliance](#)
2. The instructions for the type of platform on which the DLS appliance is hosted:
 - ▶ [Upgrading a Container-Based DLS Appliance in Place](#)
 - ▶ [Upgrading a VM-Based DLS Appliance in Place](#)

6.1.1.1. Initiating an In-Place Upgrade to a DLS Appliance

Initiating an in-place upgrade to a DLS appliance puts the instance that is hosted on the appliance into maintenance mode. If the instance is a node in an HA cluster of DLS instances, **all** nodes in the cluster are put into maintenance mode. In maintenance mode, a DLS instance can perform **only** licensing operations. All other operations are disabled.

For an HA cluster: Perform this task **only** from the DLS virtual appliance that hosts the current **primary** instance.

1. Log in to the DLS virtual appliance for which you want to initiate an in-place upgrade.
2. In the left navigation pane of the **NVIDIA Licensing** dashboard, click **MAINTENANCE**.
3. On the **Maintenance** page that opens, scroll down to the **Virtual Appliance Upgrade** section of the page.
4. Ensure that the **File Based Upgrade** option is **not** set.
5. Click **START IN-PLACE UPGRADE**.

Follow the instructions for the type of platform on which the DLS appliance is hosted:

- ▶ [Upgrading a Container-Based DLS Appliance in Place](#)
- ▶ [Upgrading a VM-Based DLS Appliance in Place](#)

6.1.1.2. Upgrading a Container-Based DLS Appliance in Place

Ensure that the following prerequisites are met:

- ▶ The ZIP archive that contains the update to NVIDIA License System has been downloaded from the NVIDIA Licensing Portal.

- The upgrade has been initiated as explained in [Initiating an In-Place Upgrade to a DLS Appliance](#).

For an HA cluster: Perform this task on each DLS appliance that hosts an instance in the cluster. Upgrade the DLS appliance for every other instance in the cluster **before** upgrading the DLS appliance that hosts the current **primary** instance.

1. Stop the DLS appliance container.
2. As the **postgres** user, run the `/etc/dls/take-snapshot.sh` script to create a snapshot of the PostgreSQL database container.

The command for running the script depends on the container orchestration platform that you are using.

Container Orchestration Platform	Command
Docker	<pre>\$ docker exec -it --user postgres config-postgres-nls-si-0-1 /etc/dls/ take-snapshot.sh</pre>
Kubernetes	<pre>\$ kubectl exec -it db-container-pod -- su postgres -c "/etc/dls/take- snapshot.sh" db-container-pod The name of the PostgreSQL database container pod.</pre>
Podman	<pre>\$ podman exec -it --user postgres config-postgres-nls-si-0-1 /etc/dls/ take-snapshot.sh</pre>
Red Hat OpenShift Container Platform	<pre>\$ kubectl exec -it db-container-pod -- su postgres -c "/etc/dls/take- snapshot.sh" db-container-pod The name of the PostgreSQL database container pod.</pre>
VMware Tanzu Application Platform	<pre>\$ kubectl exec -it db-container-pod -- su postgres -c "/etc/dls/take- snapshot.sh" db-container-pod The name of the PostgreSQL database container pod.</pre>

3. Stop the PostgreSQL database container.
4. Back up the `postgres-data`, `logs`, `rabbitmq_data`, and `configurations` volumes.
5. Clean up the `postgres-data` and `rabbitmq_data` volumes.



Note: Do not clean up the `configurations` volume.

6. Extract the contents of the ZIP archive that contains the update to NVIDIA License System .

```
$ sudo unzip nls-version-bios.zip
```

version

The version number of NVIDIA License System that you are upgrading to, for example, **2.2.1**.

7. Deploy the updated containers on the existing host with the existing volume mappings.

Follow the instructions for the container orchestration platform that you are using.

Container Orchestration Platform	Instructions
Docker	Deploying the Containerized DLS Software Image on Docker
Kubernetes	Deploying the Containerized DLS Software Image on Kubernetes Platforms
Podman	Deploying the Containerized DLS Software Image on Podman
Red Hat OpenShift Container Platform	Deploying the Containerized DLS Software Image on Kubernetes Platforms
VMware Tanzu Application Platform	Deploying the Containerized DLS Software Image on Kubernetes Platforms

8. Confirm that the **NVIDIA Licensing** application and associated software have been upgraded and that the existing data has been preserved on the DLS appliance.
 - a). Log in as the **dls_admin** user to the DLS appliance that you just upgraded.
 - b). In the left navigation pane, click **SUPPORT**.
 - c). On the **Support** page that opens, confirm that the **NVIDIA Delegated License System** field displays the correct software version.
 - d). In the left navigation pane, click **SERVICE INSTANCE** and confirm that the details of the DLS instance on the appliance have been preserved.
 - e). In the left navigation pane, click **DASHBOARD** and confirm that the details of the installed license server have been preserved.

6.1.1.3. Upgrading a VM-Based DLS Appliance in Place

Initiating an in-place upgrade to a DLS appliance puts the instance that is hosted on the appliance into maintenance mode. If the instance is a node in an HA cluster of DLS instances, **all** nodes in the cluster are put into maintenance mode. In maintenance mode, a DLS instance can perform **only** licensing operations. All other operations are disabled.

For an HA cluster: Perform this task **only** from the DLS virtual appliance that hosts the current **primary** instance.

1. Log in to the DLS virtual appliance for which you want to initiate an in-place upgrade
2. In the left navigation pane of the NVIDIA Licensing dashboard, click **MAINTENANCE**.
3. On the Maintenance page that opens, scroll down to the **Virtual Appliance Upgrade** section of the page.
4. Ensure that the **File-Based Upgrade Option** is not selected.

5. Click **START IN-PLACE UPGRADE**.

Ensure that the following prerequisites are met:

- ▶ The ZIP archive that contains the update to NVIDIA License System has been downloaded from the NVIDIA Licensing Portal.
- ▶ DLS Virtual Machine Appliance source node is on version 3.3.0 with license server installed.
- ▶ DLS Appliance Virtual Machine snapshot is created.

For an HA cluster: Perform this task on each DLS appliance that hosts an instance in the cluster. Upgrade the DLS appliance for every other instance in the cluster **before** upgrading the DLS appliance that hosts the current **primary** instance.

1. Login to the DLS node.
2. Navigate to the "MAINTENANCE" tab.
3. In the "Virtual Appliance Upgrade" section select "In-Place Upgrade" toggle.
4. Trigger In-place Upgrade. Users can do this by clicking on the user interface button "START IN-PLACE UPGRADE PROCESS".
 - a). Doing this will put all nodes in maintenance mode for upgrade in addition to enabling the navigation interface for Upgrade Workflow.
 - b). During this time, the standalone node will continue serving the leases as long as the node is up during the upgrade.
 - c). Users won't be able to perform any admin operations on the license server.
 - d). Once the OPEN DLS UPGRADE WORKFLOW button is enabled on the user interface on the "Maintenance" page of DLS UI, users can navigate to the "Upgrade" workflow screen on a separate browser tab.
5. On the "Upgrade" workflow screen users should upload the container images zip.
6. Upgrade will start as soon as the container images zip file upload is completed. This process is composed of 10 steps, the details of which is seen on the right side on the Upgrade Workflow screen.
7. Upgrade screen workflow display's step by step details as it progresses.
8. In step 10 of the upgrade process, the system will wait for new containers to start up. It takes up to 10 minutes for the appliance services and containers to complete the startup.
 - a). You can opt to "Rollback" the upgrade if the new containers/services don't startup on time.
 - b). You can opt to "Restart" the containers in—order to resolve any transient issues.
9. Once step 10 completes with a successful container startup, the upgrade is deemed to be completed and users will be able to navigate to the DLS UI portal using the UI button visible on Upgrade Workflow screen.
 - a). This step will close the "Upgrade workflow screen" and it will not be accessible anymore.

6.1.1.4. Troubleshooting the In-Place Upgrade of a DLS Appliance

If the in-place upgrade of a DLS appliance fails, you can revert the appliance to its previous version. How to revert the appliance to its previous version depends on whether the upgrade failed for an entire HA cluster of DLS instances or only for some instances in the cluster.

- ▶ **Container-based appliance:**
 1. Stop the upgraded DLS appliance container.
 2. Restore the backed up `postgres-data`, `logs`, and `rabbitmq_data` volumes.
 3. Deploy the previous version of the containers on the existing host with the existing volume mappings.
- ▶ If the upgrade failed only for some instances in the cluster, recreate the cluster from a node that was successfully upgraded.
 1. Mark the upgraded node as the primary node in the cluster.
 2. Remove from the cluster any node for which the upgrade failed.
 3. Install or deploy a new DLS appliance with the same network configuration as the appliance that failed to be created.
 4. Add the new DLS appliance to the cluster.

6.1.2. Performing a File-Based Migration of a DLS Instance

File based migration of a DLS instance enables you to perform an upgrade based on a binary migration file of a DLS instance when NVIDIA releases a new major version of the DLS virtual appliance.



Note:

For HA based File Based migration, licensing clients could see some transient failures when re-creating the cluster after the migration.

These transient failures would be resolved once HA cluster is restored after the Upgrade process is complete. If some clients are still failing they would need to be restarted.

To perform a file-based upgrade of a DLS instance, follow these steps:

1. [Initiating an Upgrade on a DLS Instance](#)
2. Instructions for preparing a new appliance from which to generate a DLS Migration token:
 - ▶ [Preparing a New Appliance for a Standalone DLS Instance](#)
 - ▶ [Preparing a New Appliance for a VM-Based Node in an HA Cluster](#)

3. [Generating a Migration File for the DLS Instance that You are Migrating](#)
4. [Transferring Migration Data to the DLS Instance on an Upgraded Virtual Appliance](#)
5. **For an HA cluster:** [Synchronizing Changes with the Primary Node in an HA Cluster](#)
6. **For an HA cluster:** [Configuring an HA Cluster of DLS Instances](#)

6.1.2.1. Initiating an Upgrade on a DLS Instance

1. Log in to the existing DLS virtual appliance that hosts the DLS instance that you want to migrate.
The DLS instance must be either a standalone DLS instance or the primary node in an HA cluster.
2. In the left navigation pane of the **NVIDIA Licensing** dashboard, click **MAINTENANCE**.
3. On the **Maintenance** page that opens, scroll down to the **Virtual Appliance Upgrade** section of the page.
4. Set the **File Based Upgrade** option.
5. If the DLS instance is the primary node in an HA cluster, a table that lists the nodes in the cluster is added to the Maintenance page. Each node in the table **except the primary node** has an **UPGRADE** button. In the Trigger File Based Upgrade window that opens, click **UPGRADE**.
6. Modifications to the existing DLS virtual appliance are blocked until the file based migration is complete. The state of the instance during the migration process depends on whether the instance is a standalone instance or a node in an HA cluster.
 - a). If the instance is a standalone DLS instance, all operations by the instance, including licensing operations, are disabled.
 - b). If the instance is a node in an HA cluster of DLS instances, all data is removed from the secondary nodes and the secondary nodes are removed from the cluster. The **primary** node in the cluster is put into maintenance mode and continues to operate as a standalone DLS instance. In maintenance mode, a DLS instance can perform **only** licensing operations. All other operations are disabled.

Prepare a new appliance from which to generate a DLS instance token. How to prepare the appliance depends on the type of the appliance and the type of node that the appliance will host.

- ▶ [Preparing a New Appliance for a Standalone DLS Instance](#)
- ▶ [Preparing a New Appliance for a VM-Based Node in an HA Cluster](#)

6.1.2.2. Preparing a New Appliance for a Standalone DLS Instance

Migrating a DLS instance requires a DLS Migration token to be generated from an appliance that is running the new version of the DLS software. Preparing a new appliance for a standalone DLS instance involves installing or deploying the new version of the DLS appliance in the same way as for a first-time installation of a DLS appliance.

Ensure that the ZIP archive that contains the latest DLS virtual appliance image has been downloaded from the NVIDIA Licensing Portal.

1. Install or deploy the new version of the DLS appliance. For detailed instructions, refer to the following topic: [Installing the DLS Virtual Appliance Image on a Supported Hypervisor](#)
2. Start the new VM that hosts the upgraded DLS appliance.

Once you are able to see the web interface for the node installed above, click on Upgrade. From the Upgrade page, select **File Based Upgrade** and download the **DLS Migration Token**.

Generate and upload the DLS Migration token for the new virtual appliance to generate a migration file as explained in [Generating a Migration File for the DLS Instance that You are Migrating](#)

6.1.2.3. Preparing a New Appliance for a VM-Based Node in an HA Cluster

Migrating a DLS instance requires a DLS Migration token to be generated from an appliance that is running the new version of the DLS software. Preparing a new appliance for a VM-based node in an HA cluster involves installing the new version of the DLS appliance in the same way as for a first-time installation of a DLS appliance. The new appliance will host the secondary node in the cluster.

Ensure that the ZIP archive that contains the latest DLS virtual appliance image for your chosen hypervisor has been downloaded from the NVIDIA Licensing Portal.

1. Shut down the VM that hosts the existing secondary node that you have chosen to for upgrade in a migration file in [Initiating an Upgrade on a DLS Instance](#).
2. Install the new version of the DLS appliance on your chosen hypervisor as explained in [Installing the DLS Virtual Appliance Image on a Supported Hypervisor](#).
3. Configure the VM that hosts the new version of the DLS appliance with the same network properties as the VM that hosts the existing secondary node.

Once you are able to see the web interface for the node installed above, click on Upgrade. From the Upgrade page, select **File Based Upgrade** and download the **DLS Migration Token**.

Upload the DLS Migration token for the new virtual appliance to generate a migration file as explained in [Generating a Migration File for the DLS Instance that You are Migrating](#).

6.1.2.4. Generating a Migration File for the DLS Instance that You are Migrating

After generating the migration token for the upgraded DLS instance, you must generate a migration file for the instance. Generating the migration file involves generating and uploading the DLS migration token for the new virtual appliance that will host the migrated DLS instance.

Ensure that you have the DLS migration token from upgraded DLS instance as explained in the following topics:

- ▶ [Preparing a New Appliance for a Standalone DLS Instance](#)
 - ▶ [Preparing a New Appliance for a VM-Based Node in an HA Cluster](#)
1. Log in to the existing DLS virtual appliance that hosts the DLS instance that you want to migrate.
 2. In the left navigation pane of the NVIDIA Licensing dashboard, click **MAINTENANCE**.
 3. On the Maintenance page that opens, scroll down to the **Virtual Appliance Upgrade** section of the page.
 4. Set the **File Based Upgrade** option.
 5. Upload the Migration Token downloaded from the upgraded DLS instance. When prompted, confirm that you want to upload the migration token.
 6. Wait for the Migration File to be generated. After the migration file has been generated, click **DOWNLOAD MIGRATION FILE**. A migration file that is named `on_prem_migration_file_mm-dd-yyyy-hh-mm-ss.bin` is saved to your downloads folder.

Once you have the migration file, follow the instructions in [Transferring Migration Data to the DLS Instance on an Upgraded Virtual Appliance](#)

6.1.2.5. Transferring Migration Data to the DLS Instance on an Upgraded Virtual Appliance

1. Return to the Virtual Appliance Upgrade page of the NVIDIA Licensing application on the upgraded virtual appliance.
2. On the Virtual Appliance Upgrade page, select File Base Upgrade.
3. Click on **UPLOAD MIGRATION FILE**.
4. In the Upload Migration File pop-up window that opens, click **Choose File**.
5. In the file browser that opens, navigate to the folder that contains the migration file that is named `on_prem_migration_file_mm-dd-yyyy-hh-mm-ss.bin` that you downloaded and select the file.
6. Back in the Upload Migration File, click **UPLOAD**.

If you are transferring migration data for a standalone DLS instance, no further action is required. The virtual appliance is ready for use.

If you are transferring migration data for a node in an HA cluster of DLS instances, synchronize data changes with the primary node in the cluster as explained in [Synchronizing Changes with the Primary Node in an HA Cluster](#). The data changes to synchronize arise from licensing operations that were performed during the migration.

6.1.2.6. Synchronizing Changes with the Primary Node in an HA Cluster

1. Log in to the **existing** DLS virtual appliance that hosts the DLS instance from which you want to synchronize data.
This instance is operating in maintenance mode.
2. In the left navigation pane of the **NVIDIA Licensing** dashboard, click **MAINTENANCE**.
3. On the **Maintenance** page that opens, scroll down to the **Virtual Appliance Upgrade** section of the page.
4. In table that lists the nodes in the cluster, click **Synchronize changes**.
Any changed data on the existing DLS instance is copied to the new DLS instance. After the changed data has been copied, the old and the new instance are operating in maintenance mode.
5. Log in to the **new** DLS virtual appliance that hosts the DLS instance to which the data was synchronized.
If you are already logged in, reload the browser page.
6. On the **NVIDIA Licensing** dashboard, click **Acknowledge**.
All data on the existing DLS instance is removed. The new DLS instance is now the primary node in the cluster and continues to be updated with changes from the existing DLS instance. When the new DLS instance is updated with all changes from the existing instance, the existing DLS instance is deleted from the DLS appliance that hosts it.
7. Delete the DLS appliance that hosted the existing DLS instance.

For seven days after the new DLS instance has been updated, the **Maintenance** displays information about the cluster nodes before migration. You can use this information to re-create the cluster after the migration.

1. Install or deploy a new DLS virtual appliance for the secondary nodes in the cluster as explained in the following topics: [Installing the DLS Virtual Appliance Image on a Supported Hypervisor](#)

To get information such as IP addresses about the nodes in the cluster before the upgrade, refer to the **Maintenance** page for the upgraded primary node.
2. Add the new secondary nodes to the cluster as explained in [Configuring an HA Cluster of DLS Instances](#).

6.2. Setting the Validity Period of a Lease Authorization Token for a Service Instance

You can set the validity period of a lease authorization token to either enhance performance or increase security. Increasing the validity period enhances performance by decreasing the frequency with which clients are authorized before the service instance grants a licensing request. Decreasing the expiration time increases security by increasing the frequency with which clients are authorized before the service instance grants a licensing request.

The default validity period is one hour. You can set the validity period to any value up to 24 hours.

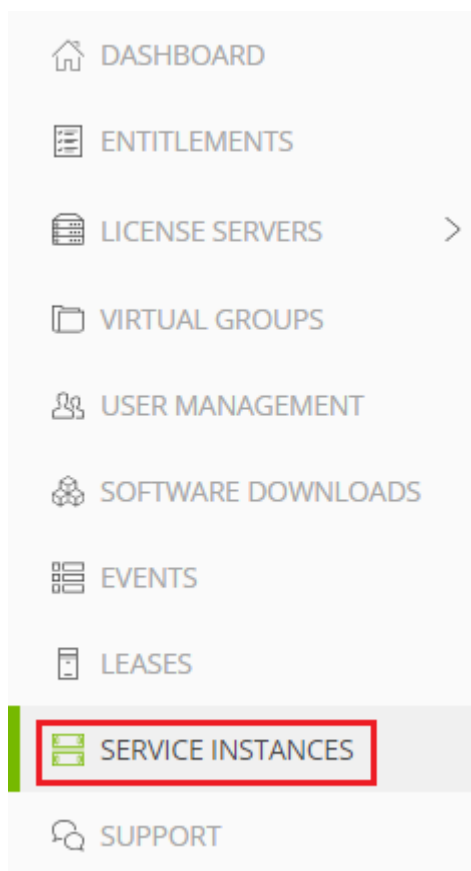
How to set the validity period of a lease authorization token for a service instance depends on whether you are setting it for a CLS instance or a DLS instance. For detailed instructions, see:

- ▶ [Setting the Validity Period of a Lease Authorization Token for a CLS Instance](#)
- ▶ [Setting the Validity Period of a Lease Authorization Token for a DLS Instance](#)

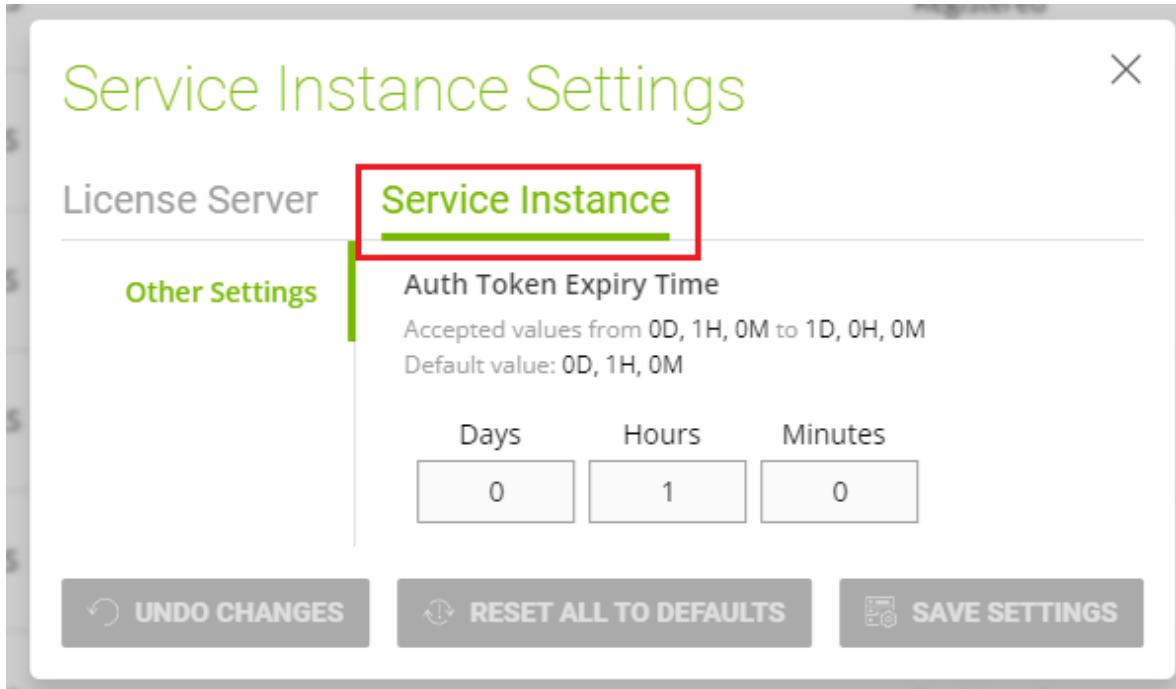
6.2.1. Setting the Validity Period of a Lease Authorization Token for a CLS Instance

Perform this task on the NVIDIA Licensing Portal.

1. If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal, then select **SERVICE INSTANCES** from the left navigation pane.



2. Find the CLS server that you want to adjust.
3. Select the **Actions** menu from the right-hand side of the screen, then click **Settings** from the drop-down.
4. Select **Service Instance Settings** at the top of the window that opens, modify **Auth Token Expiry Time** as needed, then click **SAVE SETTINGS**.



6.2.2. Setting the Validity Period of a Lease Authorization Token for a DLS Instance

Perform this task on the **NVIDIA Licensing** application on the virtual appliance that hosts the DLS instance.

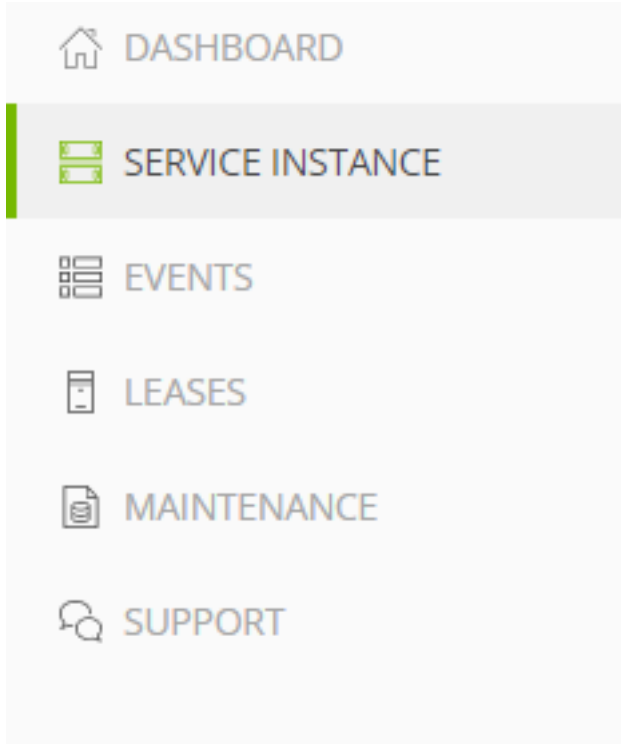
1. Open a web browser and connect to the URL `https://dls-vm-ip-address`.

dls-vm-ip-address

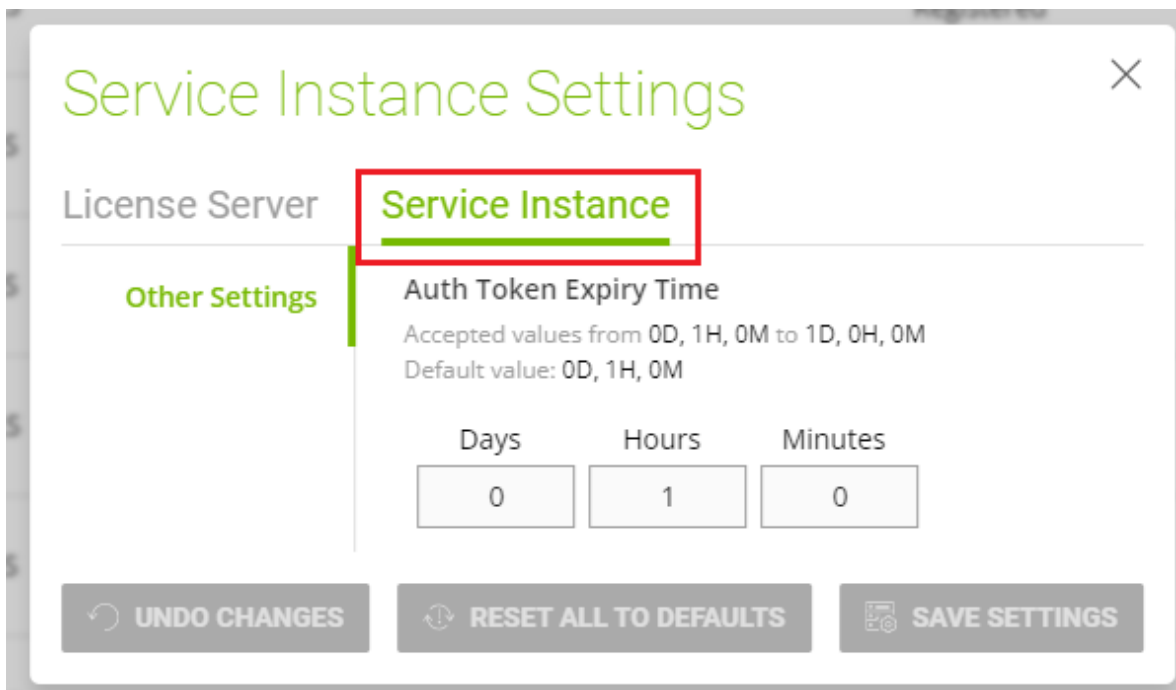
The IP address or, if defined, the fully qualified domain name or the CNAME of the VM on which the DLS virtual appliance is installed.

You can get the IP address from the management console of your hypervisor.

2. On the login page that opens, provide the user credentials for the DLS administrator user on the DLS virtual appliance and click **LOGIN**.
3. In the left navigation pane of the **NVIDIA Licensing** dashboard, click **SERVICE INSTANCE**.



4. On the **Service Instance** page that opens, click **EDIT SETTINGS**.
5. Select **Service Instance Settings** at the top of the window that opens, modify **Auth Token Expiry Time** as needed, then click **SAVE SETTINGS**.



6.3. Specifying the Retention Period for Client Registrations

To prevent excessive consumption of memory in its database, a DLS instance periodically removes details about licensed clients that have been inactive for a specific period of time. An inactive client does not have any licenses checked out from a license server that is bound to the DLS instance. You can control how long a client must be inactive before its details are removed.

1. Enable the sudo user **rsu_admin** on the DLS appliance.
2. Log in to the appliance using the **rsu_admin**.
3. You can either list the current configuration values or update these values.

- ▶ **List:** Get the current values by running this command.

```
sudo python /etc/adminscripts/
configure_client_registrations_clean_up_settings.py --action list --username
dls_user --password dls_password
```

- ▶ **Update:** Update the existing values by running this command.

```
sudo python /etc/adminscripts/
configure_client_registrations_clean_up_settings.py --action update --username
dls_user --password dls_password -time-window-for-purging-inactive-clients
P1D -time-threshold-for-inactive-clients-registration P5D
```

Specify the following configuration values in the format described in [ISO8601 Duration Format](#).

time-window-for-purging-inactive-clients

Licensed clients who do not have any active lease, and there is no lease activity in this configured window to be purged.

time-threshold-for-inactive-clients-registration

A time window of client registration to consider when purging inactive clients.



Note: The previous settings from the `/etc/dls/network/registered_origin_cleanup_config.properties` file in DLS 3.1.0 will not be migrated to DLS versions 3.2.0 and later. The current settings defined through the script, which are part of DLS 3.2.0, will be migrated to newer DLS versions.

6.4. Gathering License Usage Statistics

You can use license usage statistics to estimate the number of licenses that you need to purchase. These statistics are available only from DLS instances. They are **not** available from CLS instances.

Usage statistics require event data and if no events are available, the statistics might be incorrect. To ensure that the statistics gathered are correct, set the retention period for events to 90 as explained in [Setting the Retention Period of Events on a DLS Instance](#).

1. Open a web browser and connect to the URL `https://dls-vm-ip-address`.

dls-vm-ip-address

The IP address or, if defined, the fully qualified domain name or the CNAME of the VM on which the DLS virtual appliance is installed.

You can get the IP address from the management console of your hypervisor.

2. On the login page that opens, provide the user credentials for the DLS administrator user on the DLS virtual appliance and click **LOGIN**.
3. In the left navigation pane of the **NVIDIA Licensing** dashboard, click **METRICS**.
4. In the **Metrics** page that opens, specify the details of the statistics that you want to gather and click **GET LICENSE UTILIZATION**.
 - a). Under **Date range (up to 92 days)**, use the calendar widgets to set the start date and end date of the period during which you want to gather license usage statistics.
 - b). From the **Metric** drop-down list, select the statistic that you want to gather.
 - c). Under **Features**, select the licensed products for which you want to gather usage statistics.
 - d). To exclude licenses in use that were checked out before the start date of the period during which you want to gather license usage statistics, select the **Exclude prior leases** option.

The statistics that you requested are added to the table at the bottom of the **Metrics** page.

5. **Optional:** If you want a report in CSV format, create the report as follows:
 - a). At the top right of the table at the bottom of the **Metrics** page, click the **Export data** icon.
 - b). In the **Export Data** pop-up window that opens, specify the name of the CSV file to be exported and click **EXPORT**.

A `.csv` file is saved to your default downloads folder.

6.5. Configuring Email Alerts for Licensing Events

To avoid disruption to your services, you can configure email alerts for licensing events that relate to the expiration or exhaustion of NLS resources. These alerts remind you to renew the affected resources before they expire or are exhausted. By default, NLS does not send any alerts. If you want NLS to send alerts, you must create them. After creating an alert, you can edit it and, if you want NLS to stop sending it, disable it.

You can configure email alerts for the NVIDIA Licensing Portal and for a DLS appliance. The types of licensing events for which you can configure email alerts depend on the scope of the alerts.

Licensing Event Type	Scope
Lease consumption	► NVIDIA Licensing Portal

Licensing Event Type	Scope
	► DLS appliance
License expiration	NVIDIA Licensing Portal
API key expiration	NVIDIA Licensing Portal
Failover event	DLS appliance
Service availability	DLS appliance
System resources	DLS appliance

For each type of event, you can specify who should receive email alerts and other properties specific to each type of event that determine the criteria for sending the alerts. However, the content of the alert for each event type is preset and cannot be changed.

If you are configuring email alerts for a DLS appliance, ensure that the DLS appliance can reach a Simple Mail Transfer Protocol (SMTP) server for sending the alerts.

1. Navigate to the part of the **NVIDIA Licensing** application for configuring email alerts for the scope that you are interested in.

Scope

NVIDIA Licensing Portal

Steps

- a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.

The NVIDIA Licensing Portal dashboard opens.

- b). If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
- c). In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **EMAIL ALERTS**.

DLS appliance

- a). If you are not already logged in, log in to the **NVIDIA Licensing** application on the VM or container in which the DLS appliance is installed or deployed.

Scope**Steps**

- b). In the left navigation pane, click **SETTINGS**.
- c). On the **DLS Instance Settings** page that opens, expand the **Email Alerts** section.

If no SMTP server has been specified for the DLS appliance, you will be prompted to provide details of the SMTP server for sending alerts from the appliance.

2. If prompted, provide details of the SMTP server for sending alerts from this DLS appliance and click **CONFIGURE SMTP**.

You can change details that were provided previously by following the **Update SMTP configuration** link. You can also change the from email address used for sending emails from DLS instances with the SMTP Relay.

Host

The fully qualified domain name, for example, `smtp.example.com`, or the IP address of the SMTP server for sending alerts from the appliance.

Username

The name of the user that the DLS appliance will log in as to the SMTP server to send alerts from the appliance. Username must identify a valid user account on the SMTP server.

This field is optional.

Password

The user's password for logging in to the SMTP server.

This field is optional.

Security

- ▶ **Forced SSL:**

A client will try to establish a secure connection without asking a server about its compatibility. If it succeeds, a secure connection will be set up and a handshake will follow. If a server is not compatible or a connection times out, a transmission will be abandoned.

- ▶ **Opportunistic SSL:**

A client will run a `STARTTLS` command to upgrade a connection to an encrypted one. If a server is compatible and no errors occur, the secured SSL connection will be established. If anything fails in the process, a plain-text transmission will be established.

- ▶ **No SSL:**

No SSL cryptographic protocol is used in email transmission.

Port Number

The SMTP port that manages information transfers from one server to another. The available port numbers are 587, 465, 25, and 2525.

3. Create or edit the alert that you are interested in.
 - ▶ To create an alert, click **Create**.
 - ▶ To edit an alert, click **Edit**.
4. In the pop-up window that opens, specify who should receive email alerts and other properties specific to each type of event that determine the criteria for sending the alerts **CREATE ALERT** or **EDIT ALERT**.

Event Type**NVIDIA Licensing Portal only: API Key Expiration****Steps**

- a). **Editing an alert only:** If you want to enable or disable the alert, select or deselect the **Enabled** option.



Note: When you create an alert, the alert is enabled automatically.

- b). For each intended recipient of the alert, type the recipient's email address into the text-entry field and click the plus-sign (+) next to the field.

Ensure that the email address corresponds to a registered contact in your organization.

The address is added to the list of addresses in the **E-mail addresses to notify** field.

- c). In the **Number of days prior to expiration to receive notification** text-entry field, type the number of days before the API key expires that NLS should send an alert.

NVIDIA Licensing Portal only: Feature Expiration

- a). **Editing an alert only:** If you want to enable or disable the alert, select or deselect the **Enabled** option.



Note: When you create an alert, the alert is enabled automatically.

Event Type

Steps

- b). For each intended recipient of the alert, type the recipient's email address into the text-entry field and click the plus-sign (+) next to the field.

Ensure that the email address corresponds to a registered contact in your organization.

The address is added to the list of addresses in the **E-mail addresses to notify** field.

- c). In the **Number of days prior to expiration to receive notification** text-entry field, type the number of days before the license expires that NLS should send an alert.

Lease Consumption

- a). **Editing an alert only:** If you want to enable or disable the alert, select or deselect the **Enabled** option.



Note: When you create an alert, the alert is enabled automatically.

- b). For each intended recipient of the alert, type the recipient's email address into the text-entry field and click the plus-sign (+) next to the field.

Ensure that the email address corresponds to a registered contact in your organization.

The address is added to the list of addresses in the **E-mail addresses to notify** field.

- c). In the **Notify when feature lease consumption exceeds percentage** text-entry field, specify the percentage of available licenses consumed for a specific product that must be exceeded for NLS to send an alert.

Event Type

Steps

- d). In the **Ignore additional alerts occurring within the next hours** text-entry field, specify the number of hours after an alert is sent during which any further consumption of available licenses is ignored.

Failover Event

- a). **Editing an alert only:** If you want to enable or disable the alert, select or deselect the **Enabled** option.



Note: When you create an alert, the alert is enabled automatically.

- b). For each intended recipient of the alert, type the recipient's email address into the text-entry field and click the plus-sign (+) next to the field.

Ensure that the email address corresponds to a registered contact in your organization.

The address is added to the list of addresses in the **E-mail addresses to notify** field.

Service Availability

- a). **Editing an alert only:** If you want to enable or disable the alert, select or deselect the **Enabled** option.



Note: When you create an alert, the alert is enabled automatically.

- b). For each intended recipient of the alert, type the recipient's email address into the text-entry field and click the plus-sign (+) next to the field.

Ensure that the email address corresponds to a registered contact in your organization.

Event Type

Steps

The address is added to the list of addresses in the **E-mail addresses to notify** field.

- c). In the **Ignore additional alerts occurring within the next minutes** text-entry field, type the number of minutes after an alert is sent during which any further inactive service is ignored.

System Resources

- a). **Editing an alert only:** If you want to enable or disable the alert, select or deselect the **Enabled** option.



Note: When you create an alert, the alert is enabled automatically.

- b). For each intended recipient of the alert, type the recipient's email address into the text-entry field and click the plus-sign (+) next to the field.

Ensure that the email address corresponds to a registered contact in your organization.

The address is added to the list of addresses in the **E-mail addresses to notify** field.

- c). In the **Notify when system resource usage exceeds percentage** text-entry field, specify the percentage of CPU and memory usage that must be exceeded for NLS to send an alert.

To specify the disk space threshold, refer to [Disk Space Threshold Configuration](#).

- d). In the **Ignore additional alerts occurring within the next minutes** text-entry field, type the number of minutes after an alert is sent during which any further system

Event Type**Steps**

resource usage exceeding the monitoring threshold is ignored.

When you receive an alert, renew the affected resources before they expire or are exhausted. The affected resources and how to renew them depend on the type of the licensing event that caused NLS to send the alert:

- ▶ **API key expiration:** Renew the API key.
- ▶ **License expiration:** Renew your existing entitlements or purchase new entitlements.
- ▶ **Lease consumption:** If additional licenses are available, add more licenses to the license pool from which the licenses are consumed. Otherwise, plan to purchase more licenses and add the purchase licenses to the license pool.
- ▶ **Service availability:** Check the **DLS Service Instance** tab whether the status indicates ACTIVE. If it is not, restart the service.
- ▶ **System resources:**
 - ▶ To expand CPU and RAM on the DLS appliance, increase the number of vCPUs and the total RAM on your hypervisor.
 - ▶ To increase the hard disk storage on the DLS appliance, follow the instructions in [Expanding the Disk Space on a DLS Virtual Appliance](#). If the disk usage reaches 65%, the audit events will be purged by the system. Use the export event feature to retain the events.
- ▶ **Failover event:** When the primary DLS instance fails, an email alert is sent to the user indicating the cause of failover. When the secondary DLS instance becomes the primary instance and begins to serve licenses, an email alert is sent confirming a successful failover event.

6.5.1. Disk Space Threshold Configuration

By default, the disk space threshold is set to 65% for a DLS virtual appliance. When the disk usage exceeds the 65% threshold, the system automatically initiates a process to remove all the events and the old registration data. In addition, an email alert will notify you when the disk space threshold exceeds 60%, which is 5% lower than the default threshold set on the DLS appliance.

To adjust the default threshold value, specify a different percentage value by modifying the `PURGE_DISK_THRESHOLD` environment variable in the deployment file and then restart the application.

6.6. Setting the Retention Period of Events on a DLS Instance

A DLS instance records events related to administration of the instance and the serving of licenses from the instance to licensed clients. The events are captured in real-time and displayed on the **Events** page of the instance within 5 seconds. You can control the number of events that are displayed on this page by setting the retention period of events on a DLS instance. Any event older than the retention period is deleted from the instance.

1. Open a web browser and connect to the URL `https://dls-vm-ip-address`.

dls-vm-ip-address

The IP address or, if defined, the fully qualified domain name or the CNAME of the VM on which the DLS virtual appliance is installed.

You can get the IP address from the management console of your hypervisor.

2. On the login page that opens, provide the user credentials for the DLS administrator user on the DLS virtual appliance and click **LOGIN**.
3. In the left navigation pane, click **SETTINGS**.
4. On the **Service Instance Settings** page that opens, expand the **Basic Settings** section and click the **Events** tab.
5. In the **Events** tab, set the retention time in days for each type of event, and click **SAVE SETTINGS**.

6.7. Configuring a DLS Virtual Appliance with a Third-Party Signed SSL Certificate

By default, a DLS virtual appliance is configured with a self-signed SSL certificate that is included in the DLS virtual appliance image from which the DLS virtual appliance is created. If necessary, you can replace the self-signed certificate with an SSL certificate that is signed by a third party, such as a certificate authority (CA).

To configure a DLS virtual appliance with a third-party signed SSL certificate, follow this sequence of instructions:

1. [Obtaining a Third-Party Signed SSL Certificate for a DLS Virtual Appliance](#)
2. [Installing a Third-Party Signed SSL Certificate on a DLS Virtual Appliance](#)

6.7.1. Obtaining a Third-Party Signed SSL Certificate for a DLS Virtual Appliance

Obtain a third-party signed SSL certificate by submitting a certificate signing request (CSR) to a suitable third party, such as a certificate authority (CA).

Ensure that the IP address of any DLS virtual appliance that will be configured with the certificate is mapped to the domain name that you will specify in the certificate.

For an HA cluster of DLS instances, you can choose to obtain a single wildcard domain certificate for all nodes in the cluster or one fully qualified domain name certificate for each node in the cluster.

For each certificate that you require, submit a certificate signing request (CSR) to a CA.

Ensure that each certificate that you request meets these requirements:

- ▶ The certificate must be a PEM text file (**not** in Java keystore format) and secured with a private key.
- ▶ The certificate and the private key must be in separate files.
- ▶ To ensure that web browsers trust the domain, the domain name must be part of the Subject Alternate Name (SAN) attribute, **not** the Common Name (CN) attribute of the CSR.
- ▶ The SAN attribute of the CSR must specify the fully qualified domain name of any DLS virtual appliance that will be configured with the certificate.

Do **not** specify an IP address in the SAN attribute of the CSR. A DLS virtual appliance cannot be configured with an SSL certificate that specifies an IP address.

- ▶ The certificate must use RSA, DSA, and DH keys that are at least 2048 bits long.
- ▶ The certificate must use ECC keys greater that are longer than 224 bits long.
- ▶ The certificate must not be passphrase protected.
- ▶ If the certificate chain of trust includes intermediate certificates, the certificate must be bundled with the intermediate certificates in the following order:
 1. Domain name certificate
 2. Intermediate certificates
 3. Root certificate

If necessary, contact the CA that will provide your certificate for information about how to request a certificate that meets these requirements or convert an existing certificate to meet these requirements.

6.7.2. Installing a Third-Party Signed SSL Certificate on a DLS Virtual Appliance

Ensure that the following prerequisites are met:

- ▶ You have obtained the SSL certificate that you are installing and its private key file.
- ▶ Ensure that a license server is installed on the service instance as explained in [Installing a License Server on a Service Instance](#).

If you are installing a wildcard domain certificate for all nodes in an HA cluster, perform this task from the primary node in the cluster **only**. The certificate is propagated automatically to the secondary node in the cluster. If you are installing one fully qualified domain name certificate for each node in the cluster, perform this task separately from each node.

1. Log in to the DLS virtual appliance on which you are installing the SSL certificate.
2. In the left navigation pane, click **SETTINGS**.
3. On the **Service Instance Settings** page that opens, expand the **SSL Configuration** section.
4. In the **SSL Configuration** section, specify the SSL certificate that you are installing and its private key file.
 - a). If you are installing a wildcard domain certificate for all nodes in an HA cluster, set the **Apply Wildcard** option.
 - b). Ensure that the **Domain Name** field contains the domain name that is specified in the certificate.



Note: The **Domain Name** field is case-sensitive. The case of the name in this field must match **exactly** the case of the name as specified in the certificate.

- c). Click **Choose File** adjacent to **Certificate** and in the file browser that opens, navigate to the folder that contains the SSL certificate and select the file.
 - d). Click **Choose File** adjacent to **Private Key** and in the file browser that opens, navigate to the folder that contains the SSL certificate's private key and select the file.
5. Click **CONFIGURE**.

6.8. Hiding Organization and Virtual Group Information on a DLS Instance

You can use REST APIs to hide the organization and virtual group to which the licenses served by a DLS instance belong.

1. Submit an HTTP `POST` request with basic authentication to log in to the DLS appliance that hosts the DLS instance as the DLS administrator user.

```
POST https://host:port/auth/v1/login
```

host

The IP address or fully qualified domain name of the VM or container that is hosting the DLS appliance.

port

The port on which the VM or container that is hosting the DLS appliance listens for HTTPS requests.

In the request, pass the user name and password of the DLS administrator user. The default user name is `dls_admin`.

The following example uses the `curl` command to log in to the DLS appliance `dls.example.com` on port 8081 as the user `dls_admin` with the password `changeit`.

```
curl -k --location --request POST 'https://dls.example.com:8081/auth/v1/login' \
--header 'Content-Type: application/json' \
--data-raw '{
  "username": "dls_admin"
  "password": "changeit"}'
```

The response sets the authorization token in the `LIC-TOKEN` cookie and includes the token in the response body.

2. Submit an HTTP `PUT` request to hide the organization and virtual group on the DLS instance.

```
PUT https://host/service_instance_manager/v1/config/hide-name-on-ui
```

host

The IP address or fully qualified domain name of the VM or container that is hosting the DLS appliance.

If the tool that you use to submit the request supports cookies, you can submit the request without the authorization token. Otherwise, you must pass the authorization token that was returned in the previous step in the header as a bearer token.

The following example uses the `curl` command to hide the organization and virtual group on the DLS instance on the appliance `dls.example.com`. For clarity, the example shows only the placeholder `token`.

```
curl -k --location --request PUT \
' https://dls.example.com/service_instance_manager/v1/config/hide-name-on-ui' \
--header 'Authorization: Bearer token'
```

If the organization and virtual group are hidden after the DLS administrator user has logged in to the DLS appliance, the DLS administrator user must reload the web-based management interface to see the changes.

To show the organization and virtual group again, submit a `PUT` request on the REST endpoint for hiding the organization and virtual group with the query parameter `hide=false`.

```
PUT https://host/service_instance_manager/v1/config/hide-name-on-ui?hide=false
```

6.9. Reconfiguring the Rsyslog Tool in a DLS Virtual Appliance

A VM-based DLS virtual appliance uses the Rsyslog tool for forwarding log messages in an IP network. If you want to export events and alerts for the DLS instance on the

appliance to an external `syslog` or security information and event management (SIEM) server, you can reconfigure the Rsyslog tool in the appliance.



Note: You can perform this task **only** on a VM-based DLS virtual appliance. You **cannot** perform this task on a containerized DLS software image. Instead, you can use standard interfaces of the OS on which the container orchestration platform is running to make this change.

The format of each type of log message from a DLS virtual appliance is as follows:

- ▶ `syslog` messages are in the standard `syslog` format, with the fields in each message separated by a space:

```
timestamp hostname program-name message-text
```

- ▶ NLS service messages are formatted as JSON strings:

```
SEMANTIC:service-name: semantic-json-string
```



Note: Common Event Format (CEF) is **not** supported. However, you can configure the format or transfer the data to the SIEM server after the logs have been forwarded to the `syslog` server.

1. Use the hypervisor management console of the appliance to log in as the user **dls_diagnostics** to the VM that hosts the DLS virtual appliance.
2. Edit the file `/etc/rsyslog.conf` to update the address and port of the external `syslog` server.
3. Restart the Rsyslog tool.

```
# sudo -u root /etc/adminscripts/restart_rsyslog.sh
```

6.10. Configuring NTP on a DLS Virtual Appliance

Network Time Protocol (NTP) Configuration is when the customer needs to sync the clock of their License Server to a desired time-keeping server.



Note: You can perform this task **only** on a VM-based DLS virtual appliance. You **cannot** perform this task on a containerized DLS software image. Instead, you can use standard interfaces of the OS on which the container orchestration platform is running to make this change.

Ensure that a license server is installed on the service instance as explained in [Installing a License Server on a Service Instance](#).

1. Open a web browser and connect to the URL `https://dls-vm-ip-address`.

dls-vm-ip-address

The IP address or, if defined, the fully qualified domain name or the CNAME of the VM on which the DLS virtual appliance is installed.

You can get the IP address from the management console of your hypervisor.

2. On the login page that opens, provide the user credentials for the DLS administrator user on the DLS virtual appliance and click **LOGIN**.
3. In the left navigation pane, click **SETTINGS**.
4. On the **Service Instance Settings** page that opens, expand the **NTP Server Configuration** section.
5. In the **NTP Server Configuration** section, enter the IP address or domain name of one or more NTP servers, with the addresses or names of multiple servers separated by commas.
6. Click **PING SERVER** to check the connectivity with the NTP Server on port 123 and to validate whether the IP address or domain name entered is a valid NTP server. If pinging the server is successful, the **CONFIGURE** button turns green.
7. Click **CONFIGURE**.
A green check mark next to the configured server indicates that the time synchronization with NTP is complete.

6.11. Setting the Maximum Size of the PostgreSQL Database Volume

To limit disk usage by a container-based DLS appliance, you can set the maximum size of the PostgreSQL database volume `postgres-data`. The default maximum size is 15 GB.



Note: You can perform this task **only** on a containerized DLS software appliance. You **cannot** perform this task on a VM-based DLS virtual appliance.

1. Log in to the DLS appliance for which you want to set the maximum size of the PostgreSQL database volume.
2. In the left navigation pane of the **NVIDIA Licensing** dashboard, click **SETTINGS**.
3. On the **Service Instance Settings** page that opens, expand the **Basic Settings** section and click the **Events** tab.
4. In the **Purge Settings** section, under **Container Volume Disk Allocation in GB**, type the maximum size of the PostgreSQL database volume in GB and click **SAVE SETTINGS**.

6.12. Troubleshooting a DLS Instance

To facilitate troubleshooting, the DLS provides access to log files, event records, and the status of a DLS instance's internal services. If the DLS instance's internal services have

failed, you can restart them from the **NVIDIA Licensing** application on the DLS virtual appliance.

6.12.1. Log File Locations and Types for a DLS Virtual Appliance

The log files for a DLS virtual appliance contain diagnostic information to help with troubleshooting. The DLS administrator user can access these log files through the hypervisor console or secure shell (SSH).

The log files for a DLS virtual appliance are in the locations in the following table.

Log Message Type	Log File Location
NVIDIA License System licensing messages	<code>/var/lib/docker/volumes/logs/_data/licensing</code>
NVIDIA License System service messages	<code>/var/lib/docker/volumes/logs/_data/op_log_archive</code>
	<code>/var/lib/docker/volumes/logs/_data/op_log_capture</code>
	<code>/var/lib/docker/volumes/logs/_data/op_log_ingest</code>
DLS web server messages	<code>/var/lib/docker/volumes/logs/_data/nginx</code>
DLS virtual appliance internal messages related to: <ul style="list-style-type: none"> ▶ HA configuration ▶ Inconsistent data between the primary and secondary nodes in an HA cluster ▶ Online migration of a DLS instance 	<code>/var/lib/docker/volumes/logs/_data/rabbitmq_stdout.log</code>

Files in the following standard Linux directories contain log messages from the operating system:

- ▶ `/tmp`
- ▶ `/var/log`
- ▶ `/var/tmp`

6.12.2. Storing Log Files for a DLS Virtual Appliance on a Network File Share

Because the amount of disk space available in a DLS virtual appliance might be limited, the DLS virtual appliance does not retain all log messages generated during the lifetime of a DLS virtual appliance. If you want to retain all log messages, you can configure the virtual appliance to store log files for a DLS virtual appliance on a network file share.

If a virtual appliance is configured to store log files for a DLS virtual appliance on a network file share, it periodically aggregates the log files and moves them from the local disk of the DLS virtual appliance to the share.

Ensure that the following prerequisites are met:

- ▶ The network file share has been created on a network attached storage server that is accessible from the DLS virtual appliance.
- ▶ The DLS administrator user on the DLS virtual appliance is granted full access to the network file share.

1. Log in to the DLS virtual appliance.
2. In the left navigation pane, click **SETTINGS**.
3. On the **Service Instance Settings** page that opens, expand the **Log Archival** section.
4. In the **Log Archival** section, provide the details of the network file share and click **MOUNT**.
 - a). From the **Platform** drop-down list, select the OS that corresponds to the type of the share on the network attached storage server.
 - ▶ For a share that corresponds to the Windows OS, such as a CIFS share, select **Windows**.
 - ▶ For a share that corresponds to the UNIX OS, such as an NFS share, select **Unix**.
 - b). In the **Network Share Path** text-entry field, type the path to the share on the network attached storage server in the following format:


```
//nas-server/file-path
```

nas-server
The IP address or fully qualified domain name of the network attached storage server on which the share has been created.

file-path
The full path to the share from the root of the file system on the network attached storage server. In the path, use the forward slash as the file separator, even if you selected **Windows** from the **Platform** drop-down list.
 - c). If you selected **Windows** from the **Platform** drop-down list, provide the user name and password for the user on the network attached storage server that will access the share.

The network file share is mounted at `/var/log/licensing` on the DLS virtual appliance. This mount point is preset in the DLS virtual appliance and cannot be changed.

6.12.3. Exporting and Importing Event Records from a DLS Instance

To help NVIDIA Enterprise Support personnel troubleshoot issues with a DLS instance, you can export event records from the DLS instance and import them into the NVIDIA Licensing Portal.



Note: The maximum size of a file that can be exported from a DLS instance or imported into the NVIDIA Licensing Portal is 250 MBytes.



Note: If no filters are selected, records will be exported for the last 7 days.

1. Log in to the DLS virtual appliance that hosts the DLS instance.
2. In the left navigation pane of the **NVIDIA Licensing** dashboard, click **MAINTENANCE**.
3. **Optional:** Filter the events that you want to export.
 - a). On the **Maintenance** page that opens, set the **Show advanced filters** option.
 - b). Use the calendar widget that is added to the page to set the range of dates for which you want to export event records.
 - c). Select the categories of event that you want to export.
4. On the **Maintenance** page, click **EXPORT EVENTS**.
An event log file that is named `on-premises_export_mm-dd-yyyy-hh-mm-ss.log` is saved to your downloads folder.
5. In the NVIDIA Licensing Portal, navigate to the organization or virtual group to which the DLS instance belongs.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, select the virtual group to which the DLS instance belongs from the list of virtual groups at the top right of the page.
6. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **DASHBOARD**.
7. On the **Dashboard** page that opens, under **DLS data synchronization**, click **Import**.
8. In the file browser that opens, navigate to the folder that contains the event log file that is named `on-premises_export_mm-dd-yyyy-hh-mm-ss.log` that you downloaded and select the file.
9. **Optional:** For each other event file that you want to import, click **SELECT ANOTHER FILE**, and navigate to and select the file.
10. After selecting all the files that you want to import, click **UPLOAD EVENTS**.

6.12.4. Restarting a DLS Instance's Internal Services

If a DLS instance has failed because its internal services are no longer active, you can restart the inactive services to recover from the failure.

1. Log in to the DLS virtual appliance that hosts the DLS instance.
If the DLS instance is a node in an HA cluster, you must log in to the DLS virtual appliance that hosts the specific node. You **cannot** restart the internal services of the secondary node from the primary node or restart the internal services of the primary node from the secondary node.
2. In the left navigation pane, click **SERVICE INSTANCE**.
3. On the **Service Instance** page that opens, under **Node Health**, determine whether the status of any of the DLS instance's internal services is inactive.
Status information is provided for the DLS instance's critical services and other services.
4. If any set of services is not active, click **RESTART** for that set of services.

6.12.5. Recovering from an Incorrect DLS Name or Address in a Client Configuration Token

A client configuration token uses an IP address or fully qualified domain name to identify the DLS instance from which to serve a license to a client. If the IP address or fully qualified domain name in the client configuration token is incorrect, licensing requests from all clients that are configured with the token fail to be fulfilled.

When this issue occurs, an error message similar to the following example is written to the licensing event log on the client.

```
Failed to acquire license from 192.0.2.206
(Info: NVIDIA RTX Virtual Workstation -
Error: Could not communicate to server, timeout has been reached)
```

To determine whether an incorrect DLS name or address caused a licensing request to fail, compare the IP address or fully qualified domain name in the error message with the same information on the **Service Instance Details** page. If the information in both places does not match, the failure of the licensing request was caused by an incorrect DLS name or address.

Recover from this issue in one of the following ways:

- Replace the client configuration token with a new token in which the IP address or fully qualified domain name is correct.
 1. Generate a new client configuration token as explained in [Generating a Client Configuration Token for a DLS Instance](#) to ensure that the IP address or fully qualified domain name in the token is correct.

2. Configure each licensed client with the new token as explained in [Configuring a Licensed Client with a Networked License](#).
- ▶ Change the IP address or fully qualified domain name of the VM on which the DLS instance is hosted to match the IP address or fully qualified domain name in the client configuration token.

6.12.6. Recovering from an HA Configuration Failure in an IPv6 Network

An attempt to create an HA cluster of VM-based DLS instances in which the VMs are communicating over IPv6 might fail. The failure occurs because the DLS virtual appliance that is hosting one of the instances cannot ping the DLS virtual appliance that is hosting the other instance. To recover from this failure, use the shell script that each DLS virtual appliance provides specifically for this purpose.



Note: You can perform this task **only** on a VM-based DLS virtual appliance. You **cannot** perform this task on a containerized DLS software image. Instead, you can use standard interfaces of the OS on which the container orchestration platform is running to make this change.

Perform this task for each VM on which the DLS virtual appliance for each DLS instance in the cluster is installed.

1. Use the hypervisor management console of the appliance to log in as the user **dls_admin** to the VM that hosts the DLS virtual appliance.
If the **dls_admin** user has not been registered, you can still log in to the VM as the **dls_admin** user with the default password **welcome**.
2. As root, run the `/etc/dls/scripts/support_only_ipv6.sh` script.

```
$ sudo -u root /etc/dls/scripts/support_only_ipv6.sh
```
3. Repeat the attempt to create the cluster by following the instructions in [Creating or Expanding an HA Cluster of DLS Instances](#).

6.12.7. DLS has overlapping disk partitions

During the backup process you may get a partition overlaps the last partition warning message. Use these steps to resolve the warning.

1. Log in as the `rsu_admin` user and execute this command:


```
**sudo sed -i 's/100/99/g' /etc/dls/scripts/expand_disk.sh**
```
2. Increase the VM disk space by 1 GB or more using the steps in [Expanding the Disk Space on a DLS Virtual Appliance](#)
3. **Optional:** The warning should now be resolved. Execute this command:


```
**`sudo gdisk /dev/sda`**
```

 as the `rsu_admin` user to confirm.

Chapter 7. Managing License Servers on the NVIDIA Licensing Portal

After creating a license server on the NVIDIA Licensing Portal, you can add licenses to and remove licenses from the server, add new licensed products to and remove licensed products from the server, and delete the server when you no longer require it.

7.1. Where to Perform Tasks for Managing a License Server

Where to perform the tasks for managing a license server depends on the task and on the type of service instance on which the license server is installed.

Task	CLS Instance	DLS Instance
Adding licenses to a license server	NVIDIA Licensing Portal	NVIDIA Licensing Portal
Removing licenses from a license server	NVIDIA Licensing Portal	The NVIDIA Licensing application on the virtual appliance that hosts the DLS instance
Adding licensed products to a license server	NVIDIA Licensing Portal	NVIDIA Licensing Portal
Removing licensed products from a license server	NVIDIA Licensing Portal	The NVIDIA Licensing application on the virtual appliance that hosts the DLS instance
Moving a license server to another virtual group	NVIDIA Licensing Portal	NVIDIA Licensing Portal
Deleting a license server	NVIDIA Licensing Portal	NVIDIA Licensing Portal



Note: You **cannot** delete a license server that is bound to and installed on a DLS or CLS instance. You must first free the license server from the CLS or DLS instance. For instructions, refer to [Freeing a License Server from a Service Instance](#).

7.2. Roles Required for Managing License Servers on the NVIDIA Licensing Portal

The role that these tasks require depends on whether they are being performed for an organization or a virtual group.

- ▶ For an organization, these tasks require the [Organization Administrator](#) or the [Organization User](#) role.
- ▶ For a virtual group, these tasks require the [Virtual Group Administrator](#) or the [Virtual Group User](#) role.

7.3. Managing Licenses and Licensed Products on a License Server

Manage the licenses on a license server to add or remove individual licenses for a specific product on the license server. Manage the licensed products on a license server if you need to add or remove licensed products from a license server. When you add a licensed product to a license server, only one license for the product is allocated. When a licensed product is removed from a license server, all licenses that were allocated for the product are returned to the entitlement.

Ensure that any licensed products that you want to remove from a license server are not in a license pool. If necessary, return licensed products from a pool to the license server as explained in [Managing Licenses and Licensed Products in a License Pool](#).

Where to perform this task depends on whether you are adding or removing licenses.

- ▶ If you are adding licenses, perform this task on the NVIDIA Licensing Portal irrespective on the type of service instance on which the license server is installed.
- ▶ If you are removing licenses, where to perform this task depends on the type of service instance on which the license server is installed:
 - ▶ On a CLS instance, perform this task on the NVIDIA Licensing Portal.
 - ▶ On a DLS instance, perform this task on the **NVIDIA Licensing** application on the virtual appliance that hosts the DLS instance.

1. Navigate to the **License Server Details** page of the license server to which you want to add or from which you want to remove licenses.

For instructions, see [Navigating to the License Server Details Page for a License Server](#).

2. In the **License Server Details** page that opens, disable the license server by clicking **DISABLE SERVER** and, when prompted, confirm that you want to disable the license server.

When the license server is disabled, it cannot serve licenses to licensed clients.

3. From the **ACTIONS** menu, choose **Manage Features**.
The **Manage Server Features** pop-up window opens with the **Modify existing** tab active.
 1. In the **Manage Server Features** pop-up window, you specify all the licenses and licensed products that you want to add or remove before confirming your changes by clicking **UPDATE SERVER FEATURES**.
 4. For each licensed product for which you want to add or remove licenses, specify the number of licenses for the product that you want to **remain on the server after updating licenses**.

Use the text-entry field in the **ALLOCATED** column for this purpose.

- ▶ To **add** licenses to the server, enter a number **greater** than the number already allocated to the server, but less than or equal to the total number of licenses available.

If you enter a number greater than the total number of licenses available, an error occurs.

- ▶ To **remove** licenses from the server, enter a number **less** than the number already allocated to the server but greater than 0.

For example, to remove 4 licenses from a server to which 10 licenses are allocated, leaving 6 licenses allocated to the server, enter **6** in the **Licenses** field.

If you enter **0**, an error occurs. You must leave at least **one** license on the license server. To remove all licenses for a product from the license server, remove the product from the server by clicking the trash can icon.

The number of licenses on the server remains unchanged until you confirm your changes by clicking **UPDATE SERVER FEATURES**.

5. Remove each licensed product that you no longer want on the server by clicking the trash can icon next to the licensed product.
The licensed product is removed from the list of licensed products on the **Modify existing** tab. However, the licensed product remains on the server until you confirm the change by clicking **UPDATE SERVER FEATURES**.
6. Add any licensed products that you want to add to the license server.
 - a). Click the **Add new** tab.
 - b). In the list of licensed products on the **Add new** tab, select the licensed products that you want to add.
7. After adding or removing all the licenses and licensed products that you are interested in, preview your changes by clicking **Preview changes**.
8. After previewing your changes, click **UPDATE SERVER FEATURES**.
9. On the Overview tab of the License Server Details page, enable the license server by clicking **ENABLE SERVER** and, when prompted, confirm that you want to enable the license server.
The license server can now serve licenses to licensed clients.

If the license server is installed on a CLS instance, no further action is required. The license server and the NVIDIA Licensing Portal are automatically updated with your changes.

If the license server is installed on a DLS instance, you must ensure that the licenses on your license server and on the NVIDIA Licensing Portal are consistent.

- ▶ If you added licenses, download and install the updated license server to ensure that the correct licenses are available on the DLS instance. For detailed instructions, see [Installing a License Server on a DLS Instance](#). Installing the updated license server does not affect the distribution of existing licenses among license pools.
- ▶ If you removed licenses, return the licenses to the entitlement on the NVIDIA Licensing Portal. For detailed instructions, see [Returning Licenses from a License Server on a DLS Instance to the NVIDIA Licensing Portal](#).
- ▶ If you need to add or remove licenses for a specific product in the pool, see [Managing Licenses and Licensed Products in a License Pool](#).

7.4. Returning Licenses from a License Server on a DLS Instance to the NVIDIA Licensing Portal

After removing individual licenses or licensed products from a license server installed on a DLS instance, you must return them to the entitlement on the NVIDIA Licensing Portal. Returning them ensures that the licenses and products on your license server and on the NVIDIA Licensing Portal are consistent. The returned licenses and products are then available for use by other license servers.

1. If you are not already logged in, log in to the **NVIDIA Licensing** application at the IP address of the VM on which the DLS virtual appliance is installed.
2. In the left navigation pane of the **NVIDIA Licensing** dashboard, click **MAINTENANCE**.
3. On the **Maintenance** page that opens, click **Export Feature Return**.



Note: The **Export Feature Return** button is active only if individual licenses or licensed products have been removed from the license server.

- A license return file named `on-premises_feature_return_mm-dd-yyyy-hh-mm-ss.bin` is downloaded.
4. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which the license server was created.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, select the virtual group for which the license server was created from the list of virtual groups at the top right of the page.

5. In the list of license servers on the NVIDIA Licensing Portal dashboard, select the license server from which you want to return licenses or licensed products.
6. In the **License Server Details** page that opens, from the **Actions** menu, choose **Return Features**.
7. In the Return Features pop-up window that opens, click **SELECT FEATURE RETURN FILE**.
8. In the file browser that opens, navigate to the folder that contains the license return file named `on-premises_feature_return_mm-dd-yyyy-hh-mm-ss.bin` that you downloaded and select the file.
9. Back in the Return Features pop-up window, click **RETURN FEATURES**.
The returned licenses and license products are added to the entitlements on the NVIDIA Licensing Portal.

7.5. Moving a License Server to Another Virtual Group

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group to which the **license server** belongs.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, expand **LICENSE SERVERS** and click **LIST SERVERS**.
3. In the list of license servers on the **License Servers** page that opens, from the **Actions** menu for the license server, choose **Move server**.
The **Move License Server** pop-up window opens and displays which licensed products can be moved and which licensed products cannot be moved because they are bound to a DLS instance or because licenses are checked out.
4. In the **Move License Server** pop-up window, select the virtual group to which you want to move the license server and click **MOVE SERVER**.
The licensed products on the license server that can be moved are moved to the selected virtual group.

7.6. Deleting a License Server



Note: You cannot delete a license server that is bound to and installed on a DLS or CLS instance. You must first free the license server from the CLS or DLS instance. For instructions, refer to [Freeing a License Server from a Service Instance](#).

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which you want to delete the license server.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the My Info window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the list of license servers on the License Servers page that opens, click the name of the license server that you want to delete.
3. In the **License Server Details** page that opens, from the **Actions** menu, choose **Delete**.
4. When asked to confirm that you want to delete the license server, click **DELETE LICENSE SERVER**.

7.7. Editing Default Service Instances

A Default Service Instance is a Service Instance which has been designated as the Service Instance to use in the event of an Express CLS Installation. If no CLS Instance has been used for Express CLS Installation previously, one will be created during the first Express CLS Installation Process. NVIDIA License System Service Instance.

Only CLS-Bound Service Instances can be designated as **Default**.

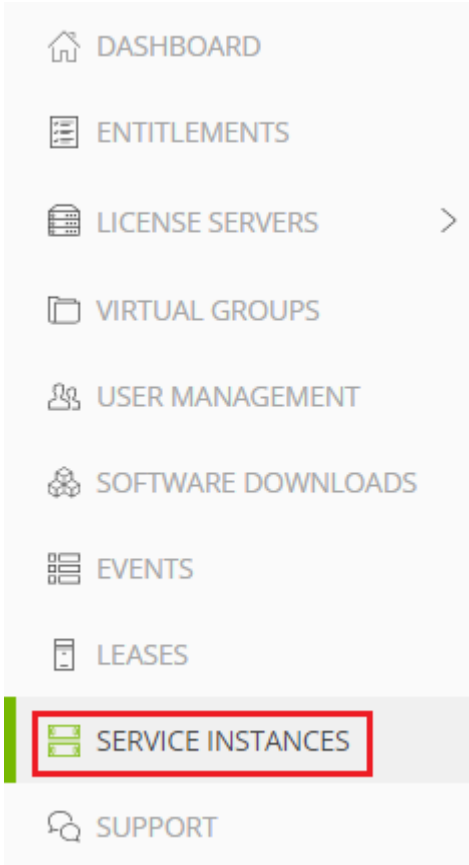
For more information on Express CLS Installation, see [Configuring a Service Instance](#).

To view which Service Instance has been designated as **Default**:

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which you want to create the license server.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the My Info window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.

If no license servers have been created for your organization or virtual group, the NVIDIA Licensing Portal dashboard displays a message asking if you want to create a license server.

2. From the left-side navigation pane, choose **SERVICE INSTANCES**.



- From the **Table** view, under the **Environment** column, look for the green **DEFAULT** indicator.

Service Instances [Help?](#) ACTIONS

View your service instances in

CLS DLS

Search service instances updated 12:43:41 PM

NAME	ENVIRONMENT	STATUS	DATE CREATED	Actions
0011w0000275iqay-2022-02-03_19-03	CLS	Registered	Feb 3, 2022 11:03 AM	Actions
DEFAULT-2022-01-31_21:05:34	DLS	Registered	Jan 31, 2022 4:45 PM	Actions
Example_CLS	CLS Default	Registered	Jan 20, 2022 3:38 PM	Actions
Example_DLS	DLS	Registered	Jan 20, 2022 3:35 PM	Actions
	CLS	Registered	Dec 15, 2021 11:51 AM	Actions

10 service instances per page (1 - 10 of 34 service instances) 1 of 4 pages

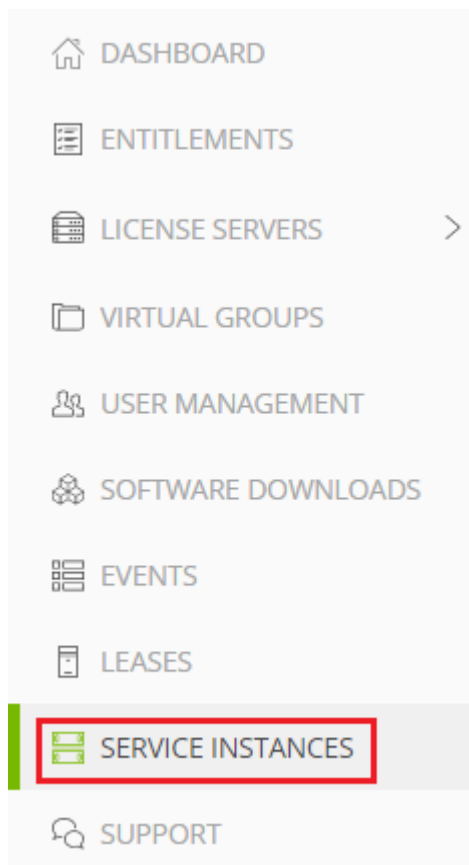
7.7.1. Edit the Service Instance Designated as Default

If you would like to change which Service Instance is bound as the Default, follow these steps.

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which you want to create the license server.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the My Info window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.

If no license servers have been created for your organization or virtual group, the NVIDIA Licensing Portal dashboard displays a message asking if you want to create a license server.

2. From the left-side navigation pane, choose **SERVICE INSTANCES**.



3. From the **Table** view, find the CLS Service Instance that you would like to bind as the Default, or create a new one: [Configuring a Service Instance](#).
4. From the **ACTIONS** menu, select **EDIT**.
5. On the dialog that pops up, select the toggle button to **Mark as default**.
6. Click **EDIT SERVICE INSTANCE**.

Chapter 8. Managing Contacts on the NVIDIA Licensing Portal

To help you manage your entitlements and licenses on the NVIDIA Licensing Portal, you can add other users as registered contacts in the organization associated with your NVIDIA Enterprise Account. You can also remove users who no longer require access from your account on the NVIDIA Licensing Portal.

To secure your entitlements and licenses, NVIDIA Licensing Portal provides role-based access for all registered contacts. Each role has a scope that determines whether the role applies to a virtual group within an organization or the organization itself. For more information, see [Role-Based Access to an Organization and Virtual Groups](#).

8.1. Role-Based Access to an Organization and Virtual Groups

Role-based access helps secure the entitlements and licenses in your organization on the NVIDIA Licensing Portal. If you partition your entitlements into isolated segments, role-based access also provides isolation between the segments into which your entitlements are partitioned. It does so by ensuring that only specific contacts in your organization are allowed to view or perform actions on the entitlements and contacts that are allocated to a virtual group.

A role is a collection of actions or capabilities within the NVIDIA Licensing Portal. Each role has a scope that determines the context to which the actions and capabilities of the role apply, specifically, a virtual group within an organization or the organization itself.

Every registered contact has at least one role, but can have multiple roles if the scope of each role is a virtual group. As a result, a contact can be a member of multiple virtual groups. However, roles with a virtual group scope and roles with an organization scope are mutually exclusive. A contact that has a virtual group role cannot also have an organization role.

To enable role-based access to an organization and virtual groups, the NVIDIA Licensing Portal provides pre-defined roles.

8.1.1. Organization Administrator

An organization administrator has the highest level of visibility and access within an organization. The person that created the organization's NVIDIA Enterprise Account is initially assigned the organization administrator role.

Each organization must have at least one organization administrator. Multiple organization administrators in an organization are allowed. To prevent the absence of a single user from denying you access to your organization's entitlements, consider adding at least two organization administrators to your organization.

An organization administrator can see all of the following items for the organization on the NVIDIA Licensing Portal:

- ▶ Entitlements
- ▶ Users
- ▶ Virtual groups
- ▶ License servers provisioned from the entitlements that have not been assigned to a virtual group

An organization administrator can manage virtual groups as follows:

- ▶ Create a virtual group.
- ▶ Delete a virtual group.
- ▶ Assign an entitlement at the organization level to a virtual group.
- ▶ Remove an entitlement from a virtual group and return it to the organization.

An organization administrator can manage other contacts in the organization as follows:

- ▶ Invite a contact currently not within the organization to register at the organization level.
- ▶ Add users and administrators to a virtual group when creating the virtual group.
- ▶ Delete any administrator or user at either the organization level or the virtual group level **except** the last virtual group administrator in a virtual group.
- ▶ Manage the role of any organization-level contact.

An organization administrator also has all the capabilities of an organization user.

8.1.2. Organization User

An organization user has restricted visibility within an organization and can access only items that are at the organization level and that have not been assigned to a virtual group. An organization user is a contact that has been added to the organization as a base user.

An organization can have no organization users, only one organization user, or multiple organization users.

An organization user can see the following items for the organization on the NVIDIA Licensing Portal:

- ▶ All organization administrators
- ▶ All other organization users
- ▶ All organization-level entitlements
- ▶ All license servers provisioned from organization-level entitlements

An organization user can manage entitlements that have not been assigned to a virtual group as follows:

- ▶ Create a license server.
- ▶ Delete a license server.
- ▶ Add licensed products to a license server.
- ▶ Remove licensed products from a license server.
- ▶ Download a license file.
- ▶ Download software.

An organization user **cannot** manage other contacts.

8.1.3. Virtual Group Administrator

A virtual group administrator has restricted visibility within an organization and can access items and manage contacts only in the virtual group to which the virtual group administrator is assigned. A virtual group administrator is a contact that has been added to the virtual group as an admin user.

Each virtual group must have at least one virtual group administrator. Multiple virtual group administrators in a virtual group are allowed. To prevent the absence of a single user from denying you access to a virtual group, consider adding at least two virtual group administrators to each virtual group in your organization.

A virtual group administrator can see the following items on the NVIDIA Licensing Portal:

- ▶ All organization administrators
- ▶ All other contacts in the virtual group
- ▶ All entitlements assigned to the virtual group
- ▶ All license servers provisioned from entitlements assigned to the virtual group

A virtual group administrator can manage other contacts in the virtual group as follows:

- ▶ Add an existing contact within the organization who is **not** an organization administrator to the virtual group.
- ▶ Invite a contact currently not within the organization to register and join the virtual group.
- ▶ Remove any other contact in the virtual group, regardless of the contact's role.

Virtual group administrators cannot remove themselves from a virtual group.

- ▶ Manage the role of any other contact in the virtual group.

Virtual group administrators cannot manage their own roles.

A virtual group administrator also has all the capabilities of a virtual group user.

8.1.4. Virtual Group User

A virtual group user has no visibility within an organization and can view and access items only in a virtual group. A virtual group user is a contact that has been added to the virtual group as a base user.

A virtual group can have no virtual group users, only one virtual group user, or multiple virtual group users.

A virtual group user can see the following items for the virtual group on the NVIDIA Licensing Portal:

- ▶ All other contacts in the virtual group
- ▶ All entitlements assigned to the virtual group
- ▶ All license servers provisioned from entitlements assigned to the virtual group

A virtual group user can manage entitlements within a virtual group as follows:

- ▶ Create a license server.
- ▶ Delete a license server.
- ▶ Add licensed products to a license server.
- ▶ Remove licensed products from a license server.
- ▶ Download a license file.
- ▶ Download software.

A virtual group user **cannot** manage other contacts.

8.2. Roles Required for Managing Contacts on the NVIDIA Licensing Portal

The role that these tasks require depends on whether they are being performed for an organization or a virtual group.

- ▶ For an organization, these tasks require the [Organization Administrator](#) role.
- ▶ For a virtual group, these tasks require the [Virtual Group Administrator](#) role.

8.3. Adding a Contact on the NVIDIA Licensing Portal

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group to which you want to add a contact.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **USER MANAGEMENT**.
3. In the **USER MANAGEMENT** page that opens, click **INVITE USER**.
4. In the **Invite User** pop-up window that opens, provide the e-mail address and the name of the contact, select the contact's role, and click **SEND INVITATION**.

The role to select depends on whether you are adding the contact to an organization or a virtual group.

- ▶ For an organization, select one of the following roles:
 - ORG ADMIN**
Assigns the contact the [Organization Administrator](#) role.
 - ORG USER**
Assigns the contact the [Organization User](#) role.
- ▶ For a virtual group, select one of the following roles:
 - VIRTUAL GROUP ADMIN**
Assigns the contact the [Virtual Group Administrator](#) role.
 - VIRTUAL GROUP USER**
Assigns the contact the [Virtual Group User](#) role.
- ▶ When a user is added to the NVIDIA Licensing Portal, an email is sent to the organization administrators providing the user and virtual group details.
- ▶ If you added a contact who is already registered to a virtual group, the contact will be able to select the virtual group after next logging in.
- ▶ If the contact is a new contact, an e-mail is sent to the contact at the e-mail address that you provided.



Note: The link to the NVIDIA Enterprise Support Portal in this e-mail provides information about how to contact NVIDIA Enterprise Support.

If the contact that you added is a new contact, tell the contact to follow the directions in the e-mail to create an NVIDIA Enterprise Account.

8.4. Removing a Contact on the NVIDIA Licensing Portal

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group from which you want to remove a contact.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **USER MANAGEMENT**.
3. In the list of contacts in the **User Management** page that opens, from the **Actions** menu, choose **Delete**.



Note: You cannot remove the only virtual group administrator from a virtual group. The **Remove** link for that contact is inactive and dimmed.

4. When prompted, confirm that you want to remove the contact.

The contact is removed from the list of contacts on the **USER MANAGEMENT** page.

- ▶ A contact that is removed from an organization is removed from the list of registered contacts for the organization.
- ▶ A contact that was a member of multiple virtual groups when removed from a virtual group remains a member of the other virtual groups.
- ▶ A contact that was a member of only the virtual group from which you removed the contact is returned to the organization with the organization user role.

8.5. Changing the Role of a Contact on the NVIDIA Licensing Portal

You can change the role of a contact on the NVIDIA Licensing Portal within the scope of the contact's current role. For example, you can change the role from organization user to organization administrator or from virtual group administrator to virtual group user. However, you **cannot** change the scope of the contact's current role, for example, from organization administrator to virtual group user.

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group to which the contact belongs.

- a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). **Optional:** If your assigned roles give you access to multiple virtual groups, click **View settings** at the top right of the page and in the **My Info** window that opens, select the virtual group from the **Virtual Group** drop-down list, and close the **My Info** window.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **USER MANAGEMENT**.
 3. In the list of contacts in the **User Management** page that opens, from the **Actions** menu, select the new role and choose **Change role**.
You are shown the contact's current role and the role to which you can change it.
 4. In the **User Role** window, click the button to change the contact's role.

8.6. Requesting Access to the NVIDIA Enterprise Support Portal

If you created your NVIDIA Enterprise Account when your entitlement contained only evaluation licenses, you do not have access to the NVIDIA Enterprise Support Portal. If you now have an entitlement that contains purchased licenses, you can request access to the NVIDIA Enterprise Support Portal.

Ensure that you have an entitlement that contains purchased licenses. You cannot request access to the NVIDIA Enterprise Support Portal if your entitlements contain only evaluation licenses.

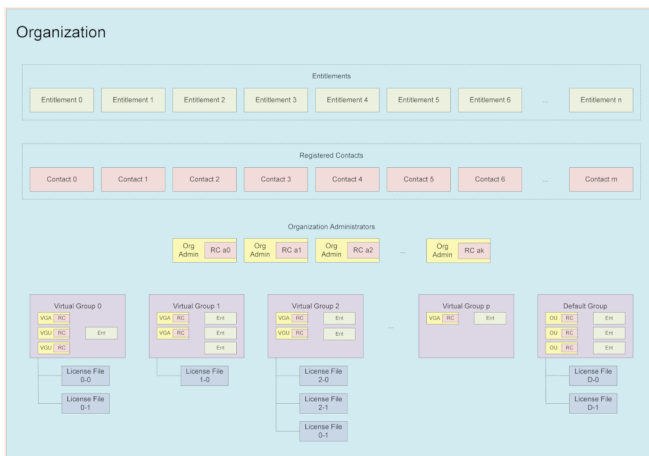
1. If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **USER MANAGEMENT**.
3. On the **User Management** page that opens, click **REQUEST ENTERPRISE SUPPORT PORTAL ACCESS**.

Chapter 9. Managing Virtual Groups

Virtual groups provide the means for segmenting your organization's entitlements into partitions. The virtual groups in an organization are isolated from each other and from the organization. An entitlement cannot be partitioned and cannot be shared between partitions. All licensed products in an entitlement are moved with the entitlement when the entitlement is added to a virtual group or returned to the organization.

You are free to determine how many virtual groups among which to partition your entitlements and what those virtual groups represent. For example, you might create virtual groups to partition your entitlements by location, division, product, or some combination of factors. Irrespective of how you choose to partition your entitlements among virtual groups, every virtual group isolates the entitlements assigned to it from other virtual groups.

The following diagram shows the relationship between an organization, the virtual groups in an organization, and the components of a virtual group.



9.1. Roles for Managing Virtual Groups

These tasks require the [Organization Administrator](#) role.

9.2. Creating a Virtual Group

Ensure that the following prerequisites are met:

- ▶ Your organization contains at least one registered contact to whom you can assign the virtual group administrator role.
- ▶ No licensed products in any of the entitlements that you want to add to the group have been added to a license server.

1. If you are not already logged in, log in to the NVIDIA Licensing Portal.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **VIRTUAL GROUPS**.
3. In the **Virtual Groups** page that opens, click **CREATE VIRTUAL GROUP**.

The **Create Virtual Group** wizard is started.

4. In the **Virtual Group Name** field, enter your choice of name for the group.
5. In the **Description** field, enter a short text description of the group.
This description will be displayed for the group in the list of virtual groups on the **Virtual Groups** page.
6. On the **Select entitlements** page of the wizard, select the entitlements that you want to add to the virtual group and click **Next: Select users**.

For each entitlement that you want to add, select the entitlement from the **Entitlements** drop-down list and click **ADD**.

You must add **at least one** entitlement. You cannot create a virtual group with no entitlements.

Each entitlement that you select is added to the **Added Entitlements** list.

7. On the **Select users** page of the wizard, add the users that you want to be members of the virtual group.

Add each user as follows:

- a). In the list of users, select the user.
- b). Click the button to add the user with the role that you want for the user.
 - ▶ To add the user with the [Virtual Group Administrator](#) role, click **ADMIN**.
 - ▶ To add the user with the [Virtual Group User](#) role, click **USER**.
- c). Click **Next: Preview group creation**.



Note: Any user with the [Organization Administrator](#) role loses that role and gains the role that you assign when added to the virtual group.

You must add **at least one** virtual group administrator to the group. You cannot create a virtual group with no administrators.



Tip: To prevent the absence of a single user from denying you access to the virtual group, consider adding at least two virtual group administrators to the virtual group.

8. On the Preview server creation page, click **CREATE VIRTUAL GROUP**.

After you create a virtual group, you can perform only the following operations on the virtual group:

- ▶ Deleting the virtual group
- ▶ Assigning an entitlement at the organization level to the virtual group
- ▶ Removing an entitlement from the virtual group and returning it to the organization

Other operations on the virtual group require the virtual group administrator or virtual group user role.

9.3. Deleting a Virtual Group

Delete a virtual group if it is no longer needed. When the group is deleted, all entitlements assigned to the group and any contacts who are members only of this group are returned to the organization. Contacts who are returned to the organization are assigned the organization user role.

1. If you are not already logged in, log in to the NVIDIA Licensing Portal.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **VIRTUAL GROUPS**.
3. In the list of virtual groups on the **Virtual Groups** page, from the **Actions** menu for the virtual group that you want to delete, choose **Delete**.
4. When asked to confirm that you want to delete the virtual group, click **DELETE VIRTUAL GROUP**.

9.4. Adding a Contact to a Virtual Group

If you have the Organization Administrator role, you can add a contact to a virtual group in your organization without the need to be a member of the group.

The contact that you add must **not** have the [Organization Administrator](#) role.

1. If you are not already logged in, log in to the NVIDIA Licensing Portal.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **VIRTUAL GROUPS**.
3. In the list of virtual groups on the **Virtual Groups** page, from the **Actions** menu for the virtual group to which you want to add the contact, choose **Invite User**.

4. In the **Invite User** pop-up window that opens, provide the e-mail address and the name of the contact, select the contact's role, and click **SEND INVITATION**.

Select one of the following roles:

Virtual Group Admin

Assigns the contact the [Virtual Group Administrator](#) role.

Virtual Group User

Assigns the contact the [Virtual Group User](#) role.

An e-mail is sent to the contact at the e-mail address that you provided.

- ▶ If the contact is **not** already be registered on the NVIDIA Licensing Portal, the e-mail provides directions for creating an NVIDIA Enterprise Account.
- ▶ Otherwise, the e-mail provides details about the contact's role change.

If the contact is **not** already be registered on the NVIDIA Licensing Portal, tell the contact to follow the directions in the e-mail to create an NVIDIA Enterprise Account.

9.5. Removing a Contact from a Virtual Group

If you have the Organization Administrator role, you can remove a contact from a virtual group in your organization without the need to be a member of the group. The contact that you remove is returned to the organization and assigned the Organization User role.

1. If you are not already logged in, log in to the NVIDIA Licensing Portal.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **VIRTUAL GROUPS**.
3. In the list of virtual groups on the **Virtual Groups** page, from the **Actions** menu for the virtual group to which the contact belongs, choose **Remove User**.
4. In the **Remove User** dialog box that opens, select the contact that you want to remove.
5. When asked to confirm that you want to remove the contact from virtual group, click **REMOVE USER**.

9.6. Managing Entitlements in a Virtual Group

Remove an entitlement from a virtual group to return it to the organization either to make it available to users at the organization level or to transfer it to a different virtual group.

Ensure that the following prerequisites are met:

- ▶ The entitlements that you want to add belong to the organization and not to a virtual group.
- ▶ No licensed products in any of the entitlements that you want to add to the group have been added to a license server.

Ensure that no licensed products in the entitlement that you want to remove have been added to a license server.

1. If you are not already logged in, log in to the NVIDIA Licensing Portal.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **VIRTUAL GROUPS**.
3. From the **Actions** menu for the virtual group, choose **Manage Entitlements**.
4. In the **Manage Entitlements** pop-up window that opens, select the entitlements that you want to add to the virtual group.



Note: You cannot move an entitlement if any of its licensed products has been added to a license server. If any licensed products in an entitlement have been added to a license server, the entitlement is not listed on the **Add Entitlements** tab.

5. Conversely, in the list of entitlements, follow the **Remove** link for the entitlement.



Note: You cannot remove an entitlement if any of its licensed products has been added to a license server or if it is the only entitlement in a group. The **Remove** link for the entitlement is inactive and dimmed.

6. When asked to confirm that you want to remove the entitlement, click **REMOVE ENTITLEMENT**.
The entitlement is removed from the list of entitlements in the virtual group and added to the list of entitlements in the organization.
7. After adding or returning all the entitlements that you are interested in, click **UPDATE VIRTUAL GROUP**.
The entitlement is removed from the list of entitlements in the organization and added to the list of entitlements in the virtual group.

9.7. Sample Business Scenario for Virtual Groups

A common business scenario for virtual groups is a multinational corporation with subsidiaries in which licenses are managed centrally.

Organization Administrators

The organization administrators are responsible for setting up virtual groups and managing entitlements for the entire organization. The individuals chosen to be organization administrators must understand the organization structure and purchasing process, so that they are capable of routing newly purchased entitlements appropriately.

To ensure that someone is always available to move newly purchased entitlements into the correct virtual group, consider designating **at least three** organization administrators.

Virtual Groups

To simplify the allocation entitlements to the entity for which they were purchased, consider creating a virtual group for every subsidiary or geographic region, as appropriate.

Organization-Level Account



Virtual Group Contacts

To ensure redundancy at every level in your organization, designate **at least two** virtual group administrators for each virtual group.

After a virtual group is created, its virtual group administrators are free to add contacts who are **not** organization administrator as required.

Appendix A. Migrating Licenses from a Legacy NVIDIA vGPU Software License Server

Follow this work flow to minimize the disruption of service when migrating licenses from a legacy NVIDIA vGPU software license server.

This work flow consists of several separate phases. Work through the phases in the order in which they are presented.

A.1. Tasks for Preparing to Migrate Licenses

Task	Cross-Reference
Decide which type of service instances to deploy.	About Service Instances

A.2. CLS Instances Only: Tasks for Configuring CLS Instances


For each legacy NVIDIA vGPU software license server, perform the tasks in the order in which they are listed.

Task	Cross-Reference
<p>Optional: Ensure that the CLS instance on which the converted license server will be installed has been created.</p> <p>If you want to use a custom CLS instance but don't create it in advance, you can create it during the conversion of a legacy NVIDIA vGPU software license server to an NLS license server.</p>	Creating a CLS Instance on the NVIDIA Licensing Portal

Task	Cross-Reference
<p>Convert the legacy NVIDIA vGPU software license server to an NLS license server on an express CLS installation or a custom CLS instance.</p> <p>Your legacy NVIDIA vGPU software license servers continue to serve licenses to clients until you reconfigure your clients to use the converted NLS license servers.</p>	<ul style="list-style-type: none"> ▶ Converting a Legacy License Server to an NLS License Server on an Express CLS Installation ▶ Converting a Legacy License Server to an NLS License Server on a Custom CLS Instance
<p>Optional: Change the licensing mode of the converted license server.</p> <p>The conversion process sets the licensing mode of the converted license server to Standard Networked Licensing. If you want a different mode for the converted license server, you can change it after the conversion.</p>	<p>Changing the Leasing Mode of a License Server</p>

A.3. DLS Instances Only: Tasks for Installing and Configuring the DLS Virtual Appliance

Perform the tasks in the order in which they are listed.

Task	Cross-Reference
Install the DLS virtual appliance.	Installing the DLS Virtual Appliance Image on a Supported Hypervisor
If not set automatically: Set the IP address of a DLS virtual appliance.	Setting the IP Address of a DLS Virtual Appliance from the Hypervisor
Register the DLS administrator user.	Registering the DLS Administrator User
Configure an HA cluster of DLS instances.	Configuring an HA Cluster of DLS Instances
<div style="background-color: #f0f0f0; padding: 5px;">  Note: Manual configuration of a standalone DLS instance is not required. By default, a DLS instance is initially created as a standalone instance. </div>	

A.4. DLS Instances Only: Tasks for Configuring DLS Instances

For each legacy NVIDIA vGPU software license server, perform the tasks in the order in which they are listed.

Task	Cross-Reference
Ensure that the DLS instance on which the converted license server will be installed is registered.	<ul style="list-style-type: none"> ▶ Registering an on-Premises DLS Instance with the NVIDIA Licensing Portal ▶ Registering a DLS Instance on an Air-Gapped Network with the NVIDIA Licensing Portal
<p>Convert the legacy NVIDIA vGPU software license server to an NLS license server on a DLS instance.</p> <p>Your legacy NVIDIA vGPU software license servers continue to serve licenses to clients until you reconfigure your clients to use the converted NLS license servers.</p>	Converting a Legacy License Server to an NLS License Server on a DLS Instance
<p>Optional: Change the licensing mode of the converted license server.</p> <p>The conversion process sets the licensing mode of the converted license server to Standard Networked Licensing. If you want a different mode for the converted license server, you can change it after the conversion.</p>	Changing the Leasing Mode of a License Server
Install the converted license server.	Installing a License Server on a DLS Instance

A.5. Tasks for Managing Licenses on a License Server

Perform the tasks in the order in which they are listed.

Task	Cross-Reference
Optional: Create license pools for the licenses on a license server.	Creating a License Pool
Optional: Create fulfillment conditions for your license pools.	Creating a Fulfillment Condition
Generate a client configuration token.	Generating a Client Configuration Token

A.6. Tasks for Configuring a Licensed Client

Task	Cross-Reference
Configure a licensed client.	Configuring a Licensed Client with a Networked License

A.7. Tasks for Decommissioning a Legacy NVIDIA vGPU software License Server

You can wait until you have migrated your licenses and tested them on NVIDIA License System before decommissioning your existing legacy NVIDIA vGPU software license servers.

Task	Cross-Reference
Remove the legacy NVIDIA vGPU software license server from your Windows or Linux platforms.	<ul style="list-style-type: none"> ▶ Uninstalling the NVIDIA vGPU Software License Server on Windows ▶ Uninstalling the NVIDIA vGPU Software License Server on Linux

A.7.1. Uninstalling the NVIDIA vGPU Software License Server on Windows

If you want to uninstall the license server in silent mode, ensure that the license server installation directory contains the `installer.properties` file.

For information about the `installer.properties` file, refer to [Virtual GPU Software License Server User Guide](#).

1. Start the license server uninstaller.

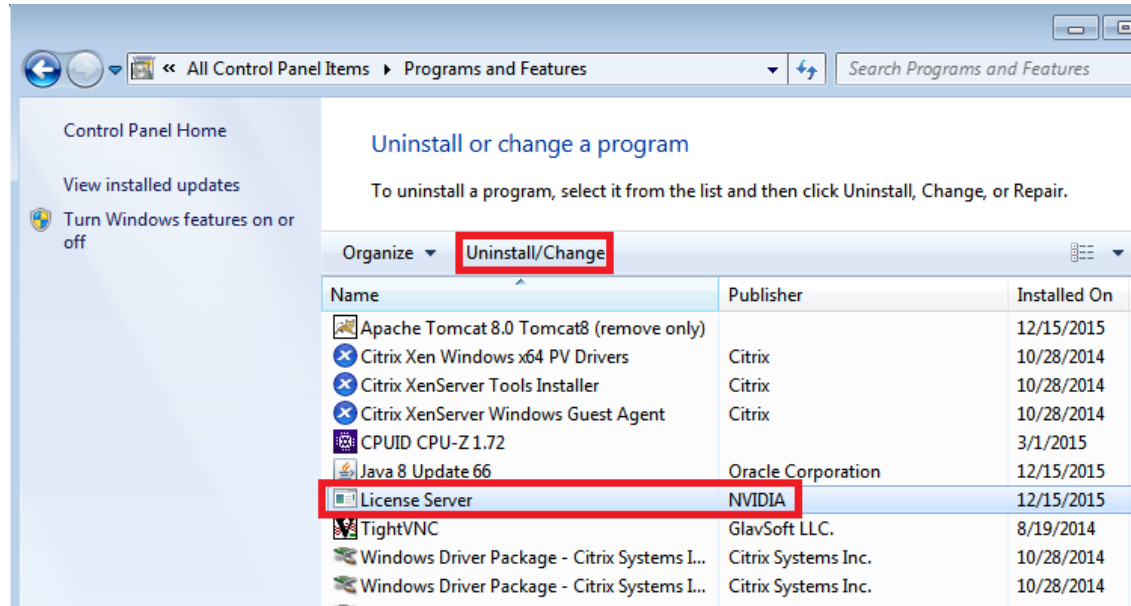
You can start the license server uninstaller from **Windows Control Panel**, **Windows Explorer**, or a **Command Prompt** window.

If you start the license server uninstaller from **Windows Control Panel** or **Windows Explorer**, how it runs depends on the mode in which license server software was installed. If the license server software was installed in console mode or silent mode, the uninstaller runs in **silent** mode. Otherwise, the **Configure License Server** dialog box opens.

To ensure that you start the license server uninstaller in the required mode, start it from a **Command Prompt** window.

- ▶ In **Windows Control Panel**, open the **Programs and Features** pane, select **License Server** from the publisher NVIDIA, and click **Uninstall/Change**.

Figure 1. Starting the Uninstaller from Windows Control Panel



- ▶ In **Windows Explorer**, open the license server installation directory and double-click the **Change License Server Installation** application.

The default license server installation folder is %SystemDrive%:\.

- ▶ In a **Command Prompt** window, change to the license server installation directory and start the license server uninstaller in **interactive console** mode.

The default license server installation folder is %SystemDrive%:\.

```
C:\>"Change License Server Installation.exe" -i console
```

- ▶ In a **Command Prompt** window, change to the license server installation directory and start the license server uninstaller in **silent** mode.

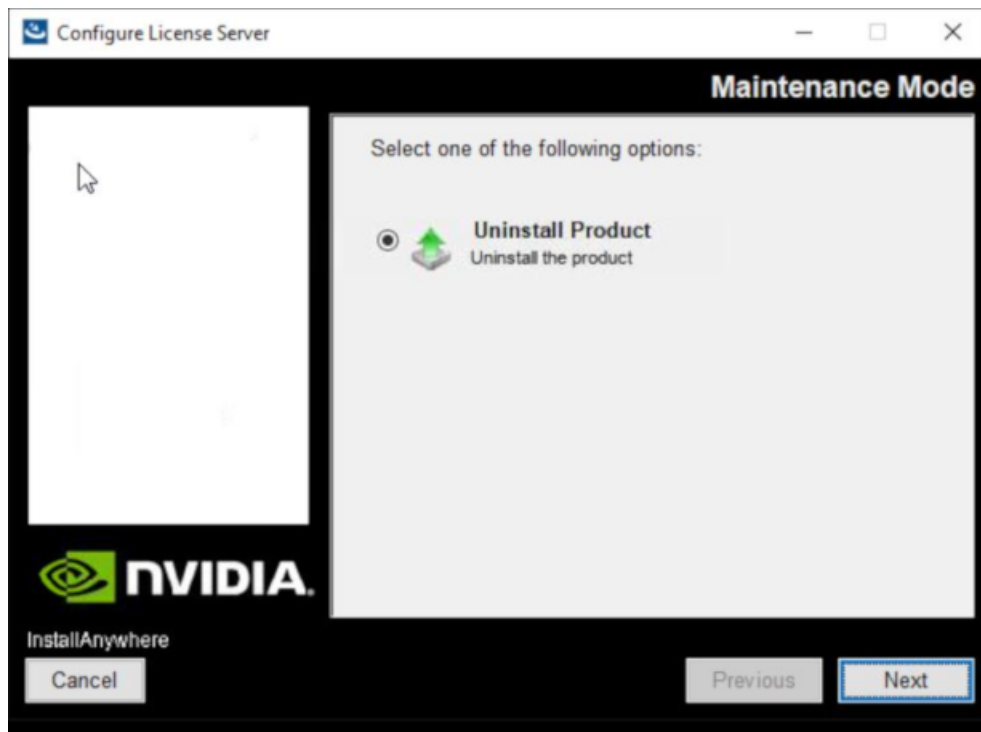
The default license server installation folder is %SystemDrive%:\.

```
C:\>"Change License Server Installation.exe" -i silent
```

If you start the license server uninstaller in silent mode, it runs without prompting you for any information. The remaining steps are **not** required.

2. If the **Configure License Server** dialog box opens, ensure that the **Uninstall Product** option is selected and click **Next**.

Figure 2. Running the License Server Uninstaller on Windows



3. When prompted, confirm that you want to uninstall the license server and specify whether you want to uninstall **Apache Tomcat**.

A.7.2. Uninstalling the NVIDIA vGPU Software License Server on Linux

If you want to uninstall the license server in silent mode, ensure that the license server installation directory contains the `installer.properties` file.

For information about the `installer.properties` file, refer to [Virtual GPU Software License Server User Guide](#).

1. Change to the license server installation directory.

The default license server installation directory is `/opt/flexnetls/nvidia`.

```
[nvidia@localhost ~]$ cd /opt/flexnetls/nvidia
```

2. As root, start the license server uninstaller.

- If the license server software was installed in graphical mode or console mode, and you want to run the uninstaller in the same mode, run the `Change License Server Installation` command without any arguments.

```
[nvidia@localhost ~]$ sudo ./Change\ License\ Server\ Installation
```

If the license server software was installed in graphical mode, the **Configure License Server** window opens.

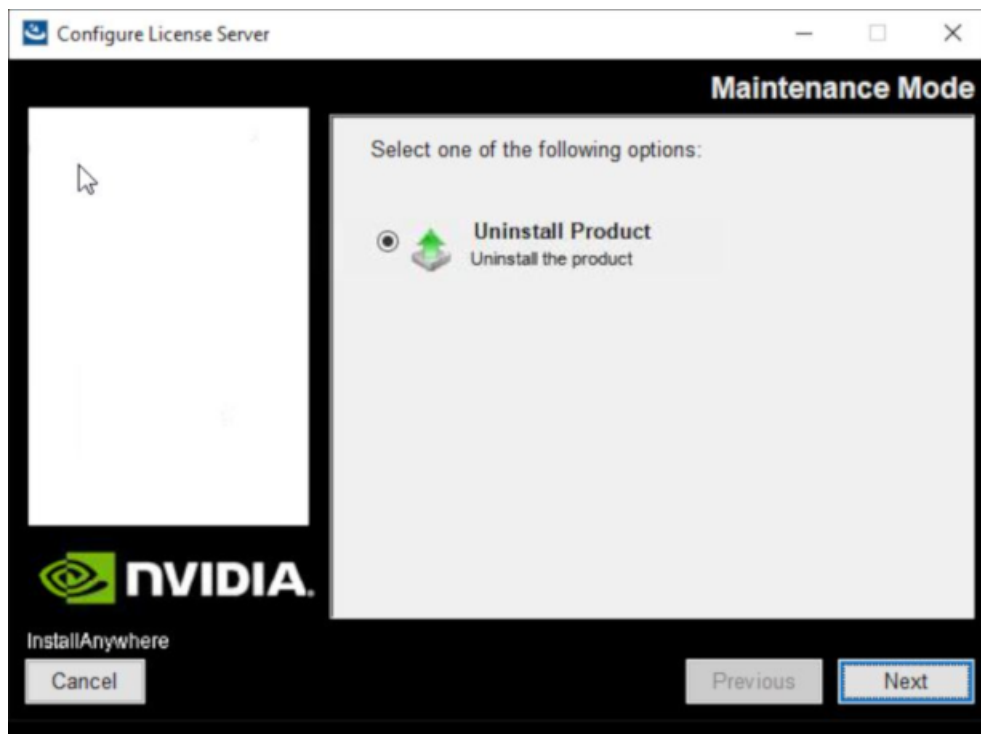
- ▶ If you want to run the uninstaller in silent mode, run the `Change License Server Installation` command with the `-i silent` argument.

```
[nvidia@localhost ~]$ sudo ./Change\ License\ Server\ Installation -i silent
```

If you start the license server uninstaller in silent mode, it runs without prompting you for any information. The remaining steps are **not** required.

3. If the **Configure License Server** window opens, ensure that the **Uninstall Product** option is selected and click **Next**.

Figure 3. Running the License Server Uninstaller on Linux



4. When prompted, confirm that you want to uninstall the license server.

Appendix B. Getting Started with the NLS RESTful APIs

NVIDIA License System (NLS) provides RESTful APIs for managing its Cloud License Service (CLS) and Delegated License Service (DLS) components. The CLS RESTful API is provided by the NVIDIA Licensing Portal. The DLS RESTful API is provided by the **NVIDIA Licensing** application on the DLS appliance.

Comprehensive reference information generated by Swagger for these APIs is available as follows:

- ▶ **CLS RESTful API:** [NVIDIA License System APIs](#)
- ▶ **DLS RESTful API:** [NVIDIA License System Virtual Appliance APIs](#)

B.1. Getting Started with the CLS RESTful API

The NVIDIA Licensing Portal provides a RESTful API for managing the Cloud License Service (CLS) component of NVIDIA License System.

Comprehensive reference information generated by Swagger for this API is available in [NVIDIA License System APIs](#).

B.1.1. Creating a Licensing State API Key

Before using the CLS RESTful API to manage a CLS instance, you must create a licensing state API key from the NVIDIA Licensing Portal. This key allows **only** read access to an organization's resources on the NVIDIA Licensing Portal. Therefore, you can use this key for authentication **only** in HTTP `GET` requests.

1. If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **API KEYS**.
3. In the **API Key Management** window that opens, click **CREATE API KEY**.
4. In the **Create API Key** window that opens, provide the name, type of access, and lifetime of the key, and click **CREATE API KEY**.
 - a). In the **Key name** text-entry field, type your choice of name for the key.

- b). From the **Access type** drop-down list, select **Licensing State**.
- c). Limit the virtual groups or DLS instances where this key can be used.
By default, the key is valid for all resources and DLS instances in the organization.
- d). In the **Days until expiration** text-entry field, type the lifetime of the key in days or, if you don't want the key to expire, deselect the **Expire this key after a specified number of days?** option.

The key is added to the table of API keys in the **API Key Management** window.

5. When you're ready to use the licensing state key in an HTTP request, follow the **view api key** link and copy the key to the clipboard.

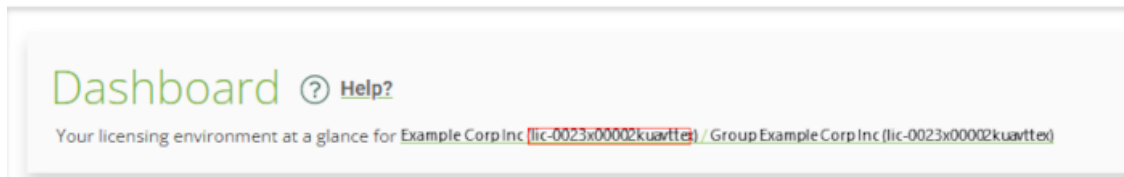
Pass the licensing state API key in the `x-api-key` header in all subsequent requests.

B.1.2. Getting Information Required in Other RESTful API Calls to a CLS Instance

The organization ID, virtual group ID, and license server ID are required in other RESTful API calls to a CLS instance.

Ensure that you have created the licensing state API key that you need for authentication as explained in [Creating a Licensing State API Key](#).

1. Get the ID of the organization to which the user that created the API key belongs.
 - a). If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
 - b). On the NVIDIA Licensing Portal dashboard, locate the ID in parentheses next to the organization name.



2. Submit an HTTP `GET` request to get information about all virtual groups that can access the license servers bound to the CLS instance.

```
GET https://api.licensing.nvidia.com/v1/org/org-name/virtual-groups
org-name
```

The ID of the organization to which the user that created the API key belongs, which you obtained in the previous step. For example: `lic-0023x00002kuavttex`.

Pass the required API key in the `x-api-key` header. For information about how to create this key, refer to [Creating a Licensing State API Key](#).

This example uses the `curl` command to get information for other RESTful API calls to the CLS instance for a user in the organization `lic-0023x00002kuavttex`. The user is authenticated by an API key. For clarity, the example shows only the placeholder `api-key`.

```
curl -k --location --request GET \
```

```
'https://api.licensing.nvidia.com/v1/org/lic-0023x00002kuavttex/virtual-groups' \
--header 'x-api-key: api-key'
```

The information required is returned in the following fields in the body of the response:

licenseServerId

The ID of the license server that is bound to the CLS instance. For example:

```
"licenseServerId": "5244e9b3-8675-11ed-857a-03a0ffbeaa4c"
```

virtualGroupId

The ID of the virtual group to which the license server that is bound to the CLS instance belongs. For example:

```
"virtualGroupId": 89
```

B.1.3. Getting License Server Information for a CLS Instance

Ensure that the following prerequisites are met:

- ▶ You have created the licensing state API key that you need for authentication as explained in [Creating a Licensing State API Key](#).
- ▶ You have obtained the information that you need to complete this task as explained in [Getting Information Required in Other RESTful API Calls to a CLS Instance](#).

Submit an HTTP `GET` request to get information about the license server that you are interested in, which is bound to the CLS instance.

```
GET https://api.licensing.nvidia.com/v1/org/org-name/virtual-groups/virtual-
group-id/license-servers/license-server-id
```

org-name

The code for the organization to which the user that created the API key belongs.

For example: `lic-0023x00002kuavttex`.

virtual-group-id

The identifier of the virtual group to which the license server that is bound to the CLS instance belongs. For example: `89`.

license-server-id

The identifier of the license server that you are interested in, which is bound to the CLS instance. For example: `5244e9b3-8675-11ed-857a-03a0ffbeaa4c`.

You can obtain this information as explained in [Getting Information Required in Other RESTful API Calls to a CLS Instance](#).

Pass the required API key in the `x-api-key` header. For information about how to create this key, refer to [Creating a Licensing State API Key](#).

This example uses the `curl` command to get information for license server `5244e9b3-8675-11ed-857a-03a0ffbeaa4c`. The license server is bound to the CLS instance for a user in virtual group `89` of the organization `lic-0023x00002kuavttex`. The user is authenticated by an API key. For clarity, the example shows only the placeholder `api-key`

```
curl -k --location --request GET \
'https://api.licensing.nvidia.com/v1/org/lic-0023x00002kuavttex/virtual-groups/89/
license-servers/5244e9b3-8675-11ed-857a-03a0ffbeaa4c' \
--header 'x-api-key: api-key'
```

B.1.4. Listing Licenses that Are Being Served by a CLS Instance

Ensure that the following prerequisites are met:

- ▶ You have created the licensing state API key that you need for authentication as explained in [Creating a Licensing State API Key](#).
- ▶ You have obtained the information that you need to complete this task as explained in [Getting Information Required in Other RESTful API Calls to a CLS Instance](#).

1. Get the ID of the CLS instance that you are interested in.
 - a). Submit an HTTP `GET` request to get information about all license servers that you can access from your virtual group.

```
GET https://api.licensing.nvidia.com/v1/org/org-name/virtual-groups/virtual-group-id/license-servers
```

org-name

The code for the organization to which the user that created the API key belongs. For example: `lic-0023x00002kuavttex`.

virtual-group-id

The identifier of the virtual group to which the license server that is bound to the CLS instance belongs. For example: `89`.

You can obtain this information as explained in [Getting Information Required in Other RESTful API Calls to a CLS Instance](#).

Pass the API key required for authentication in the `x-api-key` header. For information about how to create this key, refer to [Creating a Licensing State API Key](#).

This example uses the `curl` command to get information for all license servers that can be accessed by a user in virtual group `89` of the organization `lic-0023x00002kuavttex`. The user is authenticated by an API key. For clarity, the example shows only the placeholder `api-key`

```
curl -k --location --request GET \
'https://api.licensing.nvidia.com/v1/org/lic-0023x00002kuavttex/virtual-groups/89/license-servers' \
--header 'x-api-key: api-key'
```

- b). Examine each pair of `serviceInstanceId` and `serviceInstanceName` fields in the body of the response to find the ID of the CLS instance that you are interested in. For example:

```
"serviceInstanceId": "1d85d904-84db-405f-b1dc-8ef9ded4f4cd",
"serviceInstanceName": "CWBI-DLS-2-1-0"
```

2. Submit an HTTP `GET` request to list the licenses that are being served by the CLS instance that you're interested in.

```
GET https://api.licensing.nvidia.com/v1/org/org-name/virtual-groups/virtual-group-id/leases
```

org-name

The `org-name` that you specified in the previous step.

virtual-group-id

The `virtual-group-id` that you specified in the previous step.

In the request, pass the headers listed in the following table.

Header	Value
x-nv-service-instance-id	The ID of the service instance that you are interested in, which you obtained in the previous step.
x-api-key	The API key required for authentication that you passed in the previous step.

This example uses the `curl` command to list the licenses that are being served by the CLS instance `1d85d904-84db-405f-b1dc-8ef9ded4f4cd`. The user is authenticated by an API key. For clarity, the example shows only the placeholder `api-key`.

```
curl -k --location --request GET \
'https://api.licensing.nvidia.com/v1/org/lic-0023x00002kuavttex/virtual-groups/89/
license-servers/5244e9b3-8675-11ed-857a-03a0ffbbaa4c' \
--header 'x-nv-service-instance-id: 1d85d904-84db-405f-b1dc-8ef9ded4f4cd' \
--header 'x-api-key: api-key'
```

B.2. Getting Started with the DLS RESTful API

The **NVIDIA Licensing** application on the Delegated License Service (DLS) appliance provides a RESTful API for managing the DLS component of NVIDIA License System.

Comprehensive reference information generated by Swagger for this API is available in [NVIDIA License System Virtual Appliance APIs](#).

B.2.1. Exporting and Importing API Keys for a DLS Instance

API keys provide an alternative method for authentication and for getting the information required in other RESTful API calls. Any API keys that are created after a DLS instance has been registered with the NVIDIA Licensing Portal are not recognized by the DLS instance. To ensure that these API keys are recognized, you must export them from the NVIDIA Licensing Portal and import them into the DLS instance.



Note: Any existing API keys are exported and imported automatically for a DLS instance when the following tasks are performed on the instance:

- ▶ The DLS instance is registered with the NVIDIA Licensing Portal.
- ▶ A license server is installed on the DLS instance.

You can export and import only API keys whose access type contains **Licensing State**, **Enterprise** or both **Licensing State** and **Enterprise**. You cannot export and import API keys whose access type contains only **Software Downloads**. However, you can export and import API keys whose access type contains **Software Downloads** and at least one other access type.

B.2.1.1. Exporting API Keys for a DLS Instance

Ensure that the API keys that you want to export have been created.

1. If you are not already logged in, log in to the [NVIDIA Enterprise Application Hub](#) and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.
2. In the left navigation pane of the NVIDIA Licensing Portal dashboard, click **API KEYS**.
3. In the **API Key Management** window that opens, click **EXPORT DLS KEYS**.
4. In the **Export DLS Keys** window that opens, from the drop-down list, select the DLS instance for which you want to export API keys and click **EXPORT DLS KEY FILE**.

The DLS instance that you select **must** be the DLS instance into which you want to import the API keys.

A file named `dls_key_export_random-number.tok` that contains the exported API keys is saved to your default downloads folder.

B.2.1.2. Importing API Keys for a DLS Instance

Ensure that the API keys that you want to import have been exported as explained in [Exporting API Keys for a DLS Instance](#). The DLS instance that was selected when the API keys were exported **must** be the DLS instance into which you want to import the API keys.

1. If you are not already logged in, log in to the **NVIDIA Licensing** application at the IP address of the VM on which the DLS virtual appliance is installed.
2. In the left navigation pane, click **SETTINGS**.
3. On the **Service Instance Settings** page that opens, expand the **API Keys** section and click **Import API Keys**.
4. In the file browser that opens, navigate to the folder that contains a file named `dls_key_export_random-number.tok` that contains the API keys that you want to import and select the file.

B.2.2. Logging in to a DLS Appliance from the RESTful API

Before using the DLS RESTful API to manage a DLS instance, you must log in as the DLS administrator user to the DLS appliance that hosts the DLS instance. Logging in also returns authentication information that you must use in all subsequent calls to the DLS RESTful API.

Submit an HTTP `POST` request with basic authentication to log in as the DLS administrator user to the DLS appliance.

```
POST https://host:port/auth/v1/login
```

host

The IP address or fully qualified domain name of the VM or container that is hosting the DLS appliance.

If the tool that you use to submit the request supports cookies, you can submit the request with the `LIC-TOKEN` cookie. Otherwise, you must pass the required authorization token in the `Authorization: Bearer` header. The `LIC-TOKEN` cookie and the required authorization token are returned when you log in to a DLS appliance as explained in [Logging in to a DLS Appliance from the RESTful API](#).

This example uses the `curl` command to get information for other RESTful API calls to the DLS appliance `dls.example.com` on port 8081. The user is authenticated by the `LIC-TOKEN` cookie. For clarity, the example shows only the placeholder `cookie`.

```
curl -k --location --request GET 'https://dls.example.com:8081/auth/v1/user/me' \
--cookie "LIC-TOKEN=cookie"
```

The information required is returned in the following fields in the body of the response:

orgName

The ID of the organization to which the DLS administrator user that logged into the DLS appliance belongs. For example:

```
"orgName": "lic-0011w00001roiwqqay"
```

licenseServerId

The ID of the license server that is bound to the DLS instance. For example:

```
"licenseServerId": "4133f8a4-7564-11ed-968a-02a0eeafbb5d"
```

virtualGroupId

The ID of the virtual group to which the license server that is bound to the DLS instance belongs. For example:

```
"virtualGroupId": 3
```

B.2.4. Getting License Server Information for a DLS Instance

Ensure that the following prerequisites are met:

- ▶ You have logged in to the DLS instance and obtained the cookie or authorization token that you need for authentication as explained in [Logging in to a DLS Appliance from the RESTful API](#).
- ▶ You have obtained the information that you need to complete this task as explained in [Getting Information Required in Other RESTful API Calls to a DLS Instance](#).

Send an HTTP `GET` request to get information about the license server that is bound to the DLS instance.

```
GET https://host:port/admin/v1/org/org-name/virtual-groups/virtual-group-id/
license-servers
```

host

The IP address or fully qualified domain name of the VM or container that is hosting the DLS appliance.

port

The port on which the VM or container that is hosting the DLS appliance listens for HTTPS requests.

org-name

The ID for the organization to which the user that logged in to the DLS instance belongs. For example: `lic-0011w00001roiwqqay`.

You can obtain this ID as explained in [Getting Information Required in Other RESTful API Calls to a DLS Instance](#).

virtual-group-id

The ID of the virtual group to which the license server that is bound to the DLS instance belongs. For example: 3.

You can obtain this ID as explained in [Getting Information Required in Other RESTful API Calls to a DLS Instance](#).

If the tool that you use to submit the request supports cookies, you can submit the request with the `LIC-TOKEN` cookie. Otherwise, you must pass the required authorization token in the `Authorization: Bearer` header. The `LIC-TOKEN` cookie and the required authorization token are returned when you log in to a DLS appliance as explained in [Logging in to a DLS Appliance from the RESTful API](#).

This example uses the `curl` command to get license server information for the DLS instance on the DLS appliance `dls.example.com` on port 8081. The user belongs to virtual group 3 in the organization `lic-0011w00001roiwqqay` and is authenticated by the `LIC-TOKEN` cookie. For clarity, the example shows only the placeholder `cookie`.

```
curl -k --location --request GET \
'https://dls.example.com:8081/admin/v1/org/lic-0011w00001roiwqqay/virtual-groups/3/
license-servers' \
--cookie "LIC-TOKEN=cookie"
```

B.2.5. Listing Licenses that Are Being Served by a DLS Instance

Ensure that the following prerequisites are met:

- ▶ You have logged in to the DLS instance and obtained the cookie or authorization token that you need for authentication as explained in [Logging in to a DLS Appliance from the RESTful API](#).
- ▶ You have obtained the information that you need to complete this task as explained in [Getting Information Required in Other RESTful API Calls to a DLS Instance](#).

Send an HTTP `GET` request to list the licenses that are being served by the DLS instance that you're interested in.

```
GET https://host:port/admin/v1/org/org-name/virtual-groups/virtual-group-id/
leases
```

host

The IP address or fully qualified domain name of the VM or container that is hosting the DLS appliance.

port

The port on which the VM or container that is hosting the DLS appliance listens for HTTPS requests.

org-name

The code for the organization to which the user that logged in to the DLS instance belongs. For example: `lic-0011w00001roiwqqay`.

You can obtain this code as explained in [Getting Information Required in Other RESTful API Calls to a DLS Instance](#).

virtual-group-id

The identifier of the virtual group to which the license server that is bound to the DLS instance belongs. For example: 3.

You can obtain this identifier as explained in [Getting Information Required in Other RESTful API Calls to a DLS Instance](#).

If the tool that you use to submit the request supports cookies, you can submit the request with the `LIC-TOKEN` cookie. Otherwise, you must pass the required authorization token in the `Authorization: Bearer` header. The `LIC-TOKEN` cookie and the required authorization token are returned when you log in to a DLS appliance as explained in [Logging in to a DLS Appliance from the RESTful API](#).

This example uses the `curl` command to list the licenses that are being served by the DLS instance on the DLS appliance `dls.example.com` on port 8081. The user belongs to virtual group 3 in the organization `lic-0011w00001roiwqqay` and is authenticated by the `LIC-TOKEN` cookie. For clarity, the example shows only the placeholder *cookie*.

```
curl -k --location --request GET \  
'https://dls.example.com:8081/admin/v1/org/lic-0011w00001roiwqqay/virtual-groups/3/  
leases' \  
--cookie "LIC-TOKEN=cookie"
```

Appendix C. Performance Data for a CLS Instance

Use the measured performance numbers to determine whether the CLS meets your requirements based on the number and frequency of requests from licensed clients. Hosting a CLS instance on a cloud service provides robustness and dynamic scalability for the CLS instance. The cloud service dynamically scales CLS instances to maintain the measured performance numbers.



Note: For the corresponding data for a DLS virtual appliance, refer to [Sizing Guidelines for a DLS Appliance](#).

C.1. Throughput for a CLS Virtual Appliance

Throughput measures the number of clients that a CLS instance can process in 1 second.

A CLS instance can process up to 40 clients in 1 second.

The average response time is 436 ms.

C.2. Scalability for a CLS Instance

Scalability measures the maximum number of licensed clients that a CLS instance can serve in a specific interval. A CLS instance serves a licensed client by performing a licensing operation for the client, namely the borrowing, return, or renewal of a license. Registration of a licensed client is not considered a licensing operation because it occurs only once for any client.

These measurements capture the maximum number of licensed clients that a CLS instance can serve when 40 clients connect concurrently to the CLS instance. The maximum number of licensed clients is measured for different lengths of time up to 24 hours for which a license is borrowed.



Note:

- ▶ Intervals in the table are the renewal intervals when a client contacts the DLS appliance to request a licensing operation.
- ▶ The renewal interval is set to 15% of the length of time for which a license is borrowed.

Length of Time License Is Borrowed	Interval	Maximum Number of Licensed Clients
10 minutes	1.5 minutes	3,500
1 hour	9 minutes	21,000
12 hours	1.8 hours	259,000
24 hours	3.6 hours	518,000

C.3. Burst Load Performance for a CLS Instance

Burst load performance measures the time that a CLS instance requires to process the requests received from a large number of clients in a short interval of time. Burst load performance does **not** measure concurrency or throughput.



Note: Burst processing times are illustrative only because they are for retry logic in performance tests that use simulated client drivers. Times may differ with real client drivers.

Number of Clients	Interval	Processing Time
100	0-1 second	10 seconds
1,000	0-2 seconds	1 minute
5,000	100 seconds (1 minute and 40 seconds)	5 minutes
10,000	200 seconds (3 minutes and 20 seconds)	10 minutes

The NVIDIA Licensing Portal limits the maximum number of concurrent requests to 100.

C.4. NVIDIA Cloud Licensing Service (CLS) Service Level Objectives

The NVIDIA Cloud Licensing Service (CLS) is a managed service supporting automated license borrow, renewal, and return operations for NVIDIA licensed software and hardware products. It is a globally-distributed platform built to be scalable to meet customer demand.

NVIDIA strives to maintain the following Service Level Objectives for the CLS public endpoints that are used for typical license operations.

Monthly Service Availability**> 99.95%**

We define availability as the percentage of requests successfully processed, as a proportion of all valid requests received, during a given time period. A response is considered successful if it **completes in under 3 seconds** (as measured at our service edge) and **does not return an unexpected error** (e.g. 5XX code).

To calculate Monthly Service Availability, NVIDIA uses a combination of methods, including information from NVIDIA's event monitoring systems and logs and observing external causes that may impact the services (to the extent known to NVIDIA). Availability metrics are rolled-up to daily and monthly statistics that are reported on our public status page <https://status.licensing.nvidia.com>.

If your use of the CLS does not achieve the objectives of this SLO, you may notify NVIDIA. NVIDIA may make further improvements to the CLS in its sole discretion. You will not be entitled to receive any further support or remedies from NVIDIA based on the SLO.

Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. NVIDIA Corporation ("NVIDIA") makes no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer ("Terms of Sale"). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer's own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer's sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer's product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND 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 NON-INFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA's aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

VESA DisplayPort

DisplayPort and DisplayPort Compliance Logo, DisplayPort Compliance Logo for Dual-mode Sources, and DisplayPort Compliance Logo for Active Cables are trademarks owned by the Video Electronics Standards Association in the United States and other countries.

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 Maxwell, NVIDIA Pascal, NVIDIA Turing, NVIDIA Volta, 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

© 2021-2024 NVIDIA Corporation. All rights reserved.

