

# NLS2.0 许可证系统的安装配置流程和 PoC 简介

## 修订记录

Date	Version	Authors	Description
2021/11/15	1.0	Merlin Ma	Initial version written for NLS 1.0.
2022/08/04	1.1	Shen Song	Add change default vApp mode to vWS in PT mode.
2022/09/06	2.0	Merlin Ma	Adapt operation to NLS 2.0 and the current NLP portal. Add new Container based DLS installation step by step. Restructuring DLS operation order.

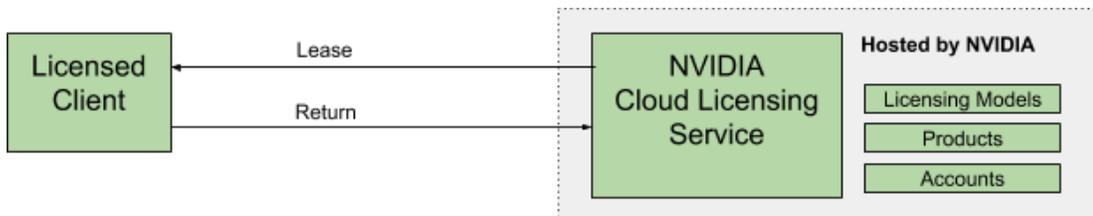
## 概述

在 2021 年 8 月伴随 vGPU 13.0 的发布，NVIDIA 推出了全新的软件 License 系统（NLS）来替代之前基于 Flexnet 的 License 服务。新的 License 系统需要至少 vGPU 版本 13.0 或者 NVAIE 才能支持使用。如果您使用的是 vGPU 13.0 之前的版本，仍然需要使用传统的 LS 授权服务器。因此本文将着重介绍全新的 NLS 的安装配置过程以及在 vGPU 13.0 中如何使用新的 NLS 来实现软件授权。全新的 NLS 提供两种 License 服务形式来授权 vGPU，一种是在有公网访问环境，vGPU 可以通过 NVIDIA 官方服务器上面的 CLS 服务进行授权，另一种是在企业私有网络搭建本地的 DLS 服务进行 License 授权。这两种方式本文也将介绍和演示使用方法。同时，13.0 以及后续版本仍然在一定时间范围内兼容旧的 LS，但强烈建议尽快迁移到新的 NLS。NLS 2.0 在 2022 年 8 月底发布，主要增加了容器方式的 DLS 部署，更加灵活和可管理，本文也针对这个部分做了更新。

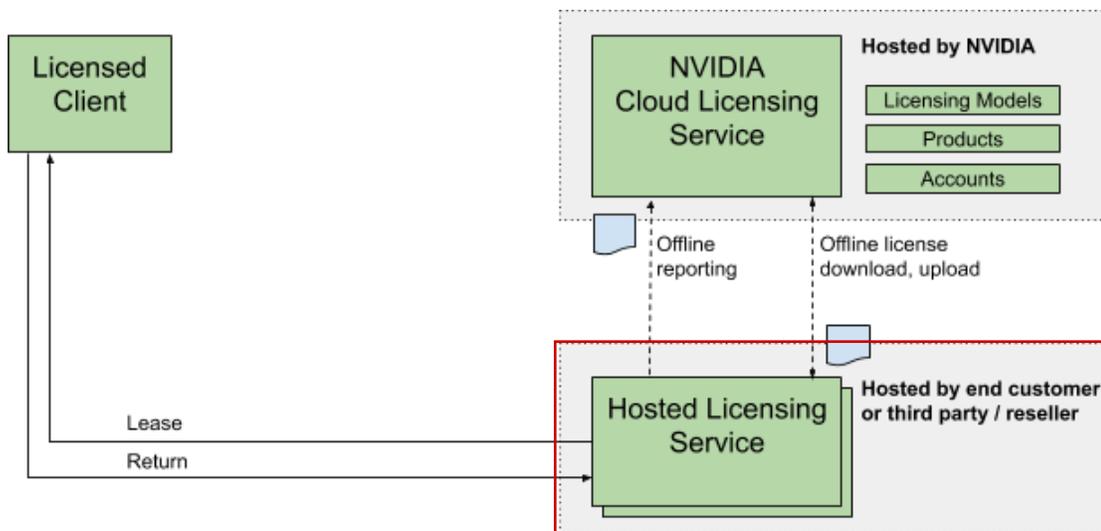
## 1. 授权服务实例类型选择 CLS 还是 DLS

NLS 提供两种 License 服务实例类型来对 vGPU 客户进行软件授权。

- 1) 如果 vGPU 客户端可以访问互联网环境，vGPU 可以访问 NVIDIA 官方服务器上面的 CLS 服务直接进行授权，就可以选择使用 CLS，使用 CLS 则无需在本地环境再搭建授权服务器，因此更加方便和快捷，但要注意公网访问的可靠性。



- 2) 另一种是在企业私网内部署本地 DLS 服务器进行 License 授权。主要面向 vGPU 客户端全部在企业内网，且不能访问互联网环境时应该采用的方式。



## 2. CLS 授权服务器的配置步骤

如果您选择 CLS 服务方式，需要在 NVIDIA license 门户上进行相应配置。

使用您的 NVIDIA 账号访问 <https://nvid.nvidia.com> 添加和配置 CLS 服务。在配置之前，请确认您已经申请了 NLP 账号并获取了测试或者正式 License. 总体部署流程如下：



- 1) 登录用户企业账号管理门户 NLP 后，先创建 License Server。因为使用新的 NLS，不要选择 Create Legacy 方式，然后开始创建，打开 **Express CLS Installation** 选项可以自动创建或者使用默认的 CLS Service 实例，从而简化安装步骤。

NVIDIA LICENSING NVIDIA APPLICATION HUB merlin@nvidia.com NVIDIA INFR-GEN (lic-0011w0000275yiqay) Group: NVIDIA INFR-GEN (lic-0011w0000275yiqay) logout

DASHBOARD ENTITLEMENTS LICENSE SERVERS 1 LIST SERVERS CREATE SERVER 2 NETWORK ENTITLEMENTS VIRTUAL GROUPS USER MANAGEMENT SOFTWARE DOWNLOADS EVENTS LEASES SERVICE INSTANCES API KEYS SUPPORT

### Create License Server [Help?](#)

Create a license server in NVIDIA INFR-GEN (lic-0011w0000275yiqay) / Group: NVIDIA INFR-GEN (lic-0011w0000275yiqay)

Create legacy server If the license server is to be installed on a legacy licensing system server (pre-NLS), enable "Create legacy server"

1 Basic details → 2 Select features → 3 Preview server creation

Enter a name, and description for this new license server

Name: Merlin-CLS-2022

Description: Test Only

Express CLS Installation? The server will be installed on the default CLS service instance

Next: Select features →

- 2) 下面添加您现有的 License 类型和分配的数量。

ENTITLEMENTS LICENSE SERVERS LIST SERVERS CREATE SERVER NETWORK ENTITLEMENTS VIRTUAL GROUPS USER MANAGEMENT SOFTWARE DOWNLOADS EVENTS LEASES SERVICE INSTANCES

### Create License Server [Help?](#)

Create a license server in NVIDIA INFR-GEN (lic-0011w0000275yiqay) / Group: NVIDIA INFR-GEN (lic-0011w0000275yiqay)

Create legacy server If the license server is to be installed on a legacy licensing system server (pre-NLS), enable "Create legacy server"

Basic details → 2 Select features → 3 Preview server creation

Select one or more entitlement features to add to the new license server

workstation

NAME	PRODUCT KEY ID	STATUS	START DATE	EXPIRATION	AVAILABLE	ADDED
NVIDIA RTX Virtual Workstation-5.0	1ug0mlsxzf-rkax3frpibrvq@vgxn	Active	Nov 11, 2021	Nov 11, 2022 66	31	20

预览并创建：

### Create License Server [? Help?](#)

Create a license server in **NVIDIA INFR-GEN (lic-0011w000027i5yjqay)** / Group **NVIDIA INFR-GEN (lic-0011w000027i5yjqay)**

Create legacy server ⓘ If the license server is to be installed on a legacy licensing system server (pre-NLS), enable "Create legacy server"

Basic details → | Select features → | **3 Preview server creation**

ⓘ Review your selections for this license server

You are about to create a server in **NVIDIA INFR-GEN (lic-0011w000027i5yjqay)** / Group **NVIDIA INFR-GEN (lic-0011w000027i5yjqay)** with the following details:

**Server name and description**

 **Merlin-CLS-2022**

Test Only

✔ This server will be bound and installed on the default CLS service instance

View details after creation

**CREATE SERVER** ⓘ

With **3** feature(s) across **3** product key id(s)  Table view

Search Features

FEATURE	LICENSES	PRODUCT KEY ID	STATUS	START DATE
NVIDIA RTX Virtual Workstation-5.0	20	7uglmiaxof-4kux-0tqpl- kzwqjgqan	Active	Nov 11, 2021
NVIDIA RTX Virtual Workstation-5.0	12	7yldgshpc-4w-0l6vna-7f- kpwkltgqg	Active	Aug 29, 2022

### License Server Details [? Help?](#)

View details of license server in **NVIDIA INFR-GEN (lic-0011w000027i5yjqay)** / Group **NVIDIA INFR-GEN (lic-0011w000027i5yjqay)**

[REFRESH](#) [ACTIONS](#)

 **Merlin-CLS-2022 is ENABLED**

Status: **ENABLED** Type: **NVIDIA** Created: **Sep 6, 2022 6:56 PM** Modified: **Sep 6, 2022 6:56 PM**

Service Instance: **0011w000027i5yjqay-2022-09-06\_10-56** [CLS](#) Install Status: **INSTALLED**

Description: **Test Only**

**Overview** | Server Features | License Pools | Fulfillment Conditions | Leases

**ABOUT THIS SERVER**

 License server Merlin-CLS-2022 is enabled and will serve leases, you can not make changes while the server is enabled [DISABLE SERVER](#)

上面是创建后的授权服务器，也可以看到本例中生成的 CLS Service Instance 名称。此时 CLS 已经运行就绪。

### 3) 修改 CLS Service Instance 的名称 (可选)

选择 Express CLS Installation 选项会自动创建 CLS Service Instance (如果之前已经有默认的实例则使用默认项, 新用户因为没有 CLS Service Instance 所有会自动创建一个实例并设为默认) 并绑定 License Server, 自动创建的实例名称可能不易于记忆, 您可以在 Service Instances 表单中找到它改名。这样更方便按名字查找实例, 方便以后创建 Client Token。

The screenshot shows the 'Service Instances' page in a web application. The left sidebar contains navigation items like 'DASHBOARD', 'ENTITLEMENTS', 'LICENSE SERVERS', etc. The main content area has a header 'Service Instances' and a table of instances. The first instance is '0011w00027i5iyqay-2022-09-06\_10-56' with environment 'CLS' and status 'Default'. Its 'Actions' menu is open, showing options like 'Generate client config token', 'Edit', 'Settings', and 'Delete'.

NAME	ENVIRONMENT	STATUS	DATE CREATED
0011w00027i5iyqay-2022-09-06_10-56 (fa58aa40-e663-4600-a6b9-adaaed0b5297)	CLS	Default	Registered
RKEATING_VGPU_2022-09-02 (4bea5abc-ede7-4887-8710-99b37126a0f1)	DLS	Registered	Registered
JL-TEST (3109c285-46a0-4b20-89f1-0fbc2247776f)	DLS	Registered	Registered

The screenshot shows the 'Edit Service Instance' dialog box. The 'Name' field is filled with 'Merlin-CLSi-2022' and the 'Description' field contains 'Auto generated service instance for Merlin Test'. There is a 'Mark as default' checkbox and an 'EDIT SERVICE INSTANCE' button.

The screenshot shows the 'Service Instances' page after the name change. The first instance is now 'Merlin-CLSi-2022' with environment 'CLS' and status 'Default'.

NAME	ENVIRONMENT	STATUS	DATE CREATED
Merlin-CLSi-2022 (fa58aa40-e663-4600-a6b9-adaaed0b5297)	CLS	Default	Registered
RKEATING_VGPU_2022-09-02 (4bea5abc-ede7-4887-8710-99b37126a0f1)	DLS	Registered	Registered

可以看到上面的实例名已修改为 Merlin-CLSi-2022。

#### 4) 回到 License Server 页面点击查看刚刚创建好的 License Server 的详细信息。

View license servers in NVIDIA INFR-GEN (lic-0011w0000z/rbyqay) / Group NVIDIA INFR-GEN (lic-0011w0000z/rbyqay)  
Clicking on a row will display the related server features, clicking on the server name will display the full server details.

ACTIVE DISABLED CLS DLS PENDING INSTALL UNBOUND

Search license servers updated 8:23:17 PM

NAME	FAMILY	SERVICE INSTANCE
Merlin-CLS-2022 <b>Installed</b>	vGPU	Merlin-CLS-2022 (Cloud)

可以看到当前服务带有的 vGPU License 和数量，以及默认规划的 License 池，默认池中有所有的 License，可直接使用这个池，也可以拆分成不同的池给不同的部门使用。

Merlin-CLS-2022 is ENABLED

Status: **ENABLED** Type: NVIDIA Created: Sep 6, 2022 6:56 PM Modified: Sep 6, 2022 8:22 PM

Service Instance: Merlin-CLS-2022 CLS Install Status: **INSTALLED**

Description: Test Only

Overview **Server Features** **License Pools** Fulfillment Conditions Leases

Search license pools updated 8:25:24 PM

NAME	STATUS
Initial LP	ENABLED

Search pool features

FEATURE	IN USE / ALLOCATED	EFFECTIVE	EXPIRATION
NVIDIA RTX Virtual Workstation-5.0 1up0mlszf-rkax3f1nl-brvnyiyozn	0 / 20	Nov 11, 2021	Nov 11, 2022
NVIDIA RTX Virtual Workstation-5.0 ycfafrroc-iv5ilvma7Lltdaalk0wuk	0 / 12	Aug 29, 2022	Aug 29, 2023
NVIDIA Virtual Compute Server-9.0 is3ofrzcm-ujdlbdikfn-lbenuomn05	0 / 20	Jun 15, 2022	Jun 15, 2023

5) 到服务实例页面，下载用于 vGPU 客户的 Token 文件。

**Service Instances** Help? ACTIONS

View your service instances in **NVIDIA INFR-GEN (lic-0011w000275yiqay)**

CLS DLS

Search service instances updated 8:32:07 PM

NAME	ENVIRONMENT	STATUS	DATE CREATED
<b>Merlin-CLSi-2022</b> (fa58aa40-e663-4600-a6b9-adaaed0b5297)	CLS <b>Default</b>	Registered	Sep 6, 2022 6:56 PM
test (819f5d11-838d-40a9-be80-d3d96a8e1157)	CLS	Registered	Jul 19, 2022 10:...
ig_instance (9c24d69c-203a-4143-84ef-c75a845b17e4)	CLS	Registered	Jul 6, 2022 1:20...

Actions: Generate client config token, Edit, Settings, Delete

## 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
<b>Merlin-CLS-2022</b>	18898a72-b2a3-4526-b31f-4df0c501049a

(1 - 1 of 1 scope references) 1 of 1 pages

**DOWNLOAD CLIENT CONFIGURATION TOKEN**



[client\\_configuration\\_token\\_09-06-2022-20-35-40.tok](#)

blob:https://ui.licensing.nvidia.com/ef9ae5a1-6b0b-4371

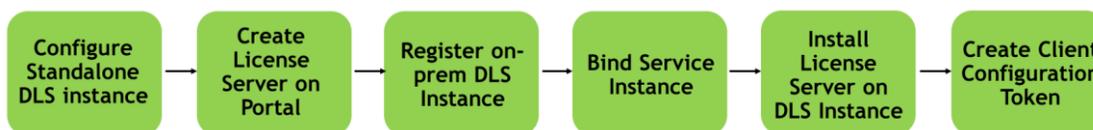
[Show in folder](#)

生成 License 配置 Token 以后，下载该 tok 文件，后面复制到客户端授权使用。

将 Tok 文件复制到 vGPU 客户端 VM 内。然后参见《配置 vGPU 客户端的 License 授权》章节。

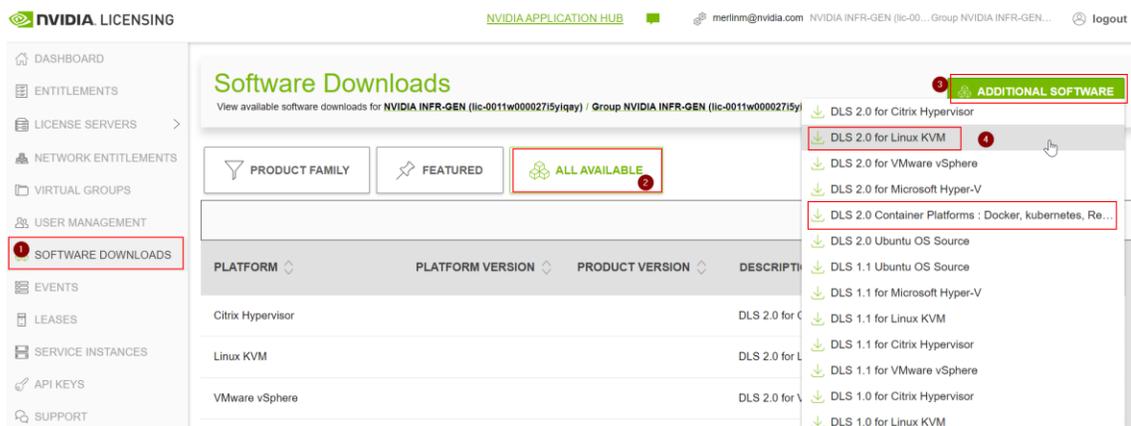
### 3. DLS 本地授权方式的配置步骤

如果您选择 DLS 服务方式，需要在本地部署 DLS 授权服务器虚拟机。然后使用您的 NVIDIA 账号访问 <https://nvid.nvidia.com> 注册您的本地 DLS 服务实例并添加 License Server。在配置之前，请确认已经获取了测试或者正式 License。DLS 部署流程如下：



不同于传统授权服务器的基于 OS 和软件包的安装方法，新的 DLS 的安装直接提供了两种部署方式，一种是**基于虚拟机映像**的安装，另一方式是**通过容器映像来部署** DLS 服务，请**二选一**进行安装。所有安装映像的下载和 vGPU 软件一样要通过企业用户 portal (NLP) 。

#### 1) 虚拟机映像方式部署 DLS 的本地服务



这里以 Linux KVM 平台为例，下载 DLS 2.0 for KVM 软件([nls-2.0.0-bios-KVM.zip](#))。

创建用于 DLS 的虚拟机最小配置为：vCPU:4, MEMORY: 8GB, DISK: 10GB。

KVM 使用 DHCP 分配虚拟机网络地址，并启用 NTP。KVM 环境和 zip 软件包准备好后开始在 KVM Host 上安装：

```
unzip nls-2.0.0-bios-KVM.zip
```

```
cp nls-2.0.0-bios.qcow2 /var/lib/libvirt/images/
```

```
virt-install -v --memory 16384 --vcpus=8 --name dls2-test --import --disk
```

```
/var/lib/libvirt/images/nls-2.0.0-bios.qcow2 --os-variant ubuntu18.04 --network=default
```

```
--noautoconsole --autostart
```

```
[root@kvm ~]# unzip nls-2.0.0-bios-KVM.zip
Archive: nls-2.0.0-bios-KVM.zip
  inflating: nls-2.0.0-bios.qcow2
[root@kvm ~]# mv nls-2.0.0-bios.qcow2 /var/lib/libvirt/images/
[root@kvm ~]# virt-install -v --memory 16384 --vcpus=8 --name dls2-test --import --disk /var/lib/
libvirt/images/nls-2.0.0-bios.qcow2 --os-variant ubuntu18.04 --network=default --noautoconsole --
autostart

Starting install...
Domain creation completed.
[root@kvm ~]# virsh list
-----
 Id   Name                                     State
-----
  2   licserv                                 running
  3   ubun18_000084000_8Q_1                  running
  5   win10-2_000084000_8Q_1                 running
 17   docker2004_000084000_nogpu_1           running
 18   dls2-test                                running
-----

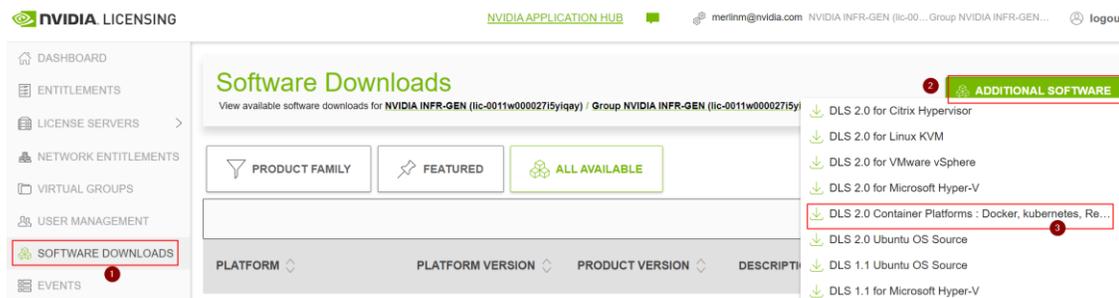
[root@kvm ~]#
[root@kvm ~]# virsh domifaddr dls2-test
-----
 Name      MAC address      Protocol  Address
-----
 vnet4    52:54:00:66:76:16  ipv4     192.168.122.218/24
-----
```

看到 DLS 的虚拟机启动正常，virsh domifaddr 查看 DLS 分配的 IP 地址。  
本案为 192.168.122.218.

下一步请跳转到 [第 3 节《配置 DLS 本地服务》](#)

## 2) 容器方式部署 DLS 的本地服务（可选方式）

下载容器 DLS 的软件包：



这里以 Ubuntu 20.04 为演示平台为例，root 身份执行：

### 1. 安装前准备：

```
apt-get update
apt-get remove docker docker-engine docker.io containerd runc
apt-get install -y ca-certificates curl gnupg unzip lsb-release
```

### 2. 安装 docker 和 docker-compose：

版本要求请务必参见 release notes 中的说明，dls2.0.0 要求 docker version 20.10.17，docker-compose version 2.6.0。



### 3. 部署 NLS 2.0 image:

将软件包 nls-2.0.0-bios.zip 解压到一个目录中。

```
root@docker:~# ls -l
total 376864
-rw-r--r-- 1 root root 385896858 Sep  7 14:50 nls-2.0.0-bios.zip
drwxr-xr-x 3 root root    4096 Apr 20  2021 snap
root@docker:~# mkdir DLS
root@docker:~# cd DLS/
root@docker:~/DLS# unzip ../nls-2.0.0-bios.zip
Archive:  ../nls-2.0.0-bios.zip
  inflating: dls_appliance_2.0.0.tar.gz
  inflating: dls_pgsql_2.0.0.tar.gz
  inflating: docker-compose.yml
root@docker:~/DLS#
```

加载容器映像:

```
docker load --input dls_pgsql_2.0.0.tar.gz
```

```
docker load --input dls_appliance_2.0.0.tar.gz
```

```
root@docker:~/DLS# ls
dls_appliance_2.0.0.tar.gz dls_pgsql_2.0.0.tar.gz docker-compose.yml
root@docker:~/DLS# docker load --input dls_pgsql_2.0.0.tar.gz && docker load --input dls_appliance_2.0.0.tar.gz
8d3ac3489996: Loading layer [=====] 5.866MB/5.866MB
6cab14f8a434: Loading layer [=====] 12.8kB/12.8kB
b737c2580132: Loading layer [=====] 2.048kB/2.048kB
82de5388a1ef: Loading layer [=====] 199MB/199MB
1921cd61465c: Loading layer [=====] 52.22kB/52.22kB
19296757164e: Loading layer [=====] 2.56kB/2.56kB
b98b052dab33: Loading layer [=====] 3.584kB/3.584kB
49f63e2b4713: Loading layer [=====] 15.36kB/15.36kB
60052ce797ae: Loading layer [=====] 5.632kB/5.632kB
717d90867840: Loading layer [=====] 10.75kB/10.75kB
1f6fc160855f: Loading layer [=====] 3.584kB/3.584kB
33c1559c7d63: Loading layer [=====] 5.632kB/5.632kB
1d42f54750bc: Loading layer [=====] 21.5kB/21.5kB
8b9d6412e5e5: Loading layer [=====] 111.6kB/111.6kB
b5682f23de0d: Loading layer [=====] 160.3kB/160.3kB
c01eb6088a7a: Loading layer [=====] 2.033MB/2.033MB
8e15a5b599df: Loading layer [=====] 194kB/194kB
243e5ba7577h: Loading layer [=====] 18.43kB/18.43kB
```

设定环境变量，并用 docker-compose 启动 DLS 服务。操作必须在上面软件及压缩的目录中进行(含 docker-compose.yml 文件)。

```
DLS_PUBLIC_IP='192.168.122.170' docker-compose up
```

容器对外提供服务的地址是该容器主机的 IP 地址。因此只需要改变主机地址，和启动 docker-compose 时的 *DLS\_PUBLIC\_IP* 环境变量即可。

也可以使用 -d 参数将容器服务放在后台运行。

```
DLS_PUBLIC_IP='192.168.122.170' docker-compose up -d
```

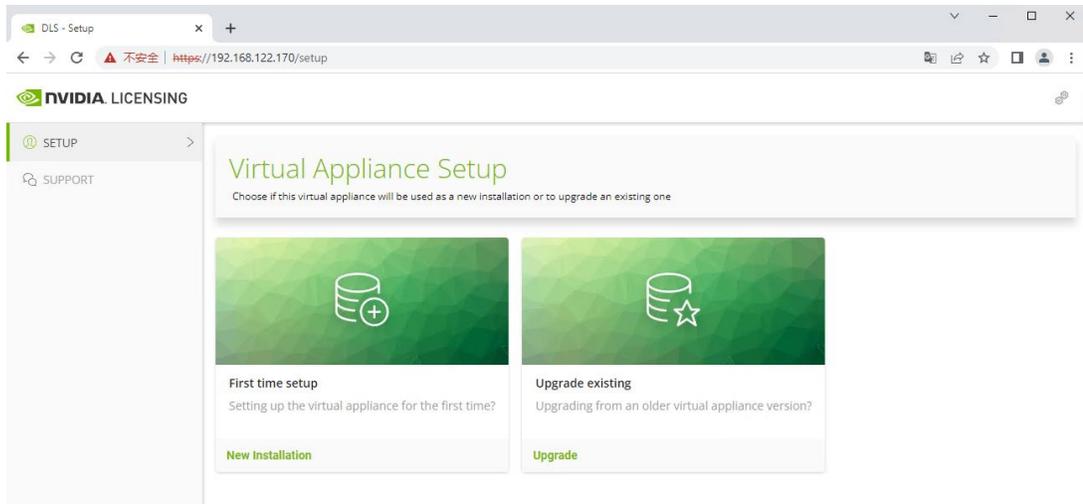
```
root@docker:~/DLS# DLS_PUBLIC_IP='192.168.122.170' docker-compose up
WARN[0000] The "DLS_PRIVATE_HOSTNAME" variable is not set. Defaulting to a blank string.
WARN[0000] The "FQDN" variable is not set. Defaulting to a blank string.
WARN[0000] The "DLS_PRIVATE_HOSTNAME" variable is not set. Defaulting to a blank string.
[+] Running 7/7
  # Network dls_back-tier                Created
  # Volume "postgres-data"              Created
  # Volume "configurations"             Created
  # Volume "rabbitmq_data"              Created
  # Volume "logs"                       Created
  # Container dls-postgres-nls-si-0-1   Created
  # Container dls-nls-si-0-1            Created
Attaching to dls-nls-si-0-1, dls-postgres-nls-si-0-1
dls-postgres-nls-si-0-1 | The files belonging to this database system will be owned by user "postgres".
dls-postgres-nls-si-0-1 | This user must also own the server process.
dls-postgres-nls-si-0-1 |
dls-postgres-nls-si-0-1 | The database cluster will be initialized with locale "en_US.utf8".
dls-postgres-nls-si-0-1 | The default database encoding has accordingly been set to "UTF8".
dls-postgres-nls-si-0-1 | The default text search configuration will be set to "english".
dls-postgres-nls-si-0-1 |
dls-postgres-nls-si-0-1 | Data page checksums are disabled.
dls-postgres-nls-si-0-1 |
dls-postgres-nls-si-0-1 | fixing permissions on existing directory /var/lib/postgresql/data ... ok
dls-postgres-nls-si-0-1 | creating subdirectories ... ok
```

首次运行会初始化数据库，之后重新运行时信息如下：

```
root@docker:~/DLS#
root@docker:~/DLS# DLS_PUBLIC_IP='192.168.122.170' docker-compose up
WARN[0000] The "DLS_PRIVATE_HOSTNAME" variable is not set. Defaulting to a blank string.
WARN[0000] The "FQDN" variable is not set. Defaulting to a blank string.
WARN[0000] The "DLS_PRIVATE_HOSTNAME" variable is not set. Defaulting to a blank string.
[+] Running 2/0
  # Container dls-postgres-nls-si-0-1   Completed
  # Container dls-nls-si-0-1            Completed
Attaching to dls-nls-si-0-1, dls-postgres-nls-si-0-1
dls-postgres-nls-si-0-1 | PostgreSQL Database directory appears to contain a database; Skipping initialization
dls-postgres-nls-si-0-1 |
dls-postgres-nls-si-0-1 | 2022-09-07 15:05:50.373 UTC [1] LOG:  listening on IPv4 address "0.0.0.0", port 5432
dls-postgres-nls-si-0-1 | 2022-09-07 15:05:50.373 UTC [1] LOG:  listening on IPv6 address "::", port 5432
dls-postgres-nls-si-0-1 | 2022-09-07 15:05:50.380 UTC [1] LOG:  listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
dls-postgres-nls-si-0-1 | 2022-09-07 15:05:50.432 UTC [37] LOG:  database system was shut down at 2022-09-07 15:05:45 UTC
dls-postgres-nls-si-0-1 | 2022-09-07 15:05:50.438 UTC [1] LOG:  database system is ready to accept connections
dls-nls-si-0-1 | 2022-09-07 15:05:52.099 WARN For [program:va], redirect_stderr=true but stderr_logfile has also been set to a filename, the filename has been ignored
dls-nls-si-0-1 | 2022-09-07 15:05:52.100 WARN For [program:auth], redirect_stderr=true but stderr_logfile has also been set to a filename, the filename has been ignored
dls-nls-si-0-1 | 2022-09-07 15:05:52.100 WARN For [program:serviceinstance], redirect_stderr=true but stderr_logfile has also been set to a filename, the filename has been ignored
dls-nls-si-0-1 | 2022-09-07 15:05:52.100 WARN For [program:admin], redirect_stderr=true but stderr_logfile has also been set to a filename, the filename has been ignored
dls-nls-si-0-1 | 2022-09-07 15:05:52.100 WARN For [program:lease], redirect_stderr=true but stderr_logfile has also been set to a filename, the filename has been ignored
dls-nls-si-0-1 | 2022-09-07 15:05:52.100 WARN For [program:fileinstallation], redirect_stderr=true but stderr_logfile has also been set to a filename, the filename has been ignored
dls-nls-si-0-1 | 2022-09-07 15:05:52.109 CRIT Server 'inet_http_server' running without any HTTP authentication checking
```

浏览器访问该容器 docker 主机地址：

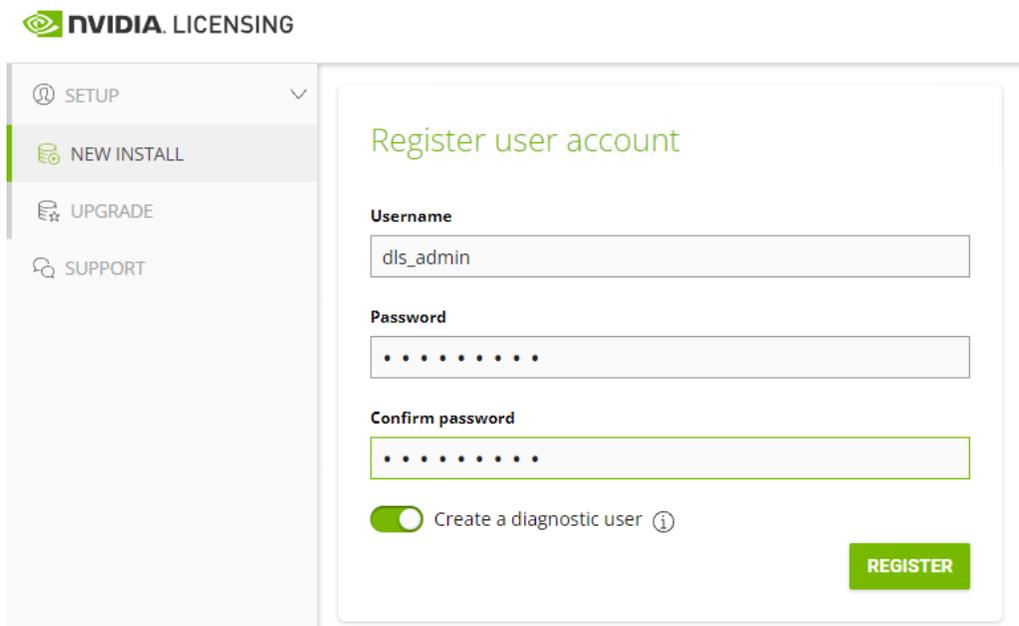
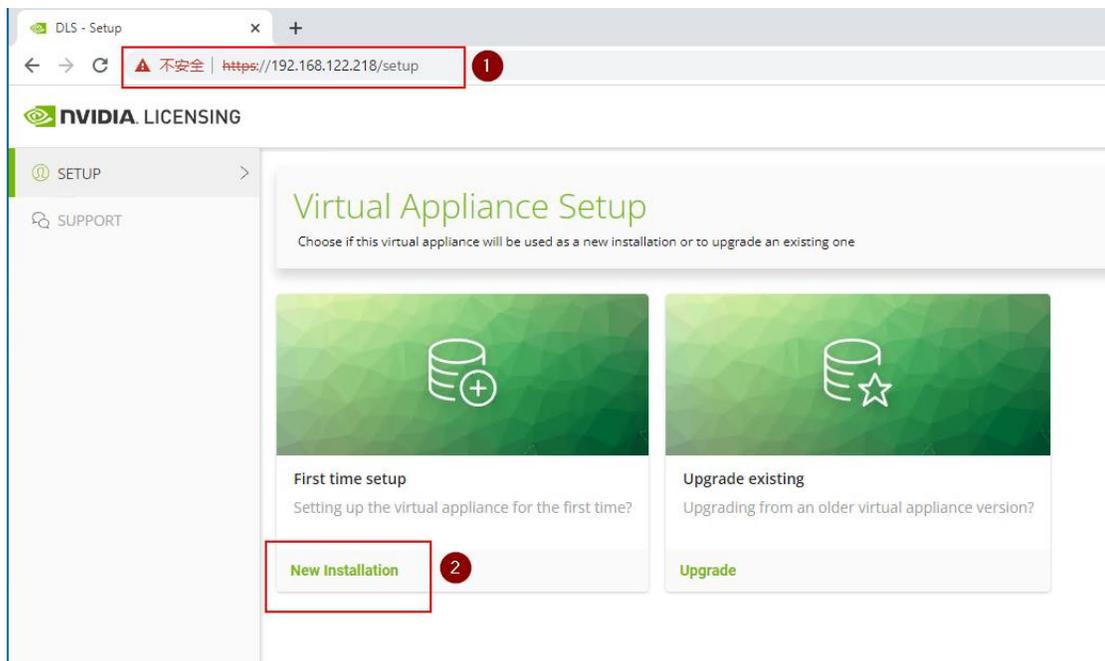
<https://docker-host-ip>



下面开始配置 DLS 服务过程。

### 3) 配置 DLS 本地服务

用浏览器访问 DLS 控制台 <https://192.168.122.218>, 选择新安装。



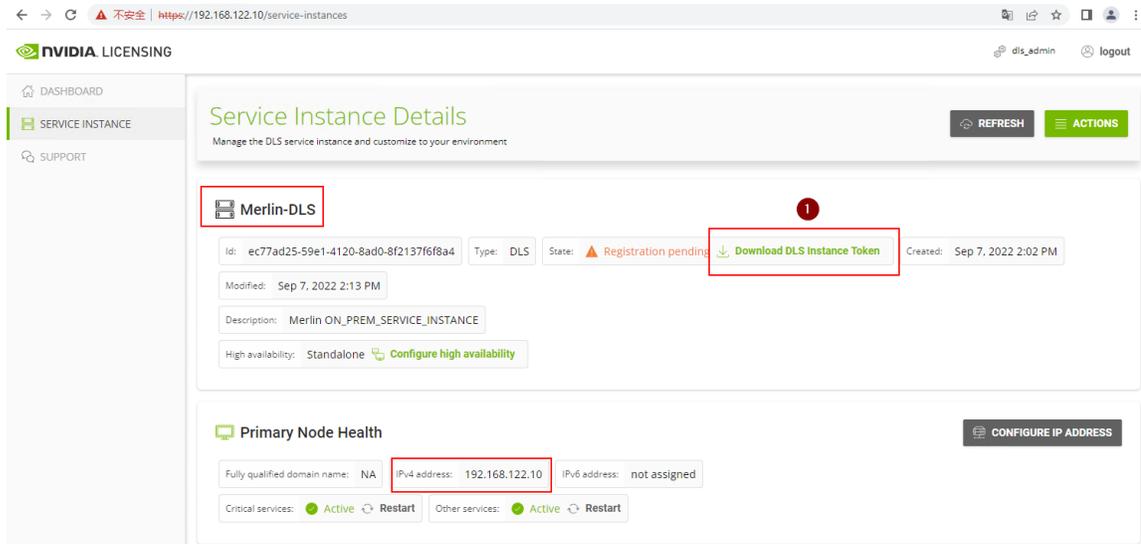
按提示操作，设置口令，登录名为 dls\_admin.

The screenshot shows the 'Service Instance Details' page for a DLS instance named 'DEFAULT\_2022-09-07\_06:02:07'. The instance ID is 'ec77ad25-59e1-4120-8ad0-8f2137f6f8a4' and its state is 'Registration pending'. A red circle '1' highlights the 'ACTIONS' menu, which includes options like 'Edit', 'Download DLS Instance Token', 'Pre-Register Service Instance', 'Log Archive Settings', 'SSL Configuration', 'NTP Server Configuration', 'Generate client config token', and 'Settings'. A red circle '2' highlights the 'Edit' option. Below the instance details, the 'Primary Node Health' section shows the fully qualified domain name 'nls-si-0', IPv4 address '192.168.122.218', and IPv6 address 'not assigned'. A red circle '3' highlights the 'CONFIGURE IP ADDRESS' button.

上面是创建完成的 DLS 实例，可以根据您的偏好更改 IP 静态地址以及实例名称。

The screenshot shows the 'Configure Node IP Address' dialog box. The title is 'Configure Node IP Address' and it includes a warning: 'The IP address must be in the subnet range of the node's virtual network'. The dialog contains several input fields: 'Static IP Address' (192.168.122.10), 'Gateway (Will use DHCP settings if left blank)' (192.168.122.1), 'Netmask Prefix (CIDR notation e.g. 24 for 255.255.255.0, 28 for 255.255.255.240)' (24), 'DNS Server' (192.168.122.1), and 'DNS Server' (DNS Server Two). A warning icon and text state: 'The configuration takes a few minutes to complete after triggering'. A green button labeled 'SET NODE IP ADDRESS' is at the bottom.

改好固定 IP 以后重新登录到新的 IP 地址。



核对全部信息正确后，点击“DOWNLOAD DLS INSTANCE TOKEN”按钮，下载 DLS 实例 Tok 文件。

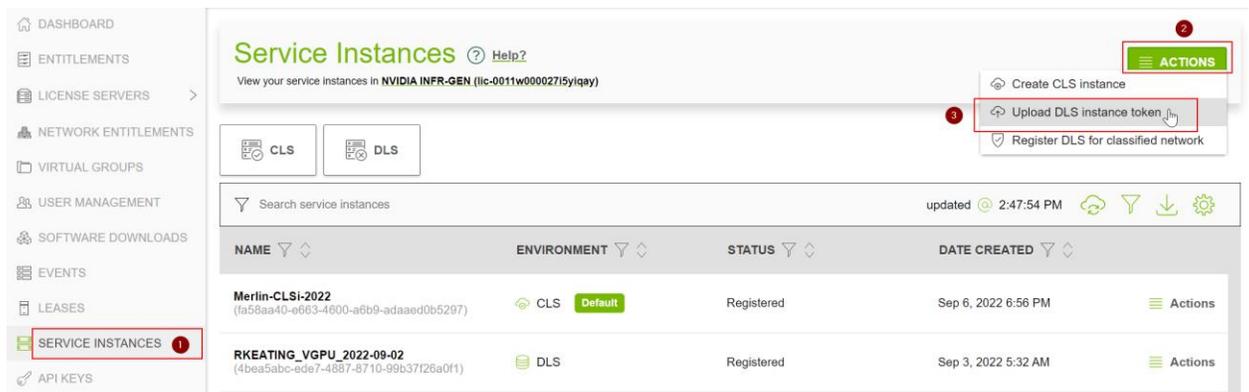
`dls_instance_token_09-07-2022-14-18-42.tok`

`blob:https://192.168.122.10/166059ea-e75f-44fc-a030`

准备将此文件提交到 NVIDIA NLP 企业用户门户。

4) 返回 NVIDIA NLP 企业用户门户 <https://nvid.nvidia.com>

在 Service Instances 页面上传上一步生成的 Tok 文件，注册该 DLS 实例：



### Upload DLS Instance Token

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

**1**

**2** SELECT INSTANCE TOKEN **3** UPLOAD TOKEN

### Upload DLS Instance Token

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

DLS\_INSTANCE\_TOKEN\_09-07-2022-1...

**1** UPLOAD TOKEN

### Service Instances

View your service instances in NVIDIA INFR-GEN (lic-0011w000027f5yiqay)

CLS DLS

Search service instances updated 3:11:38 PM

NAME	ENVIRONMENT	STATUS	DATE CREATED	Actions
Merlin-DLS (ec77ad25-59e1-4120-8ad0-8f2137f6f8a4)	DLS	Pending	Sep 7, 2022 3:11 PM	<ul style="list-style-type: none"> <li>Register</li> <li>Reject</li> <li>Delete</li> </ul>
Merlin-CLSi-2022 (fa58aa40-e663-4600-a6b9-adaaed0b5297)	CLS <b>Default</b>	Registered	Sep 6, 2022 6:56 PM	

NAME	ENVIRONMENT	STATUS	DATE CREATED	Actions
Merlin-DLS (ec77ad25-59e1-4120-8ad0-8f2137f6f8a4)	DLS	Registered	Sep 7, 2022 3:11 PM	Actions

上面看到注册成功的 Merlin-DLS 本地 DLS 空的实例。下一步要创建 License Server，然后绑定到此 DLS 服务实例。

## 5) 创建 License Server

- DASHBOARD
- ENTITLEMENTS
- LICENSE SERVERS
- LIST SERVERS
- CREATE SERVER**
- NETWORK ENTITLEMENTS
- VIRTUAL GROUPS
- USER MANAGEMENT
- SOFTWARE DOWNLOADS
- EVENTS
- LEASES
- SERVICE INSTANCES
- API KEYS
- SUPPORT

### Create License Server

Create a license server in NVIDIA INFR-GEN (lic-0011w000027f5yiqay) / Group NVIDIA INFR-GEN (lic-0011w000027f5yiqay)

Create legacy server If the license server is to be installed on a legacy licensing system server (pre-NLS), enable "Create legacy server"

**1** Basic details → | Select features → | Preview server creation

Enter a name, and description for this new license server

**Name**  
Merlin-DLS-2022

**Description**  
Merlin DLS test 2022

Express CLS Installation?

**3** Next: Select features →

创建 License Server 时需要指定服务命名并添加所需的 License 类型以及 License 数量。这里添加了 5 个 vWS 类型许可。然后选择创建服务器。

## Create License Server [Help?](#)

Create a license server in **NVIDIA INFR-GEN (lic-0011w00002715yiqay)** / **Group NVIDIA INFR-GEN (lic-0011w00002715yiqay)**

Create legacy server  ⓘ If the license server is to be installed on a legacy licensing system server (pre-NLS), enable "Create legacy server"

1 Basic details → 2 **Select features** → 3 Preview server creation

ⓘ Select one or more entitlement features to add to the new license server

NAME	PRODUCT KEY ID	STATUS	START DATE	EXPIRATION	AVAILABLE	ADDED
<input checked="" type="checkbox"/> NVIDIA RTX Virtual Workstation-5.0	1ug0mlszf-rkax3frtpi-brpvqjvgn	Active	Nov 11, 2021	Nov 11, 2022	65	7

You are about to create a server in **NVIDIA INFR-GEN (lic-0011w00002715yiqay)** / **Group NVIDIA INFR-GEN (lic-0011w00002715yiqay)** with the following details:

Server name and description



Merlin-DLS-2022

Merlin DLS test 2022

View details after creation

**CREATE SERVER**

With 1 feature(s) across 1 product key id(s)  Table view

Search Features

FEATURE	LICENSES	PRODUCT KEY ID	STATUS	START DATE
NVIDIA RTX Virtual Workstation-5.0	5	1ug0mlszf-rkax3frtpi-brpvqjvgn	Active	Nov 11, 2021

**License Server Details** [? Help?](#) REFRESH ACTIONS

View details of license server in **NVIDIA INFR-GEN (lic-0011w00002715yiqay)** / Group **NVIDIA INFR-GEN (lic-0011w00002715yiqay)**

---

**Merlin-DLS-2022** ^

Type: **NVIDIA** Created: **Sep 7, 2022 2:41 PM** Modified: **Sep 7, 2022 2:41 PM**

Service Instance: **UNBOUND** Install Status: **UNBOUND**

Description: **Merlin DLS test 2022**

**Overview** Server Features License Pools Fulfillment Conditions Leases

**ABOUT THIS SERVER**

This server is not bound to a service instance

- Click 'Express CLS Install' to bind and install this server on the default CLS service instance
- Otherwise, click the 'Bind' button to choose an existing CLS/DLS service instance.

BIND SERVICE INSTANCE EXPRESS CLS INSTALL

上面是创建好的 Merlin-DLS-2022 服务，选择 BIND SERVICE INSTANCE 按钮，绑定本 LS (Merlin-DLS-2022) 到之前注册的名为(Merlin-DLS)的 Service Instance 上。

**Bind Service Instance** ×

Bind this license server to a registered service instance

Select a service instance 1

**ALL** **CLS** **DLS** 2

**Merlin-DLS** 3

**DLS** 4

CREATED Sep 7, 2022 3:11 PM  
ec77ad25-59e1-4120-8ad0-8f2137f68a4

**RKEATING\_VGPU\_2022-09-02**

**DLS**

CREATED Sep 3, 2022 5:32 AM  
4bea5abc-ed7-4887-8710-99b37126a0f1

**License Server Details** [Help?](#)  
View details of license server in **NVIDIA INFR-GEN (lic-0011w000027f5yiqay)** / Group **NVIDIA INFR-GEN (lic-0011w000027f5yiqay)**

**Merlin-DLS-2022**

Type: NVIDIA Created: Sep 7, 2022 2:41 PM Modified: Sep 7, 2022 2:41 PM

Service Instance: **Merlin-DLS** **DLS** Install Status: **PENDING**

Description: Merlin DLS test 2022

Overview **Server Features** License Pools Fulfillment Conditions Leases

Search server features updated @ 3:27:25 PM

FEATURE (PKID)	ASSIGNED / ALLOCATED	LICENSE TYPE	EFFECTIVE	EXPIRATION
NVIDIA RTX Virtual Workstation-5.0 <a href="#">1ug@msxz1-rkax3fr1pi-brpvqiy9zn</a>	0 / 5	CONCURRENT COUNTED SINGLE	Nov 11, 2021	Nov 11, 2022

看到上面绑定成功后的状态，从 Actions 菜单中点击下载按钮，下载用于转移 License 配置的 Bin 文件。

[license\\_09-07-2022-15-31-29.bin](#)



blob:https://ui.licensing.nvidia.com/196c5c67-3af4-4b2d-a174-b07c041efc7b

6) 回到本地 DLS 控制台将 Bin 文件上传到 DLS 完成 License 传输。

← → ↻ 不安全 | https://192.168.122.10

**NVIDIA LICENSING** dis\_admin

**DASHBOARD** 1

SERVICE INSTANCE SUPPORT

**Dashboard**  
Complete basic setup

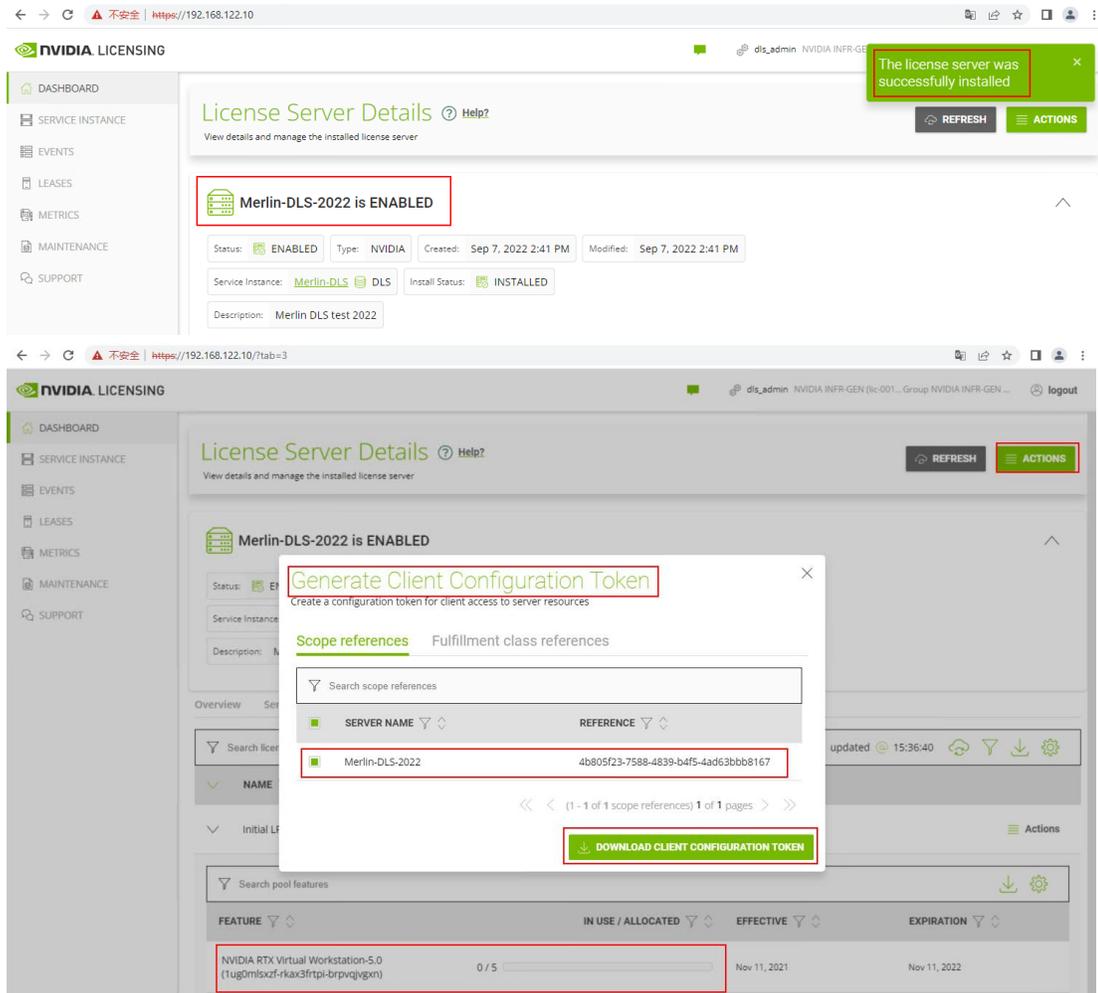
**Service instance not registered**  
This service instance has not yet been registered with the NVIDIA Licensing Portal.

[Download DLS Instance Token](#)

**No license server installed**  
Select the license server bin file downloaded from the NVIDIA Licensing Portal

**SELECT LICENSE SERVER FILE** 2

Install server



上面看到上传 Bin 文件以后，本地 DLS 即获得 License 授权能力。可以为本地 vGPU License 授权。与 CLS 相同，授权仍然使用 Tok 文件方式，需要从本地 DLS 生成并下载 .tok 文件，然后上传到 vGPU 客户端。

#### 7) 下载授权 Token 文件：

生成并下载客户端配置授权文件。

**NVIDIA LICENSING** | dis\_admin | NVIDIA INFR-GEN (lic-001... Group NVIDIA INFR-GEN ... | logout

### License Server Details [Help?](#)

View details and manage the installed license server

**Merlin-DLS-2022 is ENABLED**

Status: **ENABLED** | Type: **NVIDIA** | Created: Sep 7, 2022 2:41 PM | Modified: Sep 7, 2022 2:41 PM

Service Instance: **Merlin-DLS** | DLS | Install Status: **INSTALLED**

Description: Merlin DLS test 2022

Overview | Server Features | **License Pools** | Fulfillment Conditions | Leases

Search license pools | updated @ 15:43:18

NAME	STATUS
Initial LP	ENABLED

Actions

- Disable
- Manage Features
- Update server from NLP
- Create Pool
- Create Condition
- Generate client config token**
- Settings

### 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
Merlin-DLS-2022	4b805f23-7588-4839-b4f5-4ad63bbb8167

(1 - 1 of 1 scope references) 1 of 1 pages

**DOWNLOAD CLIENT CONFIGURATION TOKEN**

**client\_configuration\_token\_09-07-2022-15-47-12.tok**

blob:https://192.168.122.10/097821b0-8f25-47c0-bf80-61158652ae24

在文件夹中显示

将下载的 Tok 文件复制到 vGPU 客户端 VM 内。

之后请参见“[配置 vGPU 客户端的 License 授权](#)”章节。

```
[root@linuxvm_000084000_8Q_1 ~]# mv client_configuration_token_11-15-2021-20-24-24.tok /etc/nvidia/
ClientConfigToken/      gridd.conf.template      nvidia-topologyd.conf.template
gridd.conf              license/
[root@linuxvm_000084000_8Q_1 ~]# mv client_configuration_token_11-15-2021-20-24-24.tok /etc/nvidia/ClientConfigToken/
[root@linuxvm_000084000_8Q_1 ~]# systemctl restart nvidia-gridd.service
[root@linuxvm_000084000_8Q_1 ~]# systemctl status nvidia-gridd.service
● nvidia-gridd.service - NVIDIA Grid Daemon
   Loaded: loaded (/usr/lib/systemd/system/nvidia-gridd.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2021-11-15 20:37:13 CST; 14s ago
     Process: 8763 ExecStopPost=/bin/rm -rf /var/run/nvidia-gridd (code=exited, status=0/SUCCESS)
     Process: 8765 ExecStart=/usr/bin/nvidia-gridd (code=exited, status=0/SUCCESS)
    Main PID: 8766 (nvidia-gridd)
      Tasks: 4 (limit: 49632)
     Memory: 2.9M
     CGroup: /system.slice/nvidia-gridd.service
            └─8766 /usr/bin/nvidia-gridd

Nov 15 20:37:13 linuxvm_000084000_8Q_1 systemd[1]: Stopped NVIDIA Grid Daemon.
Nov 15 20:37:13 linuxvm_000084000_8Q_1 systemd[1]: Starting NVIDIA Grid Daemon...
Nov 15 20:37:13 linuxvm_000084000_8Q_1 nvidia-gridd[8766]: Started (8766)
Nov 15 20:37:13 linuxvm_000084000_8Q_1 systemd[1]: Started NVIDIA Grid Daemon.
Nov 15 20:37:13 linuxvm_000084000_8Q_1 nvidia-gridd[8766]: Configuration parameter ( ServerAddress ) not set
Nov 15 20:37:13 linuxvm_000084000_8Q_1 nvidia-gridd[8766]: vGPU Software package (0)
Nov 15 20:37:13 linuxvm_000084000_8Q_1 nvidia-gridd[8766]: Ignore service provider and node-locked licensing
Nov 15 20:37:13 linuxvm_000084000_8Q_1 nvidia-gridd[8766]: NLS initialized
Nov 15 20:37:13 linuxvm_000084000_8Q_1 nvidia-gridd[8766]: Acquiring license. (Info: 192.168.122.10; NVIDIA RTX Virtua
Nov 15 20:37:15 linuxvm_000084000_8Q_1 nvidia-gridd[8766]: License acquired successfully. (Info: 192.168.122.10; NVIDIA
lines 1-21/21 (END)
```

上面是使用 DLS 授权的 VM 成功状态。

#### 4. 配置 vGPU 客户端的 License 授权

升级到 NLS 以后，对 vGPU 客户端的配置方法也有比较大的改变。之前是通过手工添加 License Server 的主机名称或 IP/端口号来访问服务器。NLS 则是将客户端访问服务器的配置信息全部包含在 tok 文件中。

下载 CLS 或者 DLS 生成的 Token 文件后，即可通过该文件用于客户端的授权来提供 License。只需要向 vGPU 客户端提供 token 文件即可，而无需再指定 License Server 的 IP 地址。下面分 Linux 和 Windows 两种客户端说明：

##### 1) Linux 客户端

对于 Linux 客户端，先在 VM 内安装 13.0 版本以上的 vGPU 驱动程序，并下载好 .tok 客户端授权文件：

cd /etc/nvidia, 复制 gridd.conf.template 为 gridd.conf.

```
[root@linuxvm_000084000_8Q_1 ~]# cd /etc/nvidia/
[root@linuxvm_000084000_8Q_1 nvidia]# ls
ClientConfigToken  gridd.conf.template  license  nvidia-topologyd.conf.template
[root@linuxvm_000084000_8Q_1 nvidia]# cp gridd.conf.template gridd.conf
[root@linuxvm_000084000_8Q_1 nvidia]#
```

编辑 gridd.conf, 只需设置 FeatureType 的值为要请求的 vGPU License 类型编号, **不要**设置其中的 ServerAddress。

```
# Description: Set Feature to be enabled
# Data type: integer
# Possible values:
# 0 => for unlicensed state
# 1 => for NVIDIA vGPU (Optional, autodetected as per vGPU type)
# 2 => for NVIDIA RTX Virtual Workstation
# 4 => for NVIDIA Virtual Compute Server
# All other values reserved
FeatureType=2
```

然后将下载到的 Token 文件复制到 /etc/nvidia/ClientConfigToken 目录中，例如：  
`cp client_configuration_token_11-15-2021-11-53-00.tok /etc/nvidia/ClientConfigToken/`

```
[root@linuxvm_000084000_8Q_1 ~]# ls
client_configuration_token_11-15-2021-11-53-00.tok
[root@linuxvm_000084000_8Q_1 ~]# cp client_configuration_token_11-15-2021-11-53-00.tok /etc/nvidia/ClientConfigToken/
[root@linuxvm_000084000_8Q_1 ~]#
```

`systemctl restart nvidia-gridd` 重启 nvidia-gridd 服务，应看到 vGPU 授权成功。

```
[root@linuxvm_000084000_8Q_1 ~]# systemctl restart nvidia-gridd.service
[root@linuxvm_000084000_8Q_1 ~]# systemctl status nvidia-gridd.service
● nvidia-gridd.service - NVIDIA Grid Daemon
   Loaded: loaded (/usr/lib/systemd/system/nvidia-gridd.service; enabled; vendor preset: disabled)
   Active: active (running) since Mon 2021-11-15 12:40:40 CST; 8s ago
   Process: 26582 ExecStopPost=/bin/rm -rf /var/run/nvidia-gridd (code=exited, status=0/SUCCESS)
   Process: 26583 ExecStart=/usr/bin/nvidia-gridd (code=exited, status=0/SUCCESS)
   Main PID: 26585 (nvidia-gridd)
     Tasks: 4 (limit: 49632)
    Memory: 1.5M
   CGroup: /system.slice/nvidia-gridd.service
           └─26585 /usr/bin/nvidia-gridd

Nov 15 12:40:40 linuxvm_000084000_8Q_1 systemd[1]: Stopped NVIDIA Grid Daemon.
Nov 15 12:40:40 linuxvm_000084000_8Q_1 systemd[1]: Starting NVIDIA Grid Daemon...
Nov 15 12:40:40 linuxvm_000084000_8Q_1 nvidia-gridd[26585]: Started (26585)
Nov 15 12:40:40 linuxvm_000084000_8Q_1 systemd[1]: Started NVIDIA Grid Daemon.
Nov 15 12:40:40 linuxvm_000084000_8Q_1 nvidia-gridd[26585]: Configuration parameter ( ServerAddress ) not set
Nov 15 12:40:40 linuxvm_000084000_8Q_1 nvidia-gridd[26585]: vGPU Software package (0)
Nov 15 12:40:40 linuxvm_000084000_8Q_1 nvidia-gridd[26585]: Ignore service provider and node-locked licensing
Nov 15 12:40:40 linuxvm_000084000_8Q_1 nvidia-gridd[26585]: NLS initialized
Nov 15 12:40:40 linuxvm_000084000_8Q_1 nvidia-gridd[26585]: Acquiring license. (Info: api.cls.licensing.nvidia.)
Nov 15 12:40:48 linuxvm_000084000_8Q_1 nvidia-gridd[26585]: License acquired successfully. (Info: api.cls.licen>
lines 1-21/21 (END)
```

也可以用 `nvidia-smi -q` 查询授权状态：

```
[root@linuxvm_000084000_8Q_1 ~]# nvidia-smi -q |grep -i status
License Status : Licensed (Expiry: 2021-11-16 4:39:24 GMT)
[root@linuxvm_000084000_8Q_1 ~]#
```

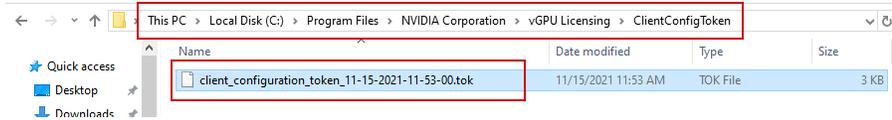
如果授权失败则输出：

```
[root@linuxvm_000084000_8Q_1 nvidia]# nvidia-smi -q |grep -i status
License Status : Unlicensed
```

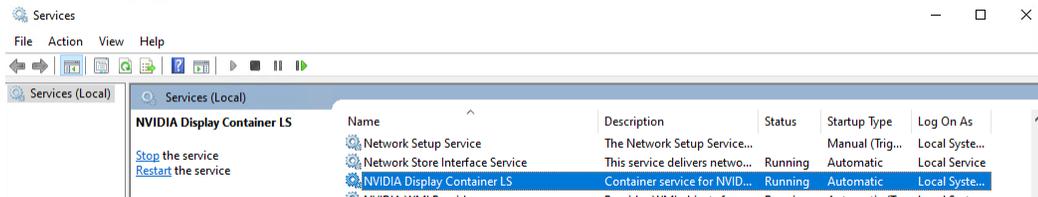
## 2) Windows 客户端

1. 安装 GRID v13.0 以上的 Windows 驱动程序。
2. 将下载的.tok 文件复制到：

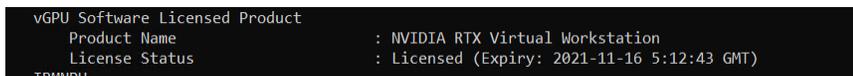
`C:\Program Files\NVIDIA Corporation\vGPU Licensing\ClientConfigToken` 目录



### 3. 重启 NvDisplayContainer 服务。



### 4. C:\Program Files\NVIDIA Corporation\NVSMI\nvidia-smi.exe -q

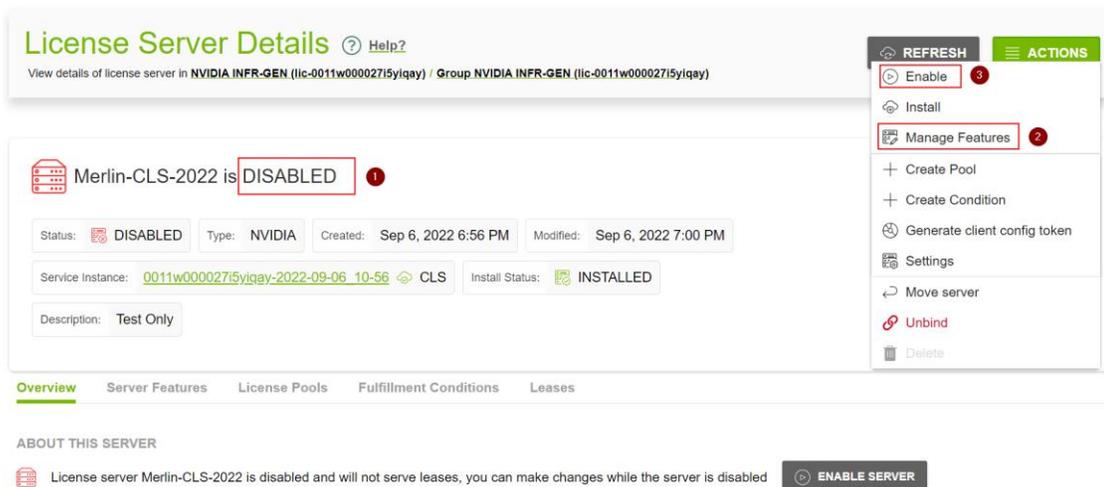


查看 Licensed 状态。

## 5. License Server 的管理

### 1. 调整 CLS 的 License 的类型或者数量：

如果后期需要调整 License Server 上的 License 数量或者类型，需要从 License Server Details 中的 Actions 中 Disable 此服务，然后对 Features 进行修改，最后 Enable 此 License Server。



## 6. 常见问题与排错

### 1. 直通 GPU VM 或者裸金属部署 vGPU 默认 vAPP 类型，而不是 vWS 的 vGPU 类型。

直通 GPU VM 或者裸金属部署 vGPU 需要在注册表内增加以下条目，否则默认会以 vAPP 方式运行（其最大分辨率为 1280\*720）。

将 **FeatureType DWord** (REG\_DWORD) 注册表值添加到 Windows 注册表

[HKEY\\_LOCAL\\_MACHINE\SOFTWARE\NVIDIA Corporation\Global\GridLicensing](#)

其值设为 2

GPU Type	Setting
NVIDIA vGPU	Do <b>not</b> change the value of this registry key. 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. • 0: NVIDIA Virtual Applications • 2: NVIDIA RTX Virtual Workstation