



管理员指南

NICE DCV



NICE DCV: 管理员指南

Copyright © 2024 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon 的商标和商业外观不得用于任何非 Amazon 的商品或服务，也不得以任何可能引起客户混淆、贬低或诋毁 Amazon 的方式使用。所有非 Amazon 拥有的其他商标均为各自所有者的财产，这些所有者可能附属于 Amazon、与 Amazon 有关联或由 Amazon 赞助，也可能不是如此。

Table of Contents

什么是 NICE DCV ?	1
NICE DCV 的工作方式	1
特征	1
定价	2
NICE DCV 服务器	3
要求	3
支持的功能	5
设置	8
安装	8
Windows	8
Linux	14
许可	50
NICE DCV 许可要求	50
安装延期评估许可证	53
安装生产许可证	54
更新生产许可证	65
Upgrading	66
兼容性注意事项	66
在 Windows 上升级 NICE DCV 服务器	66
在 Linux 上升级 NICE DCV 服务器	67
卸载	67
在 Windows 上卸载 NICE DCV 服务器	68
在 Linux 上卸载 NICE DCV 服务器	68
成像	69
建立镜像	69
添加到图像管道	70
管理服务器	72
启动服务器	72
停止服务器	74
启用 QUIC UDP	75
更改 TCP/UDP 端口和地址	76
更改服务器 TCP/UDP 端口	77
侦听特定的终端节点	79
管理 TLS 证书	81

断开连接空闲的客户端	83
在 Linux 上启用 GPU 共享	85
启用触摸屏和触控笔支持	87
配置触控笔	89
启用游戏手柄支持	90
支持 Xbox 360 控制器	90
启用 USB 远程控制	91
配置智能卡缓存	93
配置 WebAuthn 重定向	94
设置 WebAuthn 重定向浏览器扩展	95
启用会话存储	97
在 Windows 上启用会话存储	97
在 Linux 上启用会话存储	98
在 Linux 上配置打印机	99
解决打印机问题	100
在 Linux 上配置剪贴板	101
将客户端剪贴板内容粘贴到主选区	101
将主选区内容复制到客户端剪贴板	102
配置多声道音频	103
在 Windows NICE DCV 服务器上配置音频声道	105
在 Linux NICE DCV 服务器上配置音频声道	105
配置 HTTP 标头	106
在 Windows NICE DCV 服务器上配置 HTTP 标头	107
在 Linux NICE DCV 服务器上配置 HTTP 标头	107
配置身份验证	108
在 Windows 上配置身份验证	108
在 Linux 上配置身份验证	110
配置使用外部身份验证器的身份验证	111
配置授权	111
默认权限文件	112
自定义权限文件	112
使用权限文件	112
管理会话	118
NICE DCV 会话简介	118
控制台会话	119
虚拟会话	119

使用命令行工具管理会话	120
在 Windows NICE DCV 服务器上使用命令行工具	120
在 Linux NICE DCV 服务器上使用命令行工具	121
命令行工具的使用	121
启动会话	122
手动启动控制台会话和虚拟会话	123
启用自动控制台会话	127
停止会话	129
语法	129
示例	129
管理运行的会话	130
管理会话存储	130
管理会话授权	131
管理会话显示布局	133
管理会话名称	135
管理会话时区	137
查看会话	138
列出所有活动会话	138
查看有关特定会话的信息	138
获取屏幕截图	140
语法	140
Options	140
示例	142
如何...	143
使用外部身份验证	143
NICE DCV 服务器配置	144
使用令牌	145
身份验证服务要求	145
查找并停止空闲会话	147
启用到 X 服务器的远程 X 连接	148
启用到 X 服务器的远程 X 连接	148
将 NICE DCV Web 浏览器客户端嵌入到 iFrame 中	150
故障排除	152
使用日志文件	152
在 Windows 上更改日志文件详细程度	153
在 Linux 上更改日志文件详细程度	153

排除在 Linux 上创建虚拟会话时发生的故障	154
调查在 Linux 上创建虚拟会话失败的问题	154
在 Linux 上创建故障防护虚拟会话	155
在更改 UID 后，Linux 会话无法启动	156
修复 Windows 上的光标问题	157
修复复制并粘贴到 IntelliJ IDEA 的问题	157
自签名证书的重定向说明	158
Windows 上的 NVIDIA GPU 发生多显示器/全屏故障	159
监控 NICE DCV 性能和统计信息	159
计数器集	160
参数参考	168
audio 参数	169
clipboard 参数	170
connectivity 参数	172
display 参数	176
display/linux 参数	181
input 参数	182
license 参数	183
log 参数	184
printer 参数	187
redirection 参数	189
security 参数	190
session-management 参数	197
session-management/automatic-console-session 参数	200
session-management/defaults 参数	202
smartcard 参数	203
webcam 参数	204
windows 参数	204
修改配置参数	205
Windows NICE DCV 服务器	205
Linux NICE DCV 服务器	206
NICE DCV 支持生命周期结束	208
EOSL 时间表	208
客户的 EOSL 途径	209
EOSL 常见问题解答	209
安全性	211

数据保护	211
数据加密	212
合规性验证	212
发行说明和文档历史记录	214
发布说明	214
NICE DCV 2023.1-16388	216
NICE DCV 2023.1-16388	217
NICE DCV 2023.1-16220	218
NICE DCV 2023.0-15487	220
NICE DCV 2023.0-15065	221
NICE DCV 2023.0-15022	222
NICE DCV 2023.0-14852	224
NICE DCV 2022.2-14521	225
NICE DCV 2022.2-14357	226
NICE DCV 2022.2-14175	226
NICE DCV 2022.2-14126	227
NICE DCV 2022.2-13907	228
NICE DCV 2022.1-13300	229
NICE DCV 2022.1-13216	229
NICE DCV 2022.1-13067	230
NICE DCV 2022.0-12760	231
NICE DCV 2022.0-12627	232
NICE DCV 2022.0-12123	232
NICE DCV 2022.0-11954	233
NICE DCV 2021.3-11591	234
NICE DCV 2021.2-11445	234
NICE DCV 2021.2-11190	235
NICE DCV 2021.2-11135	235
NICE DCV 2021.2-11048	236
DCV 2021.1-10851	238
DCV 2021.1-10598	238
DCV 2021.1-10557	239
DCV 2021.0-10242	239
DCV 2020.2-9662	240
DCV 2020.2-9508	241
DCV 2020.1-9012	242

DCV 2020.1-9012	242
DCV 2020.1-8942	242
DCV 2020.0-8428	244
DCV 2019.1-7644	245
DCV 2019.1-7423	245
DCV 2019.0-7318	245
DCV 2017.4-6898	247
DCV 2017.3-6698	248
DCV 2017.2-6182	249
DCV 2017.1-5870	251
DCV 2017.1-5777	251
DCV 2017.0-5600	252
DCV 2017.0-5121	252
DCV 2017.0-4334	253
DCV 2017.0-4100	253
文档历史记录	254
.....	cclviii

什么是 NICE DCV ?

NICE DCV 是一种高性能远程显示协议。它允许您在不同的网络条件下，将远程桌面和应用程序流从任何云或数据中心安全地传送到任何设备。通过将 NICE DCV 与 Amazon EC2 一起使用，您可以在 Amazon EC2 实例上远程运行图形密集型应用程序。然后，您可以将结果流式传输到更适中的客户端计算机，从而消除对昂贵的专用工作站的需求。

主题

- [NICE DCV 的工作方式](#)
- [NICE DCV 功能](#)
- [NICE DCV 定价](#)

NICE DCV 的工作方式

要使用 NICE DCV，请在服务器上安装 NICE DCV 服务器软件。NICE DCV 服务器软件用于创建安全会话。在服务器上安装并运行应用程序。服务器使用其硬件执行安装的应用程序所需的高性能处理。您的用户使用 NICE DCV 客户端应用程序远程连接到会话以访问该应用程序。在建立连接后，NICE DCV 服务器软件压缩该应用程序的可视输出，并以加密的像素流将其流式传输回客户端应用程序。客户端应用程序接收压缩的像素流，对其进行解密，然后将其输出到本地显示。

NICE DCV 功能

NICE DCV 提供以下功能：

- 共享整个桌面 - 使用高性能 NICE DCV 协议共享整个远程桌面的完全控制权。
- 仅传输图像 - 将渲染的图像作为像素传输，而不是作为几何体和场景信息传输。这就提供了另外一层安全性，因为此操作不会通过网络发送专有客户信息。
- 支持基于 H.264 的编码 - 使用基于 H.264 的视频压缩和编码以降低带宽消耗。
- 支持无损质量的视频压缩 - 支持无损质量视频压缩（在网络和处理器条件允许的情况下）。
- 匹配显示布局 - 自动适应服务器的屏幕分辨率和显示布局，以与客户端窗口大小匹配。
- 支持多屏幕 - 您可以将会话桌面扩展到最多 4 个显示器。Windows 和 macOS 的本机客户端支持高像素密度显示器。
- 适应压缩级别 - 自动根据网络的可用带宽和延迟适应视频压缩级别。
- 启用协作 - 提供支持多个协作客户端的动态会话。客户端可以在会话期间随时连接和断开连接。

- 支持每个服务器具有多个会话 (仅限 Linux NICE DCV 服务器) - 支持每个 Linux NICE DCV 服务器具有多个虚拟会话以最大限度节省成本。
- 支持 GPU 共享 (仅限 Linux NICE DCV 服务器) - 您可以在 Linux NICE DCV 服务器上运行的多个虚拟会话之间共享一个或多个物理 GPU。
- 支持触摸输入、触控笔输入和游戏手柄 - 您可以使用连接到本地计算机的输入设备与远程 NICE DCV 会话进行交互。
- 支持 WebAuthn、智能卡、触控笔和 USB 远程控制 - 您可以在 NICE DCV 会话中使用外围设备，就像在本地计算机上一样。
- 支持音频输入和输出、打印以及复制和粘贴 - 您可以在会话和本地计算机之间执行这些关键操作。
- 支持文件传输 - 您可以在会话和本地计算机之间传输文件。
- 提供 HTML5 客户端 – 提供一个 HTML5 客户端，它可在 Windows 和 Linux 上与任何现代 Web 浏览器结合使用。
- 支持现代 Linux 桌面环境 - 支持现代 Linux 桌面，例如 RHEL 8 上的 Gnome 3。

NICE DCV 定价

在 Amazon EC2 实例上使用 NICE DCV 服务器不收取任何额外的费用。您需要按标准费率为使用的实例和其他 Amazon EC2 功能支付费用。

否则，需要使用许可证。有关更多信息，请参阅[为 NICE DCV 服务器授予许可](#)。

NICE DCV 服务器

NICE DCV 服务器适用于 Windows 和 Linux。这两个服务器提供了类似的功能，但有一些差异。选择最满足您的需求的 NICE DCV 服务器。下表比较了 Windows 和 Linux NICE DCV 服务器支持的功能。

主题

- [要求](#)
- [支持的功能](#)

要求

为了获得良好的 NICE DCV 用户体验，请确保您的服务器满足以下最低要求。请记住，您的用户的体验在很大程度上取决于从 NICE DCV 服务器流式传输到 NICE DCV 客户端的像素数。

如果您要在 Amazon EC2 实例上安装 NICE DCV 服务器，我们建议您使用 Amazon EC2 G3、G4dn、G4ad 或 G5 实例类型。这些实例类型提供 GPU，这些 GPU 支持基于硬件的 OpenGL 和 GPU 共享。有关更多信息，请参阅 [Amazon EC2 G3 实例](#)、[Amazon EC2 G4 实例](#) 和 [Amazon EC2 G5 实例](#)。


您可以在任何其他实例类型上安装 NICE DCV 服务器，但可能存在屏幕分辨率限制。要在 Windows Server 2016 上绕过该限制，请下载并安装[适用于 EC2 的 NICE DCV 虚拟显示驱动程序](#)。在运行 DCV 2023.1 或更高版本的 Windows Server 2019 或更高版本上，无需采取任何其他措施。

您的服务器必须满足下表中列出的最低要求。

	Windows 服务器	Linux 服务器
操作系统	<ul style="list-style-type: none"> • Windows 10 • Windows 11 • Windows Server 2016 • Windows Server 2019 • Windows Server 2022 	<ul style="list-style-type: none"> • Amazon Linux 2 • CentOS 7.6 或更高版本 • CentOS Stream 8 • CentOS Stream 9 • RHEL 7.6 或更高版本 • RHEL 8.x • RHEL 9.x • SUSE Linux Enterprise 12 SP5 或更高版本

	Windows 服务器	Linux 服务器
	<p>Note</p> <p>所有支持的 Windows 操作系统都需要 .NET Framework 4.5 , 并且必须支持 x86-64 架构。</p>	<ul style="list-style-type: none"> • SUSE Linux Enterprise 15 SP5 • Rocky Linux 8.5 或更高版本 • Rocky Linux 9 • Ubuntu 20.04 • Ubuntu 22.04
支持的架构	64 位 x86	<ul style="list-style-type: none"> • 64 位 x86 • 64 位 ARM (仅支持运行亚马逊 Linux 2、RHEL 7.x/8.x/9.x、CentOS 7/8/9、Rocky Linux 8/9 和 Ubuntu 22.04 的 Amazon EC2 实例)
GPU	<p>(可选) 基于硬件的视频编码需要使用 NVIDIA 或 AMD GPU。如果您的服务器没有 GPU , 则会使用基于软件的视频编码。</p> <p>Note</p> <ul style="list-style-type: none"> • NVIDIA GPU 需要使用 NVENC 以进行基于硬件的视频编码。需要计算能力 ≥ 3.5 的 NVIDIA GPU。 • AMD GPU 需要使用适用于 Linux 或 Windows 的高级媒体框架 (AMF) 或仅适用于 Windows 的 Rapidfire , 以进行基于硬件的视频编码。对于 Linux , 可以在 Ubuntu 实例上安装 AMD 驱动程序提供的附加软件包 <code>amf-amdgpu-pro</code> 以使用 AMF 编码器。 <p>需要使用 NVIDIA GPU 以在虚拟会话之间共享 GPU。</p> <p>Note</p> <p>仅在具有 AMD GPU 的 Linux 服务器上支持控制台会话。</p>	

	Windows 服务器	Linux 服务器
Network	默认情况下，NICE DCV 服务器通过端口 8443 进行通信。该端口是可配置的，但必须大于 1024。确保服务器允许通过所需的端口进行通信。	

 Note

NICE DCV 不支持生命周期结束的操作系统。请与您的供应商联系以了解您的操作系统。

有关 NICE DCV 客户端要求的更多信息，请参阅《NICE DCV 用户指南》中的 [NICE DCV 客户端要求](#)。

支持的功能

下表比较了 Windows 和 Linux NICE DCV 服务器支持的功能。

功能	Windows NICE DCV 服务器	Linux NICE DCV 服务器
控制台会话	✓	✓
虚拟会话	✗	✓
QUIC (UDP) 传输协议	✓	✓
可配置的 TCP/UDP 端口和地址	✓	✓
自定义 TLS 证书	✓	✓
空闲客户端断开连接	✓	✓
GPU 共享	✗	✓
USB 远程控制	✓	✓
智能卡支持	✓	✓

功能	Windows NICE DCV 服务器	Linux NICE DCV 服务器
网络摄像头支持	✓ (Windows 10 和 Server 2016 和更高版本)	✗
会话存储和文件传输	✓	✓
复制和粘贴	✓	✓
自定义 HTTP 标头	✓	✓
从会话打印	✓	✓
立体声 2.0 音频播放	✓	✓
环绕声音频播放	✓ (最高 7.1)	✓ (最高 5.1)
立体声 2.0 音频录制	✓	✓
触摸屏支持	✓ (Windows 10 和 Server 2012 和更高版本)	✓
触控笔支持	✓ (Windows 10 和 Server 2019)	✓
游戏手柄支持	✓ (Windows 10 和 Server 2016 和更高版本)	✗
全屏选定显示器	✓	✗
时区重定向	✓	✓
WebAuthn 重定向	✓	✗

有关 NICE DCV 客户端功能的更多信息，请参阅《NICE DCV 用户指南》中的 [NICE DCV 客户端功能](#)。

设置 NICE DCV 服务器

要使用 NICE DCV，请在打算托管 NICE DCV 会话的服务器上安装 NICE DCV 服务器软件。确保为该软件正确授予许可。

以下主题介绍了如何安装 NICE DCV 服务器以及为其授予许可。[许可](#)主题仅适用于在本地和其他基于云的服务器上安装 NICE DCV 服务器。这是因为在 Amazon EC2 实例上使用 NICE DCV 服务器不需要任何许可证。

主题

- [安装 NICE DCV 服务器](#)
- [为 NICE DCV 服务器授予许可](#)
- [升级 NICE DCV 服务器](#)
- [卸载 NICE DCV 服务器](#)
- [成像 NICE DCV 服务器](#)

安装 NICE DCV 服务器

以下主题介绍了如何在 Windows 和 Linux 上安装最新版本的 NICE DCV 服务器。如果要在 Amazon EC2 实例或其他本地或基于云的服务器上安装 NICE DCV，请按照以下步骤进行操作。

Note

如果要从早期版本的 NICE DCV 服务器升级到最新版本，请参阅[升级 NICE DCV 服务器](#)。

主题

- [在 Windows 上安装 NICE DCV 服务器](#)
- [在 Linux 上安装 NICE DCV 服务器](#)

在 Windows 上安装 NICE DCV 服务器

本节介绍了如何在 Windows 上安装 NICE DCV 服务器。

主题

- [Amazon EC2 实例上的 Windows NICE DCV 服务器的先决条件](#)
- [在 Windows 上安装 NICE DCV 服务器](#)

Amazon EC2 实例上的 Windows NICE DCV 服务器的先决条件

本主题介绍了如何在安装 NICE DCV 服务器之前配置 Windows Amazon EC2 实例。如果您不在 Amazon EC2 Windows 实例上安装 NICE DCV 服务器，请跳过这些先决条件。

主题

- [加速型计算实例的先决条件](#)
- [其他实例系列的先决条件](#)

加速型计算实例的先决条件

GPU 图形实例的先决条件

如果您使用 GPU 图形实例（例如 G2、G3、G4dn、G4ad 或 G5 实例），我们建议您安装并配置相应的 NVIDIA 或 AMD GPU 驱动程序。GPU 驱动程序支持以下功能：

- 适用于应用程序的 DirectX 和 OpenGL 硬件加速
- 适用于 H.264 视频流编码的硬件加速
- 可自定义的服务器监视器分辨率
- 提高了服务器显示器的最大分辨率 - 高达 4096x2160
- 增加了服务器监视器的数量

有关如何在 GPU 图形实例上安装 NVIDIA GPU 驱动程序的说明，请参阅《Amazon EC2 用户指南》中的以下主题。

- 对于具有 NVIDIA GPU 的实例（例如 G2、G3、G4dn 或 G5 实例），请参阅[在 Windows 上安装 NVIDIA 驱动程序](#)。
- 对于具有 AMD GPU 的实例（例如 G4ad 实例），请参阅[在 Windows 实例上安装 AMD 驱动程序](#)。

有关 Amazon EC2 G4ad 实例的更多信息，请参阅 [Deep dive on the new Amazon EC2 G4ad instances](#) 博客文章。

其他加速型计算实例的先决条件

如果您使用的加速型计算实例不是 GPU 图形实例（例如 P2、P3 或 P3dn 实例），我们建议您安装并配置相应的 NVIDIA GPU 驱动程序。NVIDIA GPU 驱动程序为 H.264 视频流编码启用硬件加速。

有关如何在加速计算实例上安装 NVIDIA GPU 驱动程序的说明，请参阅 Amazon EC2 用户指南中的[公共 NVIDIA 驱动程序](#)。

在加速型计算实例上安装 NVIDIA GPU 驱动程序不会提高服务器显示器限制或分辨率。要添加额外的服务器显示器分辨率支持，您可以安装 NVIDIA GRID 驱动程序。有关更多信息，请参阅 NVIDIA 网站上的[NVIDIA vGPU 软件](#)。

其他实例系列的先决条件

对于加速型计算实例以外的实例，如果您使用 Windows 2016 或运行 2023.1 之前的 NICE DCV 服务器版本，我们建议您安装 NICE DCV 虚拟显示驱动程序。这包括通用型、计算优化型、内存优化型和存储优化型实例系列中的实例。

安装 NICE DCV 虚拟显示驱动程序可以实现以下功能：

- 最多支持 4 个显示器
- 支持自定义分辨率
- 支持 4K UHD 分辨率

您无法使用 Windows 控制面板管理 NICE DCV 服务器连接的服务器显示器。

Note

在 Windows Server 2016 和更高版本上支持 NICE DCV 虚拟显示驱动程序。如果您使用 Windows Server 2019 或更高版本以及 DCV Server 2023.1 或更高版本，则不需要使用该驱动程序。

Important

将 NICE DCV 虚拟显示驱动程序与任何其他 GPU 驱动程序（例如 NVIDIA GPU 驱动程序）一起安装可能会发生冲突。为了避免冲突，我们建议您不要将 NICE DCV 虚拟显示驱动程序与任何其他 GPU 驱动程序一起安装。

在您的实例上安装 NICE DCV 虚拟显示驱动程序

1. 从 [NICE DCV 网站](#) 中下载 NICE DCV 虚拟显示驱动程序安装程序。
2. 要运行向导以安装该驱动程序，请打开或双击安装文件。或者，使用以下命令运行自动安装。

```
C:\> nice-dcv-virtual-display-x64-Release-78.msi /quiet /norestart
```

3. 重启实例，然后重新连接到它。

在 Windows 上安装 NICE DCV 服务器

您可以使用安装向导在 Windows 主机服务器上安装 NICE DCV 服务器。该向导指导您完成一系列步骤，这些步骤说明了如何自定义 NICE DCV 服务器安装。或者，您可以使用命令行执行无人值守安装。这会使用默认设置自动完成安装过程。

目录

- [使用向导](#)
- [无人值守安装](#)

使用向导

可以使用 NICE DCV 服务器安装向导进行指导式安装。

使用向导在 Windows 上安装 NICE DCV 服务器

1. 启动并连接到要安装 NICE DCV 服务器的服务器。
2. 从 [NICE DCV 网站](#) 中下载 NICE DCV 服务器安装程序。

Note

NICE DCV 服务器仅提供 64 位版本，并在 64 位 Windows 操作系统上支持 NICE DCV 服务器。

i Tip

下载网站的[最新软件包](#)页面包含始终指向最新可用版本的链接。您可以使用这些链接自动检索最新的 NICE DCV 软件包。

3. 运行 `nice-dcv-server-x64-Release-2023.1-version_number.msi`。
4. 在欢迎屏幕上，选择 Next。
5. 在 End-User License Agreement 屏幕上，阅读许可协议。如果您接受这些条款，请选中 I accept the terms in the License Agreement 复选框，然后选择 Next。
6. （可选）在 Components Selection 屏幕中选择项目以配置要安装的组件。要将组件标记为安装，请选择该项目，然后选择 Will be installed on local hard drive。要从安装中忽略某个组件，请选择该项目，然后选择 Entire feature will be unavailable。
7. 在“DCV Service Configuration (DCV 服务配置)”屏幕上：
 - a. （可选）要手动配置您服务器的防火墙以允许通过所需端口进行通信，请选择 No, I will manually configure my firewall later (不，我将稍后手动配置我的防火墙)。
 - b. （可选）要在安装后手动启动 NICE DCV 服务器，请选择 No, I want to start a DCV Service manually。如果您选择此选项，则无法在安装完成后自动启动控制台会话。如果选择此选项，则跳过步骤 9。
8. 选择下一步。
9. 在“DCV Session Management Configuration (DCV 会话管理配置)”屏幕上，指定自动控制台会话的所有者。或者，要防止安装完成后启动自动控制台会话，请选择不，我将手动创建会话。

i Note

仅当您以前选择了允许服务器自动启动的情况下，才完成此步骤。

10. 选择安装。

无人值守安装


默认情况下，无人值守安装执行以下操作：

- 添加防火墙规则以允许通过端口 8443 进行通信。
- 启用 NICE DCV 服务器自动启动。

- 创建自动控制台会话。
- 将控制台会话所有者设置为执行安装的用户。

您可以通过将以下选项附加到安装命令来覆盖默认操作：

- `DISABLE_FIREWALL=1` - 禁止安装程序添加防火墙规则。
- `DISABLE_SERVER_AUTOSTART=1` - 禁止 NICE DCV 服务器在安装后自动启动。
- `DISABLE_AUTOMATIC_SESSION_CREATION=1` - 禁止安装程序启动自动控制台会话。
- `AUTOMATIC_SESSION_OWNER=owner_name` - 为自动控制台会话指定不同的所有者。
- `ADDLOCAL=component_list` - 将组件添加到要安装的组件集中。
- `REMOVE=component_list` - 从要安装的组件集中删除组件。

 Note

REMOVE 选项在 ADDLOCAL 选项之后进行评估。不会安装位于两个列表中的组件。

`component_list` 是一个以逗号分隔的列表，可以包含以下值：

- `audioMicDriver`：麦克风驱动程序
- `audioSpkDriver`：扬声器驱动程序
- `printerDriver`：打印机驱动程序
- `usbDriver`：USB 设备远程控制驱动程序（默认禁用）
- `webcamDriver`：网络摄像头驱动程序
- `gamepadDriver`：游戏手柄驱动程序
- `webClient`：Web 客户端
- `webauthn`：WebAuthn 重定向
- `ALL`：所有组件

使用无人值守安装在 Windows 上安装 NICE DCV 服务器

1. 启动并连接到打算安装 NICE DCV 服务器的服务器。
2. 从 [NICE 网站](#) 中下载 NICE DCV 服务器安装程序。

Note

NICE DCV 服务器仅提供 64 位版本，并在 64 位 Windows 操作系统上支持 NICE DCV 服务器。

3. 打开命令提示符窗口，导航到下载安装程序的文件夹。
4. 运行无人值守安装程序，如以下示例之一所示：

- 安装默认组件：

```
C:\> msixexec.exe /i nice-dcv-server-x64-Release-2023.1-version_number.msi /quiet /norestart /l*v dcv_install_msi.log
```

- 安装所有组件：

```
C:\> msixexec.exe /i nice-dcv-server-x64-Release-2023.1-version_number.msi ADDLOCAL=ALL /quiet /norestart /l*v dcv_install_msi.log
```

- 安装一部分组件：

```
C:\> msixexec.exe /i nice-dcv-server-x64-Release-2023.1-version_number.msi ADDLOCAL=audioMicDriver,audioSpkDriver,printerDriver,webcamDriver /quiet /norestart /l*v dcv_install_msi.log
```

在 Linux 上安装 NICE DCV 服务器

本节介绍了如何在 Linux 上安装 NICE DCV 服务器。

主题

- [Linux NICE DCV 服务器的先决条件](#)
- [在 Linux 上安装 NICE DCV 服务器](#)
- [安装后检查](#)

Linux NICE DCV 服务器的先决条件

NICE DCV 允许客户端访问 Linux 服务器上的远程图形 X 会话。这会提供对相应 Linux 桌面的访问。NICE DCV 支持两种类型的 Linux 桌面流：控制台会话和虚拟会话。有关控制台会话和虚拟会话的更多信息，请参阅[管理 NICE DCV 会话](#)。

本主题介绍了如何在 Linux 服务器上安装 NICE DCV 所需的必备组件。

内容

- [安装桌面环境和桌面管理器](#)
- [禁用 Wayland 协议 \(仅限 GDM3 \)](#)
- [配置 X 服务器](#)
- [安装 glxinfo 实用程序](#)
- [验证 OpenGL 软件渲染](#)
- [为图形实例安装 GPU 驱动程序](#)
- [为非 GPU 实例安装 XDummy 驱动程序](#)

安装桌面环境和桌面管理器

安装桌面环境和桌面管理器以改善您在 Linux 服务器上安装 NICE DCV 的体验。

桌面环境是一个图形用户界面 (GUI)，有助于您与 Linux 操作系统交互。具有多种桌面环境，NICE DCV 可以与其中的很多环境一起使用。桌面管理器是管理用户登录屏幕、启动和停止桌面环境会话和 X 服务器的项目。

以下选项卡式内容显示了在支持的操作系统上安装默认桌面环境和桌面管理器的步骤，还说明了如何在支持的操作系统上配置和启动 X 服务器。

RHEL, CentOS, and Rocky Linux

RHEL、CentOS 和 Rocky Linux 的默认桌面环境是 Gnome3，默认桌面管理器是 GDM。

在 RHEL、CentOS 和 Rocky Linux 上安装和配置桌面环境和桌面管理器

1. 安装桌面环境和桌面管理器程序包

- RHEL 和洛基 Linux

```
$ sudo yum groupinstall 'Server with GUI'
```

- CentOS

```
$ sudo yum groupinstall "GNOME Desktop"
```

2. 更新软件包来确保 Linux 服务器是最新的。

```
$ sudo yum upgrade
```

3. 重新启动 Linux 服务器。

```
$ sudo reboot
```

Amazon Linux 2

Note

目前，NICE DCV 与亚马逊 Linux 2023 不兼容。AL2023 不包括运行 NICE DCV 所需的图形桌面环境。

Amazon Linux 2 的默认桌面环境是 Gnome3，默认桌面管理器是 GDM。

在 Amazon Linux 2 上安装和配置桌面环境和桌面管理器

1. 安装桌面环境和桌面管理器程序包

```
$ sudo yum install gdm gnome-session gnome-classic-session gnome-session-xsession
```

```
$ sudo yum install xorg-x11-server-Xorg xorg-x11-fonts-Type1 xorg-x11-drivers
```

```
$ sudo yum install gnome-terminal gnu-free-fonts-common gnu-free-mono-fonts gnu-free-sans-fonts gnu-free-serif-fonts
```

2. 更新软件包来确保 Linux 服务器是最新的。

```
$ sudo yum upgrade
```

3. 重新启动 Linux 服务器。


```
$ sudo reboot
```

Ubuntu 20.x and 22.x

对于 Ubuntu 20.x/22.x，默认桌面环境是 Gnome3，默认桌面管理器是 GDM3。从 Ubuntu 20.x 开始，在 NICE DCV 中不再支持 LightDM。

在 Ubuntu 20.x/22.x 上安装和配置桌面环境和桌面管理器

1. 安装桌面环境和桌面管理器程序包

```
$ sudo apt update
```

```
$ sudo apt install ubuntu-desktop
```

安装 GDM3

```
$ sudo apt install gdm3
```

2. 验证 GDM3 是否设置为默认桌面管理器。

```
$ cat /etc/X11/default-display-manager
```

输出如下所示。

```
/usr/sbin/gdm3
```

如果 GDM3 未设置为默认桌面管理器，请使用以下命令将其设置为默认桌面管理器。

```
$ sudo dpkg-reconfigure gdm3
```

3. 更新软件包来确保 Linux 服务器是最新的。

```
$ sudo apt upgrade
```

4. 重新启动 Linux 服务器。

```
$ sudo reboot
```

Note

在虚拟会话中使用早于 2022.2 的 NICE DCV 版本时，您可能会遇到[一个已知的 GDM 问题](#)。要使虚拟会话正常工作，您可以采用以下解决方案之一：

- 在没有 GPU 的服务器上，您可以禁用桌面管理器，因为不需要使用桌面管理器即可运行虚拟会话。在创建虚拟会话之前运行以下命令，以将系统配置为在多用户模式下运行：

```
sudo systemctl isolate multi-user.target
```

- 在具有 GPU 的服务器上，除了禁用桌面管理器以外，您还需要在创建虚拟会话之前在系统上启动 X 服务器。为此，请运行以下命令：

```
sudo systemctl isolate multi-user.target
```

```
sudo dcvstartx &
```

NICE DCV 2022.2 和更高版本不受该问题影响。

SUSE Linux Enterprise 12.x

SUSE Linux Enterprise 12.x 的默认桌面环境是 SLE Classic，默认桌面管理器是 GDM。

在 SUSE Linux Enterprise 12.x 上安装和配置桌面环境和桌面管理器

1. 安装桌面环境和桌面管理器程序包

```
$ sudo zypper install -t pattern gnome-basic
```

2. 验证 GDM 是否设置为默认桌面管理器。

```
$ sudo update-alternatives --set default-displaymanager /usr/lib/X11/  
displaymanagers/gdm
```

```
$ sudo sed -i "s/DEFAULT_WM=\"\"/DEFAULT_WM=\"gnome\"/" /etc/sysconfig/windowmanager
```

3. 更新软件包来确保 Linux 服务器是最新的。

```
$ sudo zypper update
```

4. 重新启动 Linux 服务器。

```
$ sudo reboot
```

SUSE Linux Enterprise 15.x

SUSE Linux Enterprise 15.x 的默认桌面环境是 SLE Classic，默认桌面管理器是 GDM3。

在 SUSE Linux Enterprise 15.x 上安装和配置桌面环境和桌面管理器

1. 安装桌面环境和桌面管理器程序包

```
$ sudo zypper install -t pattern gnome_basic
```

2. 验证 GDM 是否设置为默认桌面管理器。

```
$ sudo update-alternatives --set default-displaymanager /usr/lib/X11/displaymanagers/gdm
```

```
$ sudo sed -i "s/DEFAULT_WM=\"\"/DEFAULT_WM=\"gnome\"/" /etc/sysconfig/windowmanager
```

3. 更新软件包来确保 Linux 服务器是最新的。

```
$ sudo zypper update
```

4. 重新启动 Linux 服务器。

```
$ sudo reboot
```

Note

在虚拟会话中使用早于 2022.2 的 NICE DCV 版本时，您可能会遇到[一个已知的 GDM 问题](#)。要使虚拟会话正常工作，您可以采用以下解决方案之一：

- 在没有 GPU 的服务器上，您可以禁用桌面管理器，因为不需要使用桌面管理器即可运行虚拟会话。在创建虚拟会话之前运行以下命令，以将系统配置为在多用户模式下运行：

```
sudo systemctl isolate multi-user.target
```

- 在具有 GPU 的服务器上，除了禁用桌面管理器以外，您还需要在创建虚拟会话之前在系统上启动 X 服务器。为此，请运行以下命令：

```
sudo systemctl isolate multi-user.target
```

```
sudo dcvstartx &
```

NICE DCV 2022.2 和更高版本不受该问题影响。

禁用 Wayland 协议 (仅限 GDM3)

NICE DCV 不支持 Wayland 协议。如果使用 GDM3 桌面管理器，您必须禁用 Wayland 协议。如果不使用 GDM3，请跳过该步骤。

禁用 Wayland 协议

1. 使用常用的文本编辑器打开以下文件。

- RHEL、CentOS 和 SUSE Linux Enterprise 15.x

```
/etc/gdm/custom.conf
```

- Ubuntu 18.x/20.x/22.x

```
/etc/gdm3/custom.conf
```

2. 在 [daemon] 部分中，将 WaylandEnable 设置为 false。

```
[daemon]  
WaylandEnable=false
```

3. 重新启动 GDM 服务。

- RHEL 和 CentOS

```
$ sudo systemctl restart gdm
```

- Ubuntu 18.x/20.x/22.x

```
$ sudo systemctl restart gdm3
```

- SUSE Linux Enterprise 15.x

```
$ sudo systemctl restart xdm
```

配置 X 服务器

如果要使用控制台会话或 GPU 共享，则必须确保 Linux 服务器具有一个配置正确且正在运行的 X 服务器。

Note

如果您打算使用虚拟会话而不共享 GPU，则不需要使用 X 服务器。

X 服务器程序包通常作为桌面环境和桌面管理器的依赖项安装。我们建议您将 X 服务器配置为当 Linux 服务器启动时自动启动。

在 RHEL、CentOS、Rocky Linux、Amazon Linux 2、Ubuntu 18.x、20.x 和 22.x 以及 SUSE Linux Enterprise 12.x 和 15.x 上配置和启动 X 服务器：

1. 将 X 服务器配置为当 Linux 服务器启动时自动启动。

```
$ sudo systemctl get-default
```

如果命令返回 `graphical.target`，则表明 X 服务器已配置为自动启动。继续下一步。

如果该命令返回 `multi-user.target`，则 X 服务器未配置为自动启动。执行下面的命令：

```
$ sudo systemctl set-default graphical.target
```

2. 启动 X 服务器。

```
$ sudo systemctl isolate graphical.target
```

3. 验证 X 服务器是否正在运行。

```
$ ps aux | grep X | grep -v grep
```

下面显示 X 服务器在运行时的示例输出。

```
root 1891 0.0 0.7 277528 30448 tty7 Ssl+ 10:59 0:00 /usr/bin/Xorg :0 -  
background none -verbose -auth /run/gdm/auth-for-gdm-wltseN/database -  
seat seat0 vt7
```

安装 glxinfo 实用程序

glxinfo 实用程序提供关于您的 Linux 服务器的 OpenGL 配置的信息。该实用程序可用于确定您的 Linux 服务器是否配置为支持 OpenGL 硬件或软件渲染。它提供有关驱动程序和支持的扩展的信息。

glxinfo 实用程序作为 DCV GL 的程序包依赖项安装。因此，如果您安装了 DCV GL，则 glxinfo 实用程序已随之安装在您的 Linux 服务器上。

RHEL, CentOS, Rocky Linux, and Amazon Linux 2

安装 glxinfo 实用程序

运行以下命令：

```
$ sudo yum install glx-utils
```

Ubuntu

安装 glxinfo 实用程序

运行以下命令：

```
$ sudo apt install mesa-utils
```

SUSE Linux Enterprise

安装 glxinfo 实用程序

运行以下命令：

```
$ sudo zypper in Mesa-demo-x
```

验证 OpenGL 软件渲染

在非 GPU 的 Linux 服务器上，仅在软件渲染模式下使用 Mesa 驱动程序支持 OpenGL。如果您使用非 GPU Linux 服务器并打算使用 OpenGL，请确保在 Linux 服务器上安装并正确配置 Mesa 驱动程序。

Note

这仅适用于非 GPU 的 Linux 服务器。

验证 OpenGL 软件渲染是否可用

确保 X 服务器正在运行，并使用以下命令。

```
$ sudo DISPLAY=:0 XAUTHORITY=$(ps aux | grep "X.*\-auth" | grep -v grep | sed -n 's/.*-auth \([^ ]+\).*\1/p') glxinfo | grep -i "opengl.*version"
```

下面显示了 OpenGL 软件渲染可用时的示例输出：

```
OpenGL core profile version string: 3.3 (Core Profile) Mesa 17.0.5
OpenGL core profile shading language version string: 3.30
OpenGL version string: 3.0 Mesa 17.0.5
OpenGL shading language version string: 1.30
OpenGL ES profile version string: OpenGL ES 3.0 Mesa 17.0.5
OpenGL ES profile shading language version string: OpenGL ES GLSL ES 3.00
```

为图形实例安装 GPU 驱动程序

主题

- [安装和配置 NVIDIA 驱动程序](#)
- [安装和配置 AMD 驱动程序](#)

安装和配置 NVIDIA 驱动程序

对于具有专用 NVIDIA GPU 的 Linux 服务器，请确保安装并正确配置相应的 NVIDIA 驱动程序。有关如何在亚马逊 EC2 Linux 实例上安装 NVIDIA 驱动程序的说明，请参阅亚马逊 EC2 用户指南中的[在 Linux 服务器上安装 NVIDIA 驱动程序](#)。

Note

- 这仅适用于具有 NVIDIA GPU 的 Linux 服务器。
- 对于每个安装的 GPU，GRID 驱动程序支持最多 4 个 4K 显示器。对于每个安装的 GPU，游戏驱动程序仅支持一个 4K 显示器。

在 Linux 服务器上安装 NVIDIA 驱动程序后，更新 `xorg.conf`。

生成更新的 `xorg.conf`

1. 运行以下命令。

```
sudo nvidia-xconfig --preserve-busid --enable-all-gpus
```

如果您使用 G3、G4 或 G5 Amazon EC2 实例，并希望使用多显示器控制台会话，请包含 `--connected-monitor=DFP-0,DFP-1,DFP-2,DFP-3` 参数。该参数如下所示。

```
sudo nvidia-xconfig --preserve-busid --enable-all-gpus --connected-monitor=DFP-0,DFP-1,DFP-2,DFP-3
```

Note

确保您的服务器没有旧 `/etc/X11/XF86Config` 文件。如果服务器具有该文件，则 `nvidia-xconfig` 将更新该配置文件，而不是生成所需的 `/etc/X11/xorg.conf` 文件。运行以下命令可删除旧 `XF86Config` 文件：

```
sudo rm -rf /etc/X11/XF86Config*
```

2. 重新启动 X 服务器以使更改生效。

- ```
$ sudo systemctl isolate multi-user.target
```

```
$ sudo systemctl isolate graphical.target
```

验证您的 NVIDIA GPU 是否支持基于硬件的视频编码

确保它支持 NVENC 编码，并且其计算容量大于或等于 3.0；对于 Ubuntu 20，确保其计算容量大于或等于 3.5。

要验证 NVENC 支持，请参阅 [NVIDIA Video Encode and Decode GPU Support Matrix](#)。要检查计算容量，请参阅 [NVIDIA Compute Capacity tables](#)。

如果您的 NVIDIA GPU 不支持 NVENC 编码或没有所需的计算容量，则会使用基于软件的视频编码。

验证 OpenGL 硬件渲染是否可用

使用以下命令可确保 X 服务器正在运行。

```
$ sudo DISPLAY=:0 XAUTHORITY=$(ps aux | grep "X.*\-auth" | grep -v grep | sed -n 's/.*-auth \([^]*\).*\1/p') glxinfo | grep -i "opengl.*version"
```

下面显示了 OpenGL 硬件渲染可用时的示例输出。

```
OpenGL core profile version string: 4.4.0 NVIDIA 390.75
OpenGL core profile shading language version string: 4.40 NVIDIA via Cg compiler
OpenGL version string: 4.6.0 NVIDIA 390.75
OpenGL shading language version string: 4.60 NVIDIA
OpenGL ES profile version string: OpenGL ES 3.2 NVIDIA 390.75
OpenGL ES profile shading language version string: OpenGL ES GLSL ES 3.20
```

## 安装和配置 AMD 驱动程序

已挂载 AMD GPU 的实例（如 G4ad 实例）必须安装相应的 AMD 驱动程序。有关如何在兼容的 Amazon EC2 实例上安装 AMD GPU 驱动程序的说明，请参阅[在 Linux 实例上安装 AMD 驱动程序](#)。

有关 Amazon EC2 G4ad 实例的更多信息，请参阅 [Deep dive on the new Amazon EC2 G4ad instances](#) 博客文章。

## 为非 GPU 实例安装 XDummy 驱动程序

### 主题

- [安装和配置 XDummy 驱动程序](#)

### 安装和配置 XDummy 驱动程序

要在没有专用 GPU 的 Linux 服务器上使用控制台会话，请确保安装并正确配置了 Xdummy 驱动程序。在不存在真正的 GPU 时，XDummy 驱动程序允许 X 服务器使用虚拟帧缓冲区运行。

#### Note

- 如果您打算使用虚拟会话，则不需要这样做。
- XDummy 驱动程序只能支持其配置中定义的分辨率。

## RHEL, CentOS, Rocky Linux, and Amazon Linux 2

### 安装 XDummy 驱动程序

运行以下命令：

```
$ sudo yum install xorg-x11-drv-dummy
```

## Ubuntu

### 安装 XDummy 驱动程序

运行以下命令：

```
$ sudo apt install xserver-xorg-video-dummy
```

## SUSE Linux Enterprise

### 安装 XDummy 驱动程序

运行以下命令：

```
$ sudo zypper in xf86-video-dummy
```

在 Linux 服务器上安装 XDummy 驱动程序后，更新 `xorg.conf`。

在 `xorg.conf` 中配置 XDummy

1. 使用常用的文本编辑器打开 `/etc/X11/xorg.conf` 文件。
2. 将以下部分添加到配置中。

```
Section "Device"
 Identifier "DummyDevice"
 Driver "dummy"
 Option "UseEDID" "false"
 VideoRam 512000
EndSection

Section "Monitor"
 Identifier "DummyMonitor"
 HorizSync 5.0 - 1000.0
 VertRefresh 5.0 - 200.0
 Option "ReducedBlanking"
EndSection

Section "Screen"
 Identifier "DummyScreen"
 Device "DummyDevice"
 Monitor "DummyMonitor"
 DefaultDepth 24
 SubSection "Display"
 Viewport 0 0
 Depth 24
 Virtual 4096 2160
 EndSubSection
EndSection
```

#### Note

提供的配置是一个示例。您可以添加更多模式并设置不同的 `virtual` 分辨率。您还可以配置多个虚拟显示器。

3. 重新启动 X 服务器以使更改生效。

- ```
$ sudo systemctl isolate multi-user.target
```

```
$ sudo systemctl isolate graphical.target
```

在 Linux 上安装 NICE DCV 服务器

NICE DCV 服务器是使用一系列 RPM 或 .deb 软件包安装的，具体取决于主机服务器的操作系统。这些程序包将安装所有必需的程序包及其依赖项，并执行必需的服务器配置。

Note

您必须以根用户身份登录才能安装 NICE DCV 服务器。

安装 NICE DCV 服务器

Amazon Linux 2 and RHEL/CentOS

NICE DCV 服务器适用于基于 64 位 x86 和 64 位 ARM 架构的 Amazon Linux 2、RHEL 和 CentOS 服务器。

Important

nice-dcv-gl 和 nice-dcv-gltest 软件包不适用于基于 64 位 ARM 架构的服务器。

在 Amazon Linux 2、RHEL 和 CentOS 上安装 NICE DCV 服务器

1. 启动并连接到您打算安装 NICE DCV 服务器的服务器。
2. NICE DCV 服务器软件包使用安全 GPG 签名进行数字签名。要允许软件包管理器验证软件包签名，您必须导入 NICE GPG 密钥。为此，打开一个终端窗口并导入 NICE GPG 密钥。

```
$ sudo rpm --import https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY
```

3. 从 [NICE DCV 下载网站](#) 中下载软件包。RPM 和 deb 程序包打包到一个 .tgz 存档中。确保为您的操作系统下载正确的归档。
 - 64 位 x86

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2023.1/Servers/nice-dcv-2023.1-16388-el7-x86_64.tgz
```

- 64 位 ARM

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2023.1/Servers/nice-dcv-2023.1-16388-el7-aarch64.tgz
```

 Tip

下载网站的[最新软件包](#)页面包含始终指向最新可用版本的链接。您可以使用这些链接自动检索最新的 NICE DCV 软件包。

- 64 位 x86

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-el7-x86_64.tgz
```

- 64 位 ARM

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-el7-aarch64.tgz
```

4. 提取 .tgz 归档的内容，并导航到提取的目录。

- 64 位 x86

```
$ tar -xvzf nice-dcv-2023.1-16388-el7-x86_64.tgz && cd nice-dcv-2023.1-16388-el7-x86_64
```

- 64 位 ARM

```
$ tar -xvzf nice-dcv-2023.1-16388-el7-aarch64.tgz && cd nice-dcv-2023.1-16388-el7-aarch64
```

5. 安装 NICE DCV 服务器。

- 64 位 x86

```
$ sudo yum install nice-dcv-server-2023.1.16388-1.el7.x86_64.rpm
```

- 64 位 ARM

```
$ sudo yum install nice-dcv-server-2023.1.16388-1.el7.aarch64.rpm
```

6. (可选) 要将 Web 客户端与 NICE DCV 2021.2 和更高版本一起使用, 请安装 `nice-dcv-web-viewer` 软件包。

- 64 位 x86

```
$ sudo yum install nice-dcv-web-viewer-2023.1.16388-1.el7.x86_64.rpm
```

- 64 位 ARM

```
$ sudo yum install nice-dcv-web-viewer-2023.1.16388-1.el7.aarch64.rpm
```

7. (可选) 要使用虚拟会话, 请安装 `nice-xdcv` 软件包。

- 64 位 x86

```
$ sudo yum install nice-xdcv-2023.1.565-1.el7.x86_64.rpm
```

- 64 位 ARM

```
$ sudo yum install nice-xdcv-2023.1.565-1.el7.aarch64.rpm
```

8. (可选) 如果您计划使用 GPU 共享, 请安装 `nice-dcv-gl` 程序包。

- 64 位 x86

```
$ sudo yum install nice-dcv-gl-2023.1.1047-1.el7.x86_64.rpm
```

Note

您可以有选择性地安装 `nice-dcv-gltest` 程序包。该软件包包含一个简单的 OpenGL 应用程序, 可用于确定您的虚拟会话是否正确配置为使用基于硬件的 OpenGL。

9. (可选) 如果您打算将 NICE DCV 与 NICE 配合使用 `EnginFrame`, 请安装该软件包。 `nice-dcv-simple-external-authenticator`

- 64 位 x86

```
$ sudo yum install nice-dcv-simple-external-  
authenticator-2023.1.228-1.el7.x86_64.rpm
```

- 64 位 ARM

```
$ sudo yum install nice-dcv-simple-external-  
authenticator-2023.1.228-1.el7.aarch64.rpm
```

10. (可选) 要支持使用 USB 远程控制的专用 USB 设备，请安装 DCV USB 驱动程序。

要安装 DCV USB 驱动程序，必须在服务器上安装动态内核模块支持 (DKMS)。使用以下命令安装 DKMS。

可从 Extra Packages for Enterprise Linux (EPEL) 存储库安装 DKMS。运行以下命令以启用 EPEL 存储库：

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

在启用 EPEL 存储库后，运行以下命令以安装 DKMS：

```
$ sudo yum install dkms
```

在安装 DKMS 后，运行以下命令以安装 DCV USB 驱动程序：

```
$ sudo dcvusbdriverinstaller
```

11. (可选) 如果您计划支持麦克风重定向，请验证是否在您的系统上安装了 pulseaudio-utils 软件包。请使用以下命令安装该软件包。

```
$ sudo yum install pulseaudio-utils
```

RHEL, CentOS, and Rocky Linux 8.5

NICE DCV 服务器适用于基于 64 位 x86 和 64 位 ARM 架构的 RHEL 和 CentOS 服务器以及 Rocky Linux 8.5 或更高版本。

⚠ Important

nice-dcv-gl 和 nice-dcv-gltest 软件包不适用于基于 64 位 ARM 架构的服务器。

在 RHEL、CentOS 和/或 Rocky Linux 8.5 上安装 NICE DCV 服务器

1. 启动并连接到您打算安装 NICE DCV 服务器的服务器。
2. NICE DCV 服务器软件包使用安全 GPG 签名进行数字签名。要允许软件包管理器验证软件包签名，您必须导入 NICE GPG 密钥。为此，打开一个终端窗口并导入 NICE GPG 密钥。

```
$ sudo rpm --import https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY
```

3. 从 [NICE DCV 下载网站](#) 中下载软件包。RPM 和 deb 程序包打包到一个 .tgz 存档中。确保为您的操作系统下载正确的归档。

- 64 位 x86

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2023.1/Servers/nice-dcv-2023.1-16388-el8-x86_64.tgz
```

- 64 位 ARM

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2023.1/Servers/nice-dcv-2023.1-16388-el8-aarch64.tgz
```

📘 Tip

下载网站的[最新软件包](#)页面包含始终指向最新可用版本的链接。您可以使用这些链接自动检索最新的 NICE DCV 软件包。

- 64 位 x86

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-el8-x86_64.tgz
```

- 64 位 ARM

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-el8-aarch64.tgz
```


4. 提取 .tgz 归档的内容，并导航到提取的目录。

- 64 位 x86

```
$ tar -xvzf nice-dcv-2023.1-16388-el8-x86_64.tgz && cd nice-dcv-2023.1-16388-el8-x86_64
```

- 64 位 ARM

```
$ tar -xvzf nice-dcv-2023.1-16388-el8-aarch64.tgz && cd nice-dcv-2023.1-16388-el8-aarch64
```

5. 安装 NICE DCV 服务器。

- 64 位 x86

```
$ sudo yum install nice-dcv-server-2023.1.16388-1.el8.x86_64.rpm
```

- 64 位 ARM

```
$ sudo yum install nice-dcv-server-2023.1.16388-1.el8.aarch64.rpm
```

6. (可选) 如果您计划将 Web 客户端与 NICE DCV 2021.2 和更高版本一起使用，请安装 nice-dcv-web-viewer 软件包。

- 64 位 x86

```
$ sudo yum install nice-dcv-web-viewer-2023.1.16388-1.el8.x86_64.rpm
```

- 64 位 ARM

```
$ sudo yum install nice-dcv-web-server-2023.1.16388-1.el8.aarch64.rpm
```

7. (可选) 要使用虚拟会话，请安装 nice-xdcv 软件包。

- 64 位 x86

```
$ sudo yum install nice-xdcv-2023.1.565-1.el8.x86_64.rpm
```

- 64 位 ARM

```
$ sudo yum install nice-xdcv-2023.1.565-1.el8.aarch64.rpm
```

8. (可选) 如果您计划使用 GPU 共享 , 请安装 nice-dcv-gl 程序包。

- 64 位 x86

```
$ sudo yum install nice-dcv-gl-2023.1.1047-1.el8.x86_64.rpm
```

Note

您可以有选择性地安装 nice-dcv-gltest 程序包。该软件包包含一个简单的 OpenGL 应用程序 , 可用于确定您的虚拟会话是否正确配置为使用基于硬件的 OpenGL。

9. (可选) 如果您打算将 NICE DCV 与 NICE 配合使用 EnginFrame , 请安装该软件包。nice-dcv-simple-external-authenticator

- 64 位 x86

```
$ sudo yum install nice-dcv-simple-external-authenticator-2023.1.228-1.el8.x86_64.rpm
```

- 64 位 ARM

```
$ sudo yum install nice-dcv-simple-external-authenticator-2023.1.228-1.el8.aarch64.rpm
```

10. (可选) 如果您计划使用 USB 远程控制支持专用 USB 设备 , 请安装 DCV USB 驱动程序。

要安装 DCV USB 驱动程序 , 必须在服务器上安装动态内核模块支持 (DKMS)。使用以下命令安装 DKMS。

可从 Extra Packages for Enterprise Linux (EPEL) 存储库安装 DKMS。运行以下命令以启用 EPEL 存储库 :

```
$ sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

在启用 EPEL 存储库后，运行以下命令以安装 DKMS：

```
$ sudo yum install dkms
```

在安装 DKMS 后，运行以下命令以安装 DCV USB 驱动程序：

```
$ sudo dcvusbdriverinstaller
```

11. (可选) 如果您计划支持麦克风重定向，请验证是否在您的系统上安装了 `pulseaudio-utils` 软件包。请使用以下命令安装该软件包。

```
$ sudo yum install pulseaudio-utils
```

RHEL, CentOS, and Rocky Linux 9

NICE DCV 服务器适用于基于 64 位 x86 和 64 位 ARM 架构的 RHEL 和 CentOS 服务器以及 Rocky Linux 9 或更高版本。

Important

`nice-dcv-gl` 和 `nice-dcv-gltest` 软件包不适用于基于 64 位 ARM 架构的服务器。

在 RHEL、CentOS 和/或 Rocky Linux 9 上安装 NICE DCV 服务器

1. 启动并连接到您打算安装 NICE DCV 服务器的服务器。
2. NICE DCV 服务器软件包使用安全 GPG 签名进行数字签名。要允许软件包管理器验证软件包签名，您必须导入 NICE GPG 密钥。为此，打开一个终端窗口并导入 NICE GPG 密钥。

```
$ sudo rpm --import https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY
```

3. 从 [NICE DCV 下载网站](#) 中下载软件包。RPM 和 deb 程序包打包到一个 `.tgz` 存档中。确保为您的操作系统下载正确的归档。

- 64 位 x86

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2023.1/Servers/nice-dcv-2023.1-16388-e19-x86_64.tgz
```

- 64 位 ARM

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2023.1/Servers/nice-dcv-2023.1-16388-el9-aarch64.tgz
```

 Tip

下载网站的[最新软件包](#)页面包含始终指向最新可用版本的链接。您可以使用这些链接自动检索最新的 NICE DCV 软件包。

- 64 位 x86

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-el9-x86_64.tgz
```

- 64 位 ARM

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-el9-aarch64.tgz
```

4. 提取 .tgz 归档的内容，并导航到提取的目录。

- 64 位 x86

```
$ tar -xvzf nice-dcv-2023.1-16388-el9-x86_64.tgz && cd nice-dcv-2023.1-16388-el9-x86_64
```

- 64 位 ARM

```
$ tar -xvzf nice-dcv-2023.1-16388-el9-aarch64.tgz && cd nice-dcv-2023.1-16388-el9-aarch64
```

5. 安装 NICE DCV 服务器。

- 64 位 x86

```
$ sudo yum install nice-dcv-server-2023.1.16388-1.el9.x86_64.rpm
```

- 64 位 ARM

```
$ sudo yum install nice-dcv-server-2023.1.16388-1.el9.aarch64.rpm
```

6. (可选) 如果您计划将 Web 客户端与 NICE DCV 2021.2 和更高版本一起使用, 请安装 `nice-dcv-web-viewer` 软件包。

- 64 位 x86

```
$ sudo yum install nice-dcv-web-viewer-2023.1.16388-1.el9.x86_64.rpm
```

- 64 位 ARM

```
$ sudo yum install nice-dcv-web-server-2023.1.16388-1.el9.aarch64.rpm
```

7. (可选) 要使用虚拟会话, 请安装 `nice-xdcv` 软件包。

- 64 位 x86

```
$ sudo yum install nice-xdcv-2023.1.565-1.el9.x86_64.rpm
```

- 64 位 ARM

```
$ sudo yum install nice-xdcv-2023.1.565-1.el9.aarch64.rpm
```

8. (可选) 如果您计划使用 GPU 共享, 请安装 `nice-dcv-gl` 程序包。

- 64 位 x86

```
$ sudo yum install nice-dcv-gl-2023.1.1047-1.el9.x86_64.rpm
```

Note

您可以有选择性地安装 `nice-dcv-gltest` 程序包。该软件包包含一个简单的 OpenGL 应用程序, 可用于确定您的虚拟会话是否正确配置为使用基于硬件的 OpenGL。

9. (可选) 如果您打算将 NICE DCV 与 NICE 配合使用 `EnginFrame`, 请安装该软件包。 `nice-dcv-simple-external-authenticator`

- 64 位 x86

```
$ sudo yum install nice-dcv-simple-external-  
authenticator-2023.1.228-1.el9.x86_64.rpm
```

- 64 位 ARM

```
$ sudo yum install nice-dcv-simple-external-  
authenticator-2023.1.228-1.el9.aarch64.rpm
```

10. (可选) 如果您计划使用 USB 远程控制支持专用 USB 设备，请安装 DCV USB 驱动程序。

要安装 DCV USB 驱动程序，必须在服务器上安装动态内核模块支持 (DKMS)。使用以下命令安装 DKMS。

可从 Extra Packages for Enterprise Linux (EPEL) 存储库安装 DKMS。运行以下命令以启用 EPEL 存储库：

```
$ sudo yum install https://dl.fedoraproject.org/pub/epel/epel-release-  
latest-9.noarch.rpm
```

在启用 EPEL 存储库后，运行以下命令以安装 DKMS：

```
$ sudo yum install dkms
```

在安装 DKMS 后，运行以下命令以安装 DCV USB 驱动程序：

```
$ sudo dcvusbdriverinstaller
```

11. (可选) 如果您计划支持麦克风重定向，请验证是否在您的系统上安装了 pulseaudio-utils 软件包。请使用以下命令安装该软件包。

```
$ sudo yum install pulseaudio-utils
```

SLES 12.x/15.x

NICE DCV 服务器仅适用于基于 64 位 x86 架构的 SUSE Linux Enterprise Server (SLES) 12.x/15.x 服务器。

在 SLES 12.x/15.x 上安装 NICE DCV 服务器

1. 启动并连接到您打算安装 NICE DCV 服务器的服务器。
2. NICE DCV 服务器软件包使用安全 GPG 签名进行数字签名。要允许软件包管理器验证软件包签名，您必须导入 NICE GPG 密钥。为此，打开一个终端窗口并导入 NICE GPG 密钥。

```
$ sudo rpm --import https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY
```

3. 从 [NICE DCV 下载网站](#) 中下载软件包。RPM 和 deb 程序包打包到一个 .tgz 存档中。确保为您的操作系统下载正确的归档。

- SLES 12.x

```
$ curl -O https://d1uj6qtbmh3dt5.cloudfront.net/2023.1/Servers/nice-dcv-2023.1-16388-sles12-x86_64.tgz
```

- SLES 15.x

```
$ curl -O https://d1uj6qtbmh3dt5.cloudfront.net/2023.1/Servers/nice-dcv-2023.1-16388-sles15-x86_64.tgz
```

Tip

下载网站的[最新软件包](#)页面包含始终指向最新可用版本的链接。您可以使用这些链接自动检索最新的 NICE DCV 软件包。

- SLES 12.x

```
$ curl -O https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-sles12-x86_64.tgz
```

- SLES 15.x

```
$ curl -O https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-sles15-x86_64.tgz
```

4. 提取 .tgz 归档的内容，并导航到提取的目录。

- SLES 12.x

```
$ tar -xvzf nice-dcv-2023.1-16388-sles12-x86_64.tgz && cd nice-dcv-2023.1-16388-sles12-x86_64
```

- SLES 15.x

```
$ tar -xvzf nice-dcv-2023.1-16388-sles15-x86_64.tgz && cd nice-dcv-2023.1-16388-sles15-x86_64
```

5. 安装 NICE DCV 服务器。

- SLES 12.x

```
$ sudo zypper install nice-dcv-server-2023.1.16388-1.sles12.x86_64.rpm
```

- SLES 15.x

```
$ sudo zypper install nice-dcv-server-2023.1.16388-1.sles15.x86_64.rpm
```

6. (可选) 如果您计划将 Web 客户端与 NICE DCV 2021.2 和更高版本一起使用，请安装 nice-dcv-web-viewer 软件包。

- SLES 12.x

```
$ sudo zypper install nice-dcv-web-viewer-2023.1.16388-1.sles12.x86_64.rpm
```

- SLES 15.x

```
$ sudo zypper install nice-dcv-web-viewer-2023.1.16388-1.sles15.x86_64.rpm
```

7. (可选) 要使用虚拟会话，请安装 nice-xdcv 软件包。

- SLES 12.x

```
$ sudo zypper install nice-xdcv-2023.1.565-1.sles12.x86_64.rpm
```

- SLES 15.x

```
$ sudo zypper install nice-xdcv-2023.1.565-1.sles15.x86_64.rpm
```

8. (可选) 如果您计划使用 GPU 共享，请安装 nice-dcv-g1 程序包。

- SLES 12.x

```
$ sudo zypper install nice-dcv-gl-2023.1.1047-1.sles12.x86_64.rpm
```

- SLES 15.x

```
$ sudo zypper install nice-dcv-gl-2023.1.1047-1.sles15.x86_64.rpm
```

Note

您可以有选择性地安装 `nice-dcv-gltest` 程序包。此程序包中包含一个简单的 OpenGL 应用程序，该应用程序可用于确定虚拟会话是否已正确配置为使用基于硬件的 OpenGL。

9. (可选) 如果您打算将 NICE DCV 与 NICE 配合使用 EnginFrame，请安装该软件包。 `nice-dcv-simple-external-authenticator`

- SLES 12.x

```
$ sudo zypper install nice-dcv-simple-external-authenticator-2023.1.228-1.sles12.x86_64.rpm
```

- SLES 15.x

```
$ sudo zypper install nice-dcv-simple-external-authenticator-2023.1.228-1.sles15.x86_64.rpm
```

10. (可选) 如果您计划使用 USB 远程控制支持专用 USB 设备，请安装 DCV USB 驱动程序。

要安装 DCV USB 驱动程序，必须在服务器上安装动态内核模块支持 (DKMS)。使用以下命令安装 DKMS。

运行以下命令以安装 DKMS：


- SLES 12.x

```
$ sudo zypper install http://download.opensuse.org/repositories/home:/Ximi1970:/Dkms:/Staging/SLE_12_SP4/noarch/dkms-2.5-11.1.noarch.rpm
```

- SLES 15

启用 PackageHub 存储库。

```
$ sudo SUSEConnect -p PackageHub/15/x86_64
```

 Note

如果您使用 SLES 15 SP1 或 SP2，请将上面命令中的 **15** 替换为 15.1 或 15.2。

安装 DKMS。

```
$ sudo zypper refresh
```

```
$ sudo zypper install dkms
```

安装内核源。

```
$ sudo zypper install -y kernel-source
```

重启实例。

```
$ sudo reboot
```

安装 DKMS 后，运行以下命令安装 DCV USB 驱动程序：

```
$ sudo dcvusbdriverinstaller
```

11. (可选) 如果您计划支持麦克风重定向，请验证是否在您的系统上安装了 pulseaudio-utils 软件包。请使用以下命令安装该软件包。

```
$ sudo zypper install pulseaudio-utils
```

Ubuntu 20.04/22.04

NICE DCV 服务器适用于基于 64 位 x86 和 64 位 ARM 架构的 Ubuntu 服务器。

⚠ Important

nice-dcv-gl 和 nice-dcv-gltest 软件包不适用于基于 64 位 ARM 架构的服务器。

在 Ubuntu 20.04/22.04 上安装 NICE DCV 服务器

1. 启动并连接到您打算安装 NICE DCV 服务器的服务器。
2. NICE DCV 服务器软件包使用安全 GPG 签名进行数字签名。要允许软件包管理器验证软件包签名，您必须导入 NICE GPG 密钥。为此，打开一个终端窗口并导入 NICE GPG 密钥。

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/NICE-GPG-KEY
```

```
$ gpg --import NICE-GPG-KEY
```

3. 从 [NICE DCV 下载网站](#) 中下载软件包。RPM 和 deb 程序包打包到一个 .tgz 存档中。确保为您的操作系统下载正确的归档。

- Ubuntu 20.04 (64 位 x86)

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2023.1/Servers/nice-dcv-2023.1-16388-ubuntu2004-x86_64.tgz
```

- Ubuntu 22.04 (64 位 x86)

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2023.1/Servers/nice-dcv-2023.1-16388-ubuntu2204-x86_64.tgz
```

- Ubuntu 22.04 (64 位 ARM)

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2023.1/Servers/nice-dcv-2023.1-16388-ubuntu2204-aarch64.tgz
```

i Tip

下载网站的[最新软件包](#)页面包含始终指向最新可用版本的链接。您可以使用这些链接自动检索最新的 NICE DCV 软件包。

- Ubuntu 20.04 (64 位 x86)

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-ubuntu2004-x86_64.tgz
```

- Ubuntu 22.04 (64 位 x86)

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-ubuntu2204-x86_64.tgz
```

- Ubuntu 22.04 (64 位 ARM)

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-ubuntu2204-aarch64.tgz
```

4. 提取 .tgz 归档的内容，并导航到提取的目录。

- Ubuntu 20.04 (64 位 x86)

```
$ tar -xvzf nice-dcv-2023.1-16388-ubuntu2004-x86_64.tgz && cd nice-dcv-2023.1-16388-ubuntu2004-x86_64
```

- Ubuntu 22.04 (64 位 x86)

```
$ tar -xvzf nice-dcv-2023.1-16388-ubuntu2204-x86_64.tgz && cd nice-dcv-2023.1-16388-ubuntu2204-x86_64
```

- Ubuntu 22.04 (64 位 ARM)

```
$ tar -xvzf nice-dcv-2023.1-16388-ubuntu2204-aarch64.tgz && cd nice-dcv-2023.1-16388-ubuntu2204-aarch64
```

5. 安装 NICE DCV 服务器。

- Ubuntu 20.04 (64 位 x86)

```
$ sudo apt install ./nice-dcv-server_2023.1.16388-1_amd64.ubuntu2004.deb
```

- Ubuntu 22.04 (64 位 x86)

```
$ sudo apt install ./nice-dcv-server_2023.1.16388-1_amd64.ubuntu2204.deb
```

- Ubuntu 22.04 (64 位 ARM)

```
$ sudo apt install ./nice-dcv-server_2023.1.16388-1_arm64.ubuntu2204.deb
```

6. (可选) 如果您计划将 Web 客户端与 NICE DCV 2021.2 和更高版本一起使用，请安装 nice-dcv-web-viewer 软件包。

- Ubuntu 20.04 (64 位 x86)

```
$ sudo apt install ./nice-dcv-web-viewer_2023.1.16388-1_amd64.ubuntu2004.deb
```

- Ubuntu 22.04 (64 位 x86)

```
$ sudo apt install ./nice-dcv-web-viewer_2023.1.16388-1_amd64.ubuntu2204.deb
```

- Ubuntu 22.04 (64 位 ARM)

```
$ sudo apt install ./nice-dcv-web-viewer_2023.1.16388-1_arm64.ubuntu2204.deb
```

7. 将 dcv 用户添加到 video 组。

```
$ sudo usermod -aG video dcv
```

8. (可选) 如果您计划使用虚拟会话，请安装 nice-xdcv 程序包。

- Ubuntu 20.04 (64 位 x86)

```
$ sudo apt install ./nice-xdcv_2023.1.565-1_amd64.ubuntu2004.deb
```

- Ubuntu 22.04 (64 位 x86)

```
$ sudo apt install ./nice-xdcv_2023.1.565-1_amd64.ubuntu2204.deb
```


- Ubuntu 22.04 (64 位 ARM)

```
$ sudo apt install ./nice-xdcv_2023.1.565-1_arm64.ubuntu2204.deb
```

9. (可选) 如果您计划使用 GPU 共享 , 请安装 nice-dcv-gl 程序包。

- Ubuntu 22.04 (64 位 x86)

```
$ sudo apt install ./nice-dcv-gl_2023.1.1047-1_amd64.ubuntu2204.deb
```

 Note

您可以有选择性地安装 nice-dcv-gltest 程序包。该软件包包含一个简单的 OpenGL 应用程序 , 可用于确定您的虚拟会话是否正确配置为使用基于硬件的 OpenGL。

10. (可选) 如果您打算将 NICE DCV 与 NICE 配合使用 EnginFrame , 请安装该软件包。nice-dcv-simple-external-authenticator

- Ubuntu 20.04 (64 位 x86)

```
$ sudo apt install ./nice-dcv-simple-external-authenticator_2023.1.228-1_amd64.ubuntu2004.deb
```

- Ubuntu 22.04 (64 位 x86)

```
$ sudo apt install ./nice-dcv-simple-external-authenticator_2023.1.228-1_amd64.ubuntu2204.deb
```

- Ubuntu 22.04 (64 位 ARM)

```
$ sudo apt install ./nice-dcv-simple-external-authenticator_2023.1.228-1_arm64.ubuntu2204.deb
```

11. (可选) 如果您计划使用 USB 远程控制支持专用 USB 设备 , 请安装 DCV USB 驱动程序。

要安装 DCV USB 驱动程序 , 必须在服务器上安装动态内核模块支持 (DKMS)。使用以下命令安装 DKMS。

可以在官方 Ubuntu 存储库中找到 DKMS。运行以下命令以安装 DKMS :

```
$ sudo apt install dkms
```

在安装 DKMS 后，运行以下命令以安装 DCV USB 驱动程序：

```
$ sudo dcvusbdriverinstaller
```

12. (可选) 如果您计划支持麦克风重定向，请验证是否在您的系统上安装了 `pulseaudio-utils` 软件包。请使用以下命令安装该软件包。

```
$ sudo apt install pulseaudio-utils
```

安装后检查

本主题提供了一些安装后检查，您应该在安装 NICE DCV 后执行这些检查，以确保正确配置了 NICE DCV 服务器。

内容

- [确保可以访问 NICE DCV 服务器](#)
- [确保可以访问 X 服务器](#)
- [验证是否正确安装了 DCV GL](#)
- [验证 NICE DCV DEB 软件包签名](#)

确保可以访问 NICE DCV 服务器

默认情况下，NICE DCV 服务器配置为通过 TCP 端口 8443 进行通信。确保可通过此端口访问该服务器。如果您的防火墙禁止通过端口 8443 进行访问，您必须更改 NICE DCV 服务器进行通信时使用的端口。有关更多信息，请参阅 [更改 NICE DCV 服务器 TCP/UDP 端口和侦听地址](#)。

此外，如果要在 EC2 实例上设置 NICE DCV，请创建一个安全组。这是为了允许访问 NICE DCV 服务器进行通信时使用的端口。有关更多信息，请参阅 [如何在 EC2 上配置安全组](#)。

确保可以访问 X 服务器

您必须确保 NICE DCV 控制台会话和虚拟会话可以访问 X 服务器。

控制台会话

在安装 NICE DCV 服务器后，将创建一个 `dcv` 用户。确保该用户可以访问 X 服务器。

验证 **dcv** 用户能否访问 X 服务器

运行以下命令：

```
$ sudo DISPLAY=:0 XAUTHORITY=$(ps aux | grep "X.*\ -auth" | grep -v grep | sed -n 's/.*-auth \([^ ]+\)\.*/\1/p') xhost | grep "SI:localuser:dcv$"
```

如果命令返回 `SI:localuser:dcv`，则表明该 `dcv` 用户可以访问 X 服务器。

如果该命令未返回 `SI:localuser:dcv`，则 `dcv` 用户无权访问 X 服务器。运行以下命令重新启动 X 服务器：

- RHEL、CentOS、Amazon Linux 2、Ubuntu 18.x 和 SUSE Linux Enterprise 12.x

```
$ sudo systemctl isolate multi-user.target
```

```
$ sudo systemctl isolate graphical.target
```

虚拟会话

如果您安装了 DCV GL 程序包，则必须确保本地用户可以访问 X 服务器。这可确保 OpenGL 硬件加速正确用于虚拟会话。

验证本地用户能否访问 X 服务器

运行以下命令：

```
$ sudo DISPLAY=:0 XAUTHORITY=$(ps aux | grep "X.*\ -auth" | grep -v grep | sed -n 's/.*-auth \([^ ]+\)\.*/\1/p') xhost | grep "LOCAL:$"
```

如果命令返回 `LOCAL:`，则表明本地用户可以访问 X 服务器。

如果该命令未返回 `LOCAL:`，则本地用户无权访问 X 服务器。运行以下命令来重新启动 X 服务器，并禁用和重新启用 DCV GL：

- RHEL、CentOS、Amazon Linux 2、Ubuntu 18.x 和 SUSE Linux Enterprise 12.x

```
$ sudo systemctl isolate multi-user.target
```



```
$ sudo dcvgladmin disable
```

```
$ sudo dcvgladmin enable
```

```
$ sudo systemctl isolate graphical.target
```

验证是否正确安装了 DCV GL

当您安装 DCV GL 程序包时，会自动安装 `dcvgldiag` 实用程序。可以使用该实用程序检查 Linux 服务器配置是否符合 DCV GL 要求。

运行 `dcvgldiag` 实用程序

使用以下命令：

```
$ sudo dcvgldiag
```

该实用程序会返回警告和错误列表并给出可能的解决方案。

验证 NICE DCV DEB 软件包签名

在安装 NICE DCV 后，您可以验证 Debian 软件包（DEB）上的签名。该验证过程需要使用 GPG 版本 1。

验证 DEB 软件包签名

使用以下命令：

```
gpg1 --import NICE-GPG-KEY-SECRET  
dpkg-sig --verify nice-dcv-server_2023.1.16388-1_amd64.deb
```

这会返回一条消息，其中包含 GOODSIG 术语以确认验证了签名。以下示例显示了签名确认消息。将显示实际密钥以替代 `####`。

```
Processing nice-dcv-server_2017.0.0-1_amd64.deb...  
GOODSIG _gpgbuilder Example Key
```

为 NICE DCV 服务器授予许可

NICE DCV 许可要求因安装和使用 NICE DCV 服务器的位置而异。

Important

以下许可要求仅适用于 NICE DCV 2017.0 和更高版本。

NICE DCV 许可要求

主题

- [Amazon EC2 上的 NICE DCV](#)
- [NICE DCV 的其他使用案例](#)
- [远程访问 Windows Server 的 Microsoft 许可要求](#)

Amazon EC2 上的 NICE DCV

您不需要许可证服务器即可在 EC2 实例（包括在本地 AWS 区域上运行的实例）上安装 AWS Outposts 和使用 NICE DCV 服务器。NICE DCV 服务器自动检测它是否在 Amazon EC2 实例上运行，并定期连接到 S3 存储桶以确定是否具有有效的许可证。

确保您的实例具有以下属性：

- 它可以访问 Amazon S3 终端节点。如果它有权访问互联网，则会使用 Amazon S3 公有终端节点进行连接。如果您的实例无权访问互联网，请使用出站安全组规则或访问控制列表（ACL）策略为您的 VPC 配置一个网关终端节点，以允许您通过 HTTPS 访问 Amazon S3。有关更多信息，请参阅《Amazon VPC 用户指南》中的[网关端点](#)。如果您在连接到 S3 存储桶时遇到任何问题，请参阅 AWS 知识中心的[为什么我无法使用网关 VPC 终端节点连接到 S3 存储桶？](#)
- 它有权访问所需的 Amazon S3 对象。将以下 Amazon S3 访问策略添加到实例的 IAM 角色中，并将##占位符替换为您的 AWS 区域（例如us-east-1）。有关更多信息，请参阅[创建 IAM 角色](#)。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "s3:GetObject",
```

```

        "Resource": "arn:aws:s3:::dcv-license.region/*"
    }
]
}

```

- 如果您使用 Windows 实例，请确保该实例可以访问实例元数据服务。需要访问该服务，以确保可以正确为 NICE DCV 服务器授予许可。有关实例元数据服务的更多信息，请参阅 Amazon EC2 [用户指南中的实例元数据和用户数据](#)。

如果您使用自定义 Windows AMI，您必须安装 EC2Config 服务（Windows Server 2012 R2 和更早版本）或 EC2Launch（Windows Server 2016 和更高版本）。这将确保您的实例能够访问实例元数据服务。有关更多信息，请参阅亚马逊 EC2 [用户指南中的使用 EC2Config 服务配置 Windows 实例或使用 EC2Launch 配置 Windows 实例](#)。

如果要在 Amazon EC2 实例上安装和使用 NICE DCV 服务器，您可以跳过本章的其余部分。本章的其余部分仅适用于 NICE DCV 服务器的所有其他使用案例。

NICE DCV 的其他使用案例

对于所有其他使用案例，需要具有许可证才能安装和使用 NICE DCV 服务器。提供了以下许可选项：

- 自动评估许可证 - 在您安装 NICE DCV 服务器时，将自动安装这种类型的许可证。这种类型的许可证在安装后的 30 天内有效。在许可证过期后，您无法再在服务器上创建和托管 NICE DCV 会话。这些许可证适合短期测试和评估。要测试一段较长的时间，可请求延期评估许可证。

Note

如果未配置其他许可证，NICE DCV 服务器默认使用自动评估许可证。

- 延期评估许可证 - 延期评估许可证是一种评估许可证，它延长自动评估许可证提供的初始 30 天评估期。该期限由 NICE case-by-case 根据情况确定。在到达到期日期后，延期评估许可证将失效，并且您无法再在服务器上创建和托管 NICE DCV 会话。必须向 NICE 网站的 [How to Buy](#) 页面上列出的 NICE 分销商或经销商申请延期评估许可证。这些许可证以许可证文件形式提供，必须将该文件安装在 NICE DCV 服务器上。
- 生产许可证 - 生产许可证是您从 NICE 购买的完整许可证。生产许可证是由许可证服务器管理的浮动许可证。在使用浮动许可证时，您可以在网络中运行多个 NICE DCV 服务器。同时，您还可以限制在所有服务器上创建的并发 NICE DCV 会话数。对于每个并发 NICE DCV 会话，您需要使用一个许可证。生产许可证作为许可证文件分发，您必须将它安装在 Reprise License Manager (RLM) 服务器上。提供了两种类型的生产许可证：

- 永久许可证 - 永久许可证没有到期日期，可以无限期使用。
- 订阅 - 订阅的有效期是有限的，通常为一年。许可证文件中指明了许可证的到期日期。在许可证过期后，您无法再在 NICE DCV 服务器上创建和托管 NICE DCV 会话。

有关如何购买 NICE DCV 永久许可证或订阅的信息，请参阅 NICE 网站上的 [How to Buy](#) 并查找您所在区域的 NICE 分销商或经销商。

许可要求

- NICE DCV 客户端不需要具有许可证。
- NICE DCV 服务器许可证文件与以前版本的 NICE DCV 服务器向后兼容。例如，您可以将 NICE DCV 服务器版本 2021 许可证与 NICE DCV 服务器版本 2019 一起使用。
- NICE DCV 服务器版本至少需要具有相同版本的 NICE DCV 服务器许可证。例如，如果您使用 NICE DCV 服务器版本 2021，则需要具有许可证 2021 或更高版本。如果升级到更高版本的 NICE DCV 服务器，您必须申请兼容的许可证文件。有关更多信息，请与您的 NICE DCV 分销商或经销商联系。

Note

有关 NICE DCV 服务器兼容性的信息，请参阅[兼容性注意事项](#)。

远程访问 Windows Server 的 Microsoft 许可要求

Microsoft 要求，除了 Windows Server 客户端访问许可证 (CAL) 以外，对于远程访问服务器的图形用户界面 (GUI) 的每个用户，您还必须具有适用于您的 Windows Server 版本的 Windows Server 远程桌面服务 (RDS) CAL。这与您使用的远程显示协议无关。如果您使用 NICE DCV 访问远程 Windows Server 主机的 GUI，也需要具有该许可证。

如果您在 Amazon EC2 实例上运行 NICE DCV 服务器并使用 [Windows Server AMI](#)，则 Amazon 承担 Windows Server CAL 的许可成本，并提供两个仅用于管理用途的 Windows Server RDS CAL。这仅用于测试、维护和管理。

有关更多信息，请参阅 [Microsoft 产品条款网站](#)。如果您对 Microsoft 软件的许可或权利有疑问，请咨询您的法律团队、Microsoft 或 Microsoft 经销商。

安装延期评估许可证

在您向 NICE 申请延期评估许可证时，您会收到一个定义该许可证的 `license.lic` 文件。

安装延期评估许可证

将以下文件夹中的 `license.lic` 文件放在您的服务器上：

- Windows 服务器

```
C:\Program Files\NICE\DCV\Server\license\license.lic
```

- Linux 服务器

```
/usr/share/dcv/license/license.lic
```

或者，要将 `license.lic` 放置在服务器上的其他文件夹中，您必须更新 `license-file` 配置参数，以使其指定许可证文件的完整路径。

主题

- [更改 Windows 服务器上的许可证路径](#)
- [更改 Linux 服务器上的许可证路径](#)

更改 Windows 服务器上的许可证路径

更新 Windows 服务器上的 `license-file` 配置参数

1. 打开 Windows 注册表编辑器。
2. 导航到 `HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/license/` 注册表项，并选择 `license-file` 参数。

如果该注册表项中没有 `license-file` 参数，请创建一个：

- a. 在左侧面板中打开 `license` 注册表项的上下文（右键单击）菜单，然后依次选择新建和字符串值。
 - b. 对于名称，输入 `license-file` 并按 Enter。
3. 打开 `license-file` 参数。对于值数据，输入 `license.lic` 文件的完整路径。
 4. 选择确定，并关闭 Windows 注册表编辑器。

更改 Linux 服务器上的许可证路径

更新 Linux 服务器上的 **license-file** 配置参数

1. 导航到 `/etc/dcv/`，并使用您常用的文本编辑器打开 `dcv.conf`。
2. 找到 `license-file` 部分中的 `[license]` 参数，并使用 `license.lic` 文件的新完整路径替换现有路径。

如果 `license-file` 部分中没有 `[license]` 参数，则使用以下格式手动添加此参数：

```
license-file = "/custom-path/license.lic"
```

3. 保存并关闭文件。

安装生产许可证

本主题中的以下几节介绍了如何购买和使用生产许可证（永久许可证或订阅）。

主题

- [步骤 1：安装 RLM 服务器](#)
- [步骤 2：获取 RLM 服务器的主机 ID](#)
- [步骤 3：购买永久许可证或订阅](#)
- [步骤 4：修改许可证文件](#)
- [步骤 5：配置 RLM 服务器](#)
- [步骤 6：配置 NICE DCV 服务器](#)

步骤 1：安装 RLM 服务器

当您购买永久许可证或订阅时，您将获得一个定义许可证条款的许可证文件。您必须在 Reprise License Manager (RLM) 服务器上安装该许可证文件。

有关 RLM 的更多信息，请参阅 [Reprise Software](#) 网站。

主题

- [在 Windows 上安装 RLM 服务器](#)
- [在 Linux 上安装 RLM 服务器](#)

在 Windows 上安装 RLM 服务器

在 Windows 上安装 RLM 服务器

1. 从 [Reprise Software 网站](#) 下载 RLM 许可证管理包。

Note

从 NICE DCV 版本 2022.1 开始，需要使用 14 和更高版本的 RLM 服务器。以前的版本要求使用 12 和更高版本的 RLM。
建议安装最新稳定版本的 RLM 许可证管理包。

2. 将 RLM 许可证管理包安装到 C:\RLM。

在 Linux 上安装 RLM 服务器

在 Linux 上安装 RLM 服务器

1. 从 [Reprise Software 网站](#) 下载 RLM 许可证管理包。

Note

从 NICE DCV 版本 2022.1 开始，需要使用 14 和更高版本的 RLM 服务器。以前的版本要求使用 12 和更高版本的 RLM。
建议安装最新稳定版本的 RLM 许可证管理包。

2. 创建用户组和 rlm 用户。这可以是任何有效的用户或服务账户。我们强烈建议您不要将该值设置为根账户。

```
$ groupadd -r rlm
```

```
$ useradd -r -g rlm -d "/opt/nice/rlm" -s /sbin/nologin -c "RLM License Server" rlm
```

3. 创建 RLM 服务器所需的 /opt/nice/rlm 和 /opt/nice/rlm/license 目录。

```
$ mkdir -p /opt/nice/rlm/license
```

4. 将 RLM 许可证管理包的内容提取到 /opt/nice/rlm/，并确保文件由 rlm 用户拥有。

```
$ tar xvf x64_l1.admin.tar.gz -C /opt/nice/rlm/ --strip-components 1
```

```
$ chown -R rlm:rlm /opt/nice/rlm
```

步骤 2：获取 RLM 服务器的主机 ID

安装 RLM 服务器后，您必须获取 RLM 服务器的主机 ID。在购买永久许可证或订阅时，您需要提供该主机 ID。

在 Windows 上获取 RLM 服务器主机 ID

要获取服务器的主机 ID，请打开命令提示符。

导航到 C:\RLM\，然后运行以下命令。

```
C:\> rlmutil.exe rlmhostid ether
```

该命令返回 RLM 服务器的主机 ID，如下所示。

```
Hostid of this machine: 06814example
```

记下主机 ID。您需要在下一步中使用该 ID。

在 Linux 上获取 RLM 服务器主机 ID

获取服务器的主机 ID，返回一次

1. 导航到 /opt/nice/rlm/。
2. 运行以下命令：

```
$ ./rlmutil rlmhostid ether
```

该命令返回每个网络接口的 RLM 服务器主机 ID，如下所示。

3. 记录主机 ID。您需要在下一步中使用该 ID。

Example

运行该过程并返回下面的单个 ID：


```
Hostid of this machine: 0a1b2c3d4e5f
```

然后，记下该 ID 并使用该 ID 购买 DCV 许可证。

获取服务器的主机 ID，返回多次

1. 导航到 `/opt/nice/rlm/`。
2. 运行以下命令：

```
$ ./rlmutil rlmhostid ether
```

将在 ID 列表中返回多个 ID。

3. 运行以下命令。

```
iface=$(route -n | grep " UG " | tr -s " " | cut -d" " -f8)
ip link show $iface | grep link/ether | tr -s " " | cut -d" " -f3 | tr -d ":"
```

该命令应返回网关网络接口的 RLM 服务器主机 ID。

4. 记录主机 ID。您需要在下一步中使用该 ID。

Example

运行该过程并在包含多个 ID 的列表中返回多个 ID：

```
Hostid of this machine: 0a1b2c3d4e5f 1b2c3d4e5f6a 2c3d4e5f6a7b 3d4e5f6a7b8c
```

运行 `interface` 命令并返回以下 ID：

```
Hostid of this machine: 0a1b2c3d4e5f
```

然后，记下该 ID 并使用该 ID 购买 DCV 许可证。

步骤 3：购买永久许可证或订阅

有关如何购买 NICE DCV 永久许可证或订阅的信息，请参阅 NICE 网站上的 [How to Buy](#) 并查找您所在区域的 NICE 分销商或经销商。

您必须提供 RLM 服务器的主机 ID。主机 ID 嵌入在 NICE 提供的许可证文件中。

步骤 4：修改许可证文件

在您购买 NICE DCV 永久许可证或订阅时，您会收到一个定义许可证的 `license.lic` 文件。`license.lic` 文件包含以下信息：

- RLM 服务器的主机名。
- 您在购买许可证时提供的 RLM 服务器主机 ID。
- RLM 服务器的 TCP 端口号。默认值为 5053。
- ISV 端口号。这是一个可选端口，RLM 服务器在该端口上侦听 NICE DCV 许可证请求。如果未指定，RLM 在启动时选择一个随机端口。
- 许可证涵盖的 NICE DCV 产品以及每个产品的以下详细信息：
 - 许可证涵盖的主要版本（例如，2017 NICE DCV 产品为 2017）。
 - 到期日期。Permanent 表示许可证不会过期。
 - 最大并发会话数（例如，10 表示服务器上的 10 个并发会话）。
 - 许可证校验和。
 - 许可证签名。

下面的代码块显示了 `license.lic` 文件的格式：

```
HOST RLM_server_hostname RLM_server_host_id RLM_server_port
ISV nice port=port_number
LICENSE product_1 major_version expiration_date concurrent_sessions share=hi
  _ck=checksum sig="signature"
LICENSE product_2 major_version expiration_date concurrent_sessions share=hi
  _ck=checksum sig="signature"
```

以下代码块显示了已省略 ISV 端口的 `license.lic` 文件的示例。许可证文件包括两个 NICE 产品（DCV 和 `dvc-gl`）的许可证。

```
HOST My-RLM-server abcdef123456 5053
ISV nice
LICENSE nice dcv 2017 permanent 10 share=hi _ck=456789098a
  sig="abcdefghijklmnopqrstuvwxy1234567890abcdefghijklmnopqrstuvwxy1234567890ab"
LICENSE nice dcv-gl 2017 permanent 10 share=hi _ck=123454323x
  sig="1234567890abcdefghijklmnopqrstuvwxy1234567890abcdefghijklmnopqrstuvwxy12"
```

编辑 `license.lic` 文件

1. 使用常用的文本编辑器打开 文件。
2. 在该文件中的第一行 (该行以 HOST 开头) 中添加您的 RLM 服务器的主机名和 TCP 端口号。

Warning

`RLM_server_host_id` 是您在购买许可证时提供的主机 ID。您无法编辑 `RLM_server_host_id`。

3. (可选) 在文件中以 ISV 开头的行中 , 添加 `port=port_number` 以添加 ISV 端口号。需要使用该端口才能与 DCV 服务器进行通信。

如果您不希望指定 ISV 端口 , 请省略 `port=port_number`。如果您未指定 ISV 端口 , RLM 在每次启动时使用随机端口。

Warning

如果您的防火墙设置禁止使用随机选择的端口 , 则除了 HOST 行中指定的 RLM 端口以外 , 您还需要指定随机选择的端口并配置防火墙以启用该端口。

4. 保存并关闭文件。

Warning

编辑许可证文件的任何其他部分会破坏文件的签名并使许可证无效。

步骤 5 : 配置 RLM 服务器

在修改许可证文件后 , 您必须将其放置在 RLM 服务器上 , 然后启动 RLM 服务。

主题

- [在 Windows 上配置 RLM 服务器](#)
- [在 Linux 上配置 RLM 服务器](#)

在 Windows 上配置 RLM 服务器

在 Windows 上配置 RLM 服务器

1. 连接到您的 RLM 服务器。
2. 将编辑后的 `license.lic` 文件复制到 `C:\RLM\license\`。
3. 从 NICE DCV 服务器中复制 `C:\Program Files\NICE\DCV\Server\license\nice.set` 文件，并将其放置在 RLM 服务器上的 `C:\RLM\` 文件夹中。
4. 将 RLM 服务器安装为 Windows 服务。

```
C:\> rlm.exe -nows -dlog C:\RLM\rlm.log -c C:\RLM\license -install_service -  
service_name dcv-rlm
```

有关 RLM 启动选项的更多信息，请参阅 [Reprise Software License Manager \(RLM \)](#) 产品页面。

5. 启动 RLM 服务器。

```
C:\> net start dcv-rlm
```

6. 确认 RLM 服务器正在运行。
 - a. 使用您的首选文本编辑器打开 `C:\RLM\nice.dlog`，并确认显示以下行。

```
date_time (nice) Server started on license1 (hostid: host_id) for: dcv dcv-gl
```

Note

`rlm.log` 文件的内容可能会随 RLM 服务器版本而异。

- b. 运行以下命令。

```
C:\RLM\rlmutil rlmstat -a -c rlm_server_hostname@5053
```

该命令返回有关 RLM 服务器的信息。

在 Linux 上配置 RLM 服务器

在 Linux 上配置 RLM 服务器

1. 将编辑后的 `license.lic` 文件复制到 `/opt/nice/rlm/license/`。
2. 从 NICE DCV 服务器中复制 `/usr/share/dcv/license/nice.set` 文件，并将其放置在 RLM 服务器上的 `/opt/nice/rlm` 中。
3. 创建 RLM 服务器服务，并确保该服务在启动时自动启动。
 - a. 在 `dcv-rlm` 文件夹中创建名为 `/opt/nice/rlm/` 的文件：

```
$ touch /opt/nice/rlm/dcv-rlm
```

- b. 使用您常用的文本编辑器打开文件，并添加以下脚本。保存并关闭文件。

```
#!/bin/sh
# chkconfig: 35 99 01
# description: The Reprise License Manager daemon.
# processname: dcv-rlm

### BEGIN INIT INFO
# Provides: dcv-rlm
# Required-Start: $local_fs $remote_fs $syslog
# Required-Stop: $local_fs $remote_fs $syslog
# Default-Start: 3 4 5
# Default-Stop: 0 1 2 6
# Short-Description: The Reprise License Manager daemon.
# Description: A service that runs the Reprise License Manager daemon.
### END INIT INFO

# user used to run the daemon
RLM_USER="rlm"

# root of rlm installation
RLM_ROOT="/opt/nice/rlm"

# license directory (license files should have .lic extension)
RLM_LICENSE_DIR="/opt/nice/rlm/license"

# log file
RLM_LOG_FILE="/var/log/rlm.log"
```

```
_getpid() {
    pidof -o $$ -o $PPID -o %PPID -x "$1"
}

start() {
    echo -n "Starting rlm: "
    touch ${RLM_LOG_FILE}
    chown "${RLM_USER}" ${RLM_LOG_FILE}
    su -p -s /bin/sh "${RLM_USER}" -c "${RLM_ROOT}/rlm -c ${RLM_LICENSE_DIR} \
        -nows -dlog +${RLM_LOG_FILE} &"
    if [ $? -ne 0 ]; then
        echo "FAILED"
        return 1
    fi
    echo "OK"
}

stop() {
    echo -n "Stopping rlm: "
    pid=`_getpid ${RLM_ROOT}/rlm`
    if [ -n "$pid" ]; then
        kill $pid >/dev/null 2>&1
        sleep 3
        if [ -d "/proc/$pid" ] ; then
            echo "FAILED"
            return 1
        fi
    fi
    echo "OK"
}

status() {
    pid=`_getpid ${RLM_ROOT}/rlm`
    if [ -z "$pid" ]; then
        echo "rlm is stopped"
        return 3
    fi
    echo "rlm (pid $pid) is running..."
    return 0
}

restart() {
    stop
    start
}
```

```
}

case "$1" in
  start)
    start
    ;;
  stop)
    stop
    ;;
  status)
    status
    ;;
  restart)
    restart
    ;;
  *)
    echo $"Usage: $0 {start|stop|status|restart}"
    exit 1
esac

exit $?

# ex:ts=4:et:
```

- c. 将脚本编写为可执行文件，将其复制到 `/etc/init.d/`，然后将其添加到 `chkconfig` 实用程序：

```
chmod +x /opt/nice/rlm/dcv-rlm
```

```
cp -a /opt/nice/rlm/dcv-rlm /etc/init.d/
```

```
chkconfig --add dcv-rlm
```

4. 启动 RLM 服务器：

```
$ service dcv-rlm start
```

5. 确保 RLM 服务器正在按预期运行和工作。使用您常用的文本编辑器打开 `var/log/rlm.log`，并确认显示以下行：

```
date_time (nice) Server started on license1 (hostid: host_id) for: dcv dcv-gl
```

Note

rlm.log 文件的内容可能会随 RLM 服务器版本而异。

步骤 6：配置 NICE DCV 服务器

配置您的 NICE DCV 服务器以使用 RLM 服务器。为此，您必须在 NICE DCV 服务器上配置 `license-file` 配置参数。

必须根据要连接的 RLM 服务器的规范设置 `license-file` 参数，格式为 `RLM_server_port@RLM_server`。可以将 RLM 服务器指定为主机名或 IP 地址。如果未明确配置 RLM 服务器端口，则默认为 5053。

如果正在使用多个 RLM 服务器，您可以指定一个包含多个 RLM 服务器规范的列表，以：(在 Linux 上) 或；(在 Windows 上) 分隔。然后，服务器依次尝试连接到每个服务器，直到可以与相应的 RLM 服务器建立连接。例如，如果无法访问主 RLM 服务器，在使用 RLM 失效转移服务器接替该服务器时，这可能是特别有用的。在这种情况下，您可以按以下格式指定许可证：`RLM_primary_server_port@RLM_primary_server:RLM_failover_server_port@RLM_failover_server`

Note

如果 NICE DCV 服务器安装在 Windows 上，您需要使用；分隔规范中的条目。

主题

- [Windows NICE DCV 服务器配置](#)
- [Linux NICE DCV 服务器配置](#)

Windows NICE DCV 服务器配置

配置 Windows 服务器上的 `license-file` 配置参数

1. 打开 Windows 注册表编辑器。
2. 导航到 `HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/license/` 注册表项，并选择 `license-file` 参数。

如果该注册表项中没有 `license-file` 参数，您必须创建它：

- a. 在左侧面板中打开 license 注册表项的上下文（右键单击）菜单，然后依次选择新建和字符串值。
 - b. 对于名称，输入 license-file 并按 Enter。
3. 打开 license-file 参数。对于值数据，请输入 *RLM_server_port@RLM_server* 格式的 RLM 服务器的端口号和主机名。如果您需要设置到多个 RLM 服务器的连接，请查看上面的注释。
 4. 选择确定，并关闭 Windows 注册表编辑器。

Linux NICE DCV 服务器配置

配置 Linux 服务器上的 **license-file** 配置参数

1. 导航到 /etc/dcv/，并使用您常用的文本编辑器打开 dcv.conf。
2. 在 [license] 部分中找到 license-file 参数。然后，将现有路径替换为 RLM 服务器的端口和主机名，格式为 *RLM_server_port@RLM_server*。

如果 license-file 部分中没有 [license] 参数，则使用以下格式手动添加此参数：

```
license-file = "RLM_server_port@RLM_server"
```

如果您需要设置到多个 RLM 服务器的连接，请查看上面的注释。

3. 保存并关闭文件。

更新生产许可证

NICE DCV 服务器每隔几分钟检查一次 RLM 服务器上的许可证。如果在 RLM 服务器上更新许可证，NICE DCV 服务器自动更新运行的会话使用的许可证。以下过程详细介绍了如何在 RLM 上更新 DCV 许可证。

在 RLM 服务器上更新 DCV 许可证

1. 更新以前[安装](#)的许可证文件。在 Linux 上，该文件应放在 /opt/dcv/rlm/license/license.lic 中；在 Windows 上，该文件应放在 C:\RLM\license\license.lic 中。
2. 运行 rlmutil rlmreread 命令以强制重新加载许可证文件。

在 RLM 服务器上更新许可证后，NICE DCV 服务器应在几分钟（通常为 5 分钟或更短时间）内检查新许可证的使用情况。

从 NICE DCV 版本 2021.0 开始，您可以作为管理员使用以下命令强制立即更新许可证：

```
$ dcv reload-licenses
```

升级 NICE DCV 服务器

以下主题介绍了如何升级 NICE DCV 服务器。

内容

- [兼容性注意事项](#)
- [在 Windows 上升级 NICE DCV 服务器](#)
- [在 Linux 上升级 NICE DCV 服务器](#)

兼容性注意事项

NICE DCV Server 2017 和更高版本与 NICE DCV Client 2017 和更高版本兼容。

Note

有关本地服务器和不基于 EC2 的服务器的 NICE DCV 服务器许可兼容性要求的信息，请参阅[许可要求](#)。

在 Windows 上升级 NICE DCV 服务器

在 Windows 上升级 NICE DCV 服务器

1. 通过使用 RDP 客户端，以管理员身份连接到 NICE DCV 服务器。
2. 确保不存在正在运行的 NICE DCV 会话。使用 `dcv list-sessions NICE DCV` 命令检查是否有任何运行的会话。如果有运行的会话，请使用 `dcv close-session NICE DCV` 命令将其停止。
3. 在确认没有运行的会话后，停止 NICE DCV 服务器。有关更多信息，请参阅[停止 NICE DCV 服务器](#)。

4. 对 NICE DCV 服务器配置进行备份。打开注册表编辑器，导航到 HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv，右键单击 dcv 注册表项，然后选择导出。
5. 从 [NICE](#) 网站中下载最新版本的 NICE DCV 服务器。
6. 执行[使用向导](#)中所述的步骤，从步骤 3 开始。
7. 在安装完成后，确认 NICE DCV 服务器配置仍然正确。打开注册表编辑器，导航到 HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv，并将参数与您在步骤 4 中导出的配置进行比较。
8. 启动新的 NICE DCV 会话以测试 NICE DCV 服务器。有关更多信息，请参阅 [启动 NICE DCV 会话](#)。

在 Linux 上升级 NICE DCV 服务器

在 Linux 上升级 NICE DCV 服务器

1. 使用 SSH 通过 root 用户登录到服务器。
2. 确保不存在正在运行的 NICE DCV 会话。使用 `dcv list-sessions NICE DCV` 命令检查是否有任何运行的会话。如果有运行的会话，请使用 `dcv close session NICE DCV` 命令将其停止。
3. 在确认没有运行的会话后，停止 NICE DCV 服务器。有关更多信息，请参阅 [停止 NICE DCV 服务器](#)。
4. 对 NICE DCV 服务器配置进行备份。将 `/etc/dcv/dcv.conf` 文件复制到安全位置。
5. 执行[安装 NICE DCV 服务器](#)中所述的步骤。
6. 在安装完成后，确认 NICE DCV 服务器配置仍然正确。打开您在步骤 4 中复制的文件，并将该文件与 `/etc/dcv/dcv.conf` 文件进行比较。
7. 启动新的 NICE DCV 会话以测试 NICE DCV 服务器。有关更多信息，请参阅 [启动 NICE DCV 会话](#)。

卸载 NICE DCV 服务器

以下主题介绍了如何卸载 NICE DCV 服务器。

内容

- [在 Windows 上卸载 NICE DCV 服务器](#)

- [在 Linux 上卸载 NICE DCV 服务器](#)

在 Windows 上卸载 NICE DCV 服务器

在 Windows 上卸载 NICE DCV 服务器

1. 通过使用 RDP 客户端，以管理员身份连接到 NICE DCV 服务器。
2. 确保不存在正在运行的 NICE DCV 会话。使用 `dcv list-sessions NICE DCV` 命令检查是否有任何运行的会话。如果有运行的会话，请使用 `dcv close session NICE DCV` 命令将其停止。
3. 在确认没有运行的会话后，停止 NICE DCV 服务器。有关更多信息，请参阅 [停止 NICE DCV 服务器](#)。
4. 打开 Windows 设置应用程序，并导航到应用程序和功能面板。
5. 选择 NICE DCV 服务器，然后按卸载。
6. （可选）您可能还希望删除 NICE DCV 服务器生成的任何日志文件。卸载完成后，导航到 `C:\NICEProgramData\dcv\` 并删除日志文件夹。

在 Linux 上卸载 NICE DCV 服务器

NICE DCV 服务器是使用一系列 RPM 或 .deb 软件包安装的，具体取决于主机服务器的操作系统。

Note

您必须以根用户身份登录才能卸载 NICE DCV 服务器。

在 Linux 上卸载 NICE DCV 服务器

1. 确保不存在正在运行的 NICE DCV 会话。使用 `dcv list-sessions NICE DCV` 命令检查是否有任何运行的会话。如果有运行的会话，请使用 `dcv close session NICE DCV` 命令将其停止。
2. 在确认没有运行的会话后，停止 NICE DCV 服务器。有关更多信息，请参阅 [停止 NICE DCV 服务器](#)。
3. 卸载 NICE DCV 服务器软件包。根据您的执行安装的方式，可能未在您的系统上安装某些软件包，可以从命令中省略这些软件包。有关可选的软件包列表，请参阅 [在 Linux 上安装 NICE DCV 服务器](#)。

Amazon Linux 2 and RHEL, CentOS

```
$ sudo yum remove nice-dcv-server nice-xdcv nice-dcv-gl nice-dcv-gltest nice-dcv-simple-external-authenticator
```

SLES 12.x/15.x

```
$ sudo zypper remove nice-dcv-server nice-xdcv nice-dcv-gl nice-dcv-gltest nice-dcv-simple-external-authenticator
```

Ubuntu 22.04

```
$ sudo apt remove nice-dcv-server nice-xdcv nice-dcv-gl nice-dcv-gltest nice-dcv-simple-external-authenticator
```

4. (可选) 您可能还希望删除 NICE DCV 服务器生成的任何日志文件。在卸载完成后，导航到 `/var/log` 并删除 `dcv` 文件夹。

成像 NICE DCV 服务器

自定义 [Amazon EC2](#) 实例后，您可以将这些更改捕获为 [亚马逊系统映像](#) (AMI)。此功能允许您在需要从单个 AMI 启动多个实例，所有实例的配置都相同。如果您需要使用高性能远程显示协议安全地进行流式传输，则可以在拍摄 Amazon EC2 实例的映像之前将 NICE DCV 添加到操作系统中。NICE DCV 配置包含在您的映像中，允许您在映像级别分离业务部门或在已部署的实例上设置特定的 DCV 配置。

例如，如果您要从单个 AMI 部署多个 Amazon EC2 实例，则可以为本地用户账户使用自动创建控制台，并将 NICE DCV 权限委派给最终用户。或者，您也可以使用代理，例如 [NICE DCV 会话管理器](#)，来大规模管理 NICE DCV 会话的创建。


可以通过以下两种方式之一来创建 NICE DCV AMI：

构建 NICE DCV 镜像

首先，您的系统上必须安装 NICE DCV。如果不支持，请确保 [NICE DCV 支持](#) 您的系统，然后按照 [安装说明](#) 进行操作。安装并 [配置](#) NICE DCV 后，获取该实例的 [AMI](#)。

或者，如果你满足 [Windows](#) 或 [Linux](#) 的 NICE DCV 先决条件，你可以运行亚马逊管理的 Image Builder NICE DCV 组件来安装和配置 NICE DCV。可以通过执行以下操作来检索组件：

1. 在 [Amazon EC2 Image Builder 控制台](#) 中导航到组件页面。
2. 选择筛选条件所有者下拉菜单，然后选择快速入门（亚马逊管理）。
3. 使用筛选器文本框搜索 `dcv-server-windows` 或 `dcv-server-linux`。
4. 选择组件的超链接。
5. 在 NICE DCV 组件页面上，从“内容”部分检索组件内容。
6. 使用 [T AWS ask Orchestrator 和 Executor \(AWSTOE\)](#) 在实例上本地运行组件。


 Note

有关更多信息，请参阅 [入门 AWSTOE](#)。

有关组件中参数的用法，请参阅以下部分。

将 NICE DCV 添加到图像管道中

[EC2 Image Builder 配方](#) 定义了作为创建新映像的起点的基础映像，以及为自定义映像和验证一切是否按预期运行而添加的一组组件。在此配方中，选择 `dcv-server-windows` 或 `dcv-server-linux` 组件以在管道中自动安装 NICE DCV。选择其中一个组件时，您可以微调参数以满足您的要求。

 Note

对于 Linux，需要满足所有 [先决条件](#)。这可以在基本 AMI 上完成，也可以在之前的 Image Builder 组件中完成。

参数

Windows

- `sessionOwner`— 设置自动创建的会话的默认所有者。如果未指定，则将禁用自动创建控制台。有关更多信息，请参阅《NICE DCV 管理指南》中的 [“启用自动控制台会话”](#)。
- `dcvPermissions`— 设置会话的 NICE DCV 权限。有关更多信息，请参阅《DCV 管理指南》中的 [使用权限文件](#)。

Linux

- `SessionOwner`— 设置自动创建的会话的默认所有者。如果未指定，则将禁用自动创建控制台。有关更多信息，请参阅《NICE DCV 管理指南》中的[“启用自动控制台会话”](#)。
- `Packages`— 定义将要安装的 NICE DCV 软件包。如果为空，则安装所有可用的 NICE DCV 软件包。有关更多信息，请参阅《[NICE DCV 管理指南](#)》中的[“在 Linux 上安装 NICE DCV 服务器”](#)。

如果要修改组件，可以[创建一个新的组件版本](#)。

管理 NICE DCV 服务器

您必须以管理员（Windows）或根用户（Linux）身份登录，才能启动、停止或配置 NICE DCV 服务器。

主题

- [启动 NICE DCV 服务器](#)
- [停止 NICE DCV 服务器](#)
- [启用 QUIC UDP 传输协议](#)
- [更改 NICE DCV 服务器 TCP/UDP 端口和侦听地址](#)
- [管理 TLS 证书](#)
- [断开连接空闲的客户端](#)
- [在 Linux NICE DCV 服务器上启用 GPU 共享](#)
- [启用触摸屏和触控笔支持](#)
- [启用游戏手柄支持](#)
- [启用 USB 远程控制](#)
- [配置智能卡缓存](#)
- [配置 WebAuthn 重定向](#)
- [启用会话存储](#)
- [在 Linux NICE DCV 服务器上配置打印机](#)
- [在 Linux NICE DCV 服务器上配置剪贴板](#)
- [配置多声道音频](#)
- [配置 HTTP 标头](#)
- [配置 NICE DCV 身份验证](#)
- [配置 NICE DCV 授权](#)

启动 NICE DCV 服务器

NICE DCV 服务器必须正在运行才能托管会话。

默认情况下，只要托管 NICE DCV 服务器的服务器启动，NICE DCV 服务器就会启动。如果您在安装 NICE DCV 服务器时选择禁用自动启动，您必须手动启动服务器或再次设置自动启动。要采用任一选项，请执行以下过程之一。

Windows NICE DCV server

使用 Microsoft 管理控制台的“服务”管理单元手动启动 NICE DCV 服务器。

在 Windows 上启动 NICE DCV 服务器

1. 打开 Microsoft 管理控制台的“服务”管理单元。
2. 在右侧窗格中，打开 DCV 服务器。
3. 选择启动。

Note

如果服务器已启动并正在运行，则会禁用启动按钮。

使用 Microsoft 管理控制台的“服务”管理单元配置自动启动。

在 Windows 上将 NICE DCV 服务器配置为自动启动

1. 打开 Microsoft 管理控制台的“服务”管理单元。
2. 在右侧窗格中，打开 DCV 服务器。
3. 对于启动服务，选择自动。

Linux NICE DCV server

使用命令行手动启动 NICE DCV 服务器。

在 Linux 上启动 NICE DCV 服务器

使用以下命令：

- RHEL、CentOS、SUSE Linux Enterprise 12 和 Ubuntu 18.x

```
$ sudo systemctl start dcvserver
```

使用命令行将 NICE DCV 服务器配置为自动启动。

在 Linux 上将 NICE DCV 服务器配置为自动启动

使用以下命令：

- RHEL、CentOS、SUSE Linux Enterprise 12 和 Ubuntu 18.x

```
$ sudo systemctl enable dcvserver
```

停止 NICE DCV 服务器

您可以随时停止 NICE DCV 服务器。停止服务器将终止所有活动的 NICE DCV 会话。在重新启动服务器之前，您无法启动新的会话。

Windows NICE DCV server

使用 Microsoft 管理控制台的“服务”管理单元手动停止 NICE DCV 服务器。

在 Windows 上停止 NICE DCV 服务器

1. 打开 Microsoft 管理控制台的“服务”管理单元。
2. 在右侧窗格中，打开 DCV 服务器。
3. 选择停止。

Note

如果服务器已停止，则停止按钮处于禁用状态。

使用 Microsoft 管理控制台的“服务”管理单元禁用自动启动。

在 Windows 上禁止自动启动 NICE DCV 服务器

1. 打开 Microsoft 管理控制台的“服务”管理单元。
2. 在右侧窗格中，打开 DCV 服务器。
3. 对于启动服务，选择手动。

Linux NICE DCV server

使用命令行停止 NICE DCV 服务器。

在 Linux 上停止 NICE DCV 服务器

使用以下命令：

```
$ sudo systemctl stop dcvserver
```

使用命令行禁用 NICE DCV 服务器自动启动。

在 Linux 上禁止自动启动 NICE DCV 服务器

使用以下命令：

```
$ sudo systemctl disable dcvserver
```

启用 QUIC UDP 传输协议

默认情况下，NICE DCV 使用基于 TCP 的 WebSocket 协议进行数据传输。

您可以将 NICE DCV 配置为使用 QUIC 协议传输数据。该传输协议基于 UDP。如果您的网络出现较高的延迟和丢失数据包，使用 QUIC 可能会提高性能。如果启用 QUIC，NICE DCV 服务器将使用 QUIC 协议传输数据。但是，它继续 WebSocket 用于身份验证流量。

Note

只有在您的网络和安全配置允许 UDP 流量时，您才能使用 QUIC。


如果启用 QUIC，客户端可以在连接到 NICE DCV 服务器会话时使用 QUIC 协议传输数据。如果客户端在连接时不使用 QUIC 协议，则会使用 WebSocket。有关 QUIC 协议的更多信息，请参阅《NICE DCV 用户指南》中的[连接到 NICE DCV 会话](#)。

Windows NICE DCV server

将 NICE DCV 配置为使用 QUIC (UDP) 传输数据

1. 打开 Windows 注册表编辑器，然后导航到 HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/connectivity/ 项。

2. 打开 `enable-quick-frontend` 参数。对于值数据，输入 1。

 Note

如果找不到该参数，请创建一个新的 DWORD (32 位) 参数并将其命名为 `enable-quick-frontend`。

3. 关闭 Windows 注册表编辑器。
4. [停止并重新启动](#) NICE DCV 服务器。

Linux NICE DCV server

将 NICE DCV 配置为使用 QUIC (UDP) 传输数据

1. 使用首选文本编辑器打开 `/etc/dcv/dcv.conf`。
2. 请在 [connectivity] 部分执行以下操作：
 - 对于 `enable-quick-frontend`，请指定 `true`。

```
[connectivity]
enable-quick-frontend=true
```

3. 保存并关闭文件。
4. [停止并重新启动](#) NICE DCV 服务器。

更改 NICE DCV 服务器 TCP/UDP 端口和侦听地址

默认情况下，NICE DCV 服务器配置为侦听 TCP 端口 8443，并在运行该服务器的主机中的任何网络接口上进行通信。

您可以在安装 NICE DCV 服务器后指定自定义 TCP 端口。如果将 NICE DCV 服务器配置为[启用 QUIC](#)，您也可以为 QUIC 流量指定自定义 UDP 端口。端口号必须大于 1024。

您可以指定 NICE DCV 服务器侦听的网络地址。例如，这允许您指定是否仅使用 IPv4 或 IPv6。这还允许您将服务器绑定到特定的网络接口，并确保流量流经特定的网络。

Important

每次您对 NICE DCV 服务器的网络配置进行更改时，请确保向客户端通知更改，例如，客户端需要知道用于连接到会话的端口号。

Tip

另一种控制向客户端公开的网络地址和端口的方法是，将 [NICE DCV Connection Gateway](#) 或其他 Web 代理或负载均衡器作为服务器的前端。在通过网关访问您的 NICE DCV 服务器主机时，您可以将单个地址用于服务器。这还允许使用低于 1024 的端口号，包括 443（用于 HTTPS 的标准端口号）。

有关配置其网络地址和端口的更多信息，请参阅网关文档。

主题

- [更改 NICE DCV 服务器 TCP/UDP 端口](#)
- [侦听特定的终端节点](#)

更改 NICE DCV 服务器 TCP/UDP 端口

Windows NICE DCV server

要更改 NICE DCV 服务器使用的端口，请使用 Windows 注册表编辑器配置 `web-port` 和 `quic-port` 参数。


在 Windows 上更改服务器的端口

1. 打开 Windows 注册表编辑器。
2. 导航到 `HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/connectivity/` 注册表项。
3. 要配置 TCP 端口，请选择 `web-port` 参数。

如果在注册表项中没有 `web-port` 参数，请创建该参数：

- a. 在导航窗格中，打开 `connectivity` 项的上下文（右键单击）菜单。然后，选择新建 > DWORD (32 位) 值。

- b. 对于名称，输入 `web-port` 并按 Enter。
4. 打开 `web-port` 参数。对于值数据，输入新的 TCP 端口号。如果未配置该参数，则 NICE DCV 服务器默认使用 TCP 端口 8443。


 Note

TCP 端口号必须高于 1024。

5. 如果启用了 QUIC，要配置 UDP 端口，请选择 `quic-port` 参数。

如果在注册表项中没有 `quic-port` 参数，请创建该参数：

- a. 在导航窗格中，打开 `connectivity` 项的上下文（右键单击）菜单。然后，选择新建 > DWORD（32 位）值。
 - b. 对于名称，输入 `quic-port` 并按 Enter。
6. 打开 `quic-port` 参数。对于值数据，输入新的 UDP 端口号。如果未配置该参数并启用了 QUIC 支持，则 NICE DCV 服务器默认使用 UDP 端口 8443。

 Note

UDP 端口号必须大于 1024。

7. 选择确定，并关闭 Windows 注册表编辑器。
8. [停止并重新启动](#) NICE DCV 服务器。

Linux NICE DCV server

要更改 NICE DCV 服务器使用的端口，请在 `dcv.conf` 文件中配置 `web-port` 和 `quic-port` 参数。

在 Linux 上更改服务器的端口

1. 导航到 `/etc/dcv/`，并使用您常用的文本编辑器打开 `dcv.conf`。
2. 在 `[connectivity]` 部分中找到 `web-port` 参数。然后，将现有的 TCP 端口号替换为新的 TCP 端口号。

如果在 `[connectivity]` 部分中没有 `web-port` 参数，请按以下格式手动添加该参数：

```
[connectivity]
web-port=port_number
```

Note

TCP 端口号必须为 1024 或更高。

3. 在 [connectivity] 部分中找到 quic-port 参数。然后，将现有的 UDP 端口号替换为新的 UDP 端口号。

如果在 [connectivity] 部分中没有 web-port 参数，请按以下格式手动添加该参数：

```
[connectivity]
quic-port=port_number
```

Note

UDP 端口号必须为 1024 或更高。

4. 保存并关闭文件。
5. [停止并重新启动](#) NICE DCV 服务器。

侦听特定的终端节点

要仅侦听特定的网络地址，您可以在 NICE DCV 服务器配置中设置 web-listen-endpoints 和 quic-listen-endpoints 参数。

每个终端节点由 IPv4 或 IPv6 地址表示，后跟可选的端口号并以 : 分隔。在终端节点中指定的端口号优先于 web-port 和 quic-port 参数中指定的端口。

由于可以指定多个终端节点，因此，一组终端节点由以逗号分隔的列表表示并用方括号括起来，其中的每个终端节点用单引号引起来。例如，['0.0.0.0:8443', ':::8443'] 表示端口 8443 上的任何本地 IPv4 地址和任何本地 IPv6 地址，['::%1]:8443' 表示绑定到 Windows 主机上索引为 1 的网络接口的 IPv6 地址，['::%eth1]:8443' 表示绑定到 Linux 主机上的 eth1 网络接口的 IPv6 地址。

Note

这些配置参数仅从 NICE DCV Server 2022.0 开始提供。

Windows NICE DCV server

在 Windows 上更改服务器的终端节点

1. 打开 Windows 注册表编辑器。
2. 导航到 HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/connectivity/ 注册表项。
3. 要配置 TCP 终端节点，请选择 web-listen-endpoints 参数。

如果在注册表项中没有 web-listen-endpoints 参数，请创建该参数：

- a. 在导航窗格中，打开 connectivity 项的上下文（右键单击）菜单。然后，选择新建 > 字符串值。
 - b. 对于名称，输入 web-listen-endpoints 并按 Enter。
4. 打开 web-listen-endpoints 参数。对于值数据，输入终端节点列表。
 5. 如果启用了 QUIC，要配置 UDP 终端节点，请选择 quic-listen-endpoints 参数。

如果在注册表项中没有 quic-listen-endpoints 参数，请创建该参数：

- a. 在导航窗格中，打开 connectivity 项的上下文（右键单击）菜单。然后，选择新建 > 字符串值。
 - b. 对于名称，输入 quic-listen-endpoints 并按 Enter。
6. 打开 quic-listen-endpoints 参数。对于值数据，输入终端节点列表。
 7. 选择确定，并关闭 Windows 注册表编辑器。
 8. [停止并重新启动](#) NICE DCV 服务器。

Linux NICE DCV server

在 Linux 上更改服务器的终端节点

1. 导航到 /etc/dcv/，并使用您常用的文本编辑器打开 dcv.conf。

2. 在 [connectivity] 部分中找到 web-listen-endpoints 参数。然后，替换现有的终端节点列表。

如果在 [connectivity] 部分中没有 web-listen-endpoints 参数，请按以下格式手动添加该参数：

```
[connectivity]
web-listen-endpoints=[endpoint1, endpoint2]
```

3. 在 [connectivity] 部分中找到 quic-listen-endpoints 参数。然后，替换现有的终端节点列表。

如果在 [connectivity] 部分中没有 quic-listen-endpoints 参数，请按以下格式手动添加该参数：

```
[connectivity]
quic-listen-endpoints=[endpoint1, endpoint2]
```

4. 保存并关闭文件。
5. [停止并重新启动](#) NICE DCV 服务器。

管理 TLS 证书

NICE DCV 自动生成自签名证书，用于保护 NICE DCV 客户端和 NICE DCV 服务器之间的流量。默认情况下，如果未安装其他证书，则使用该证书。默认证书包含两个文件。它们是证书本身 (dcv.pem) 和密钥 (dcv.key)。有关详细信息，请参阅[the section called “自签名证书的重定向说明”](#)。

在 DCV 客户端用户连接到服务器时，他们可能会收到服务器证书警告，他们可以在建立连接之前验证服务器证书。

如果他们使用 Web 浏览器进行连接，则浏览器可能会警告客户端用户不要轻易信任服务器的证书，并提醒他们应与管理员联系以确认证书的真实性。

同样，如果他们使用 Windows、Linux 或 macOS 客户端，可能会建议他们向 NICE DCV 服务器管理员确认给定证书的指纹。

要验证其证书指纹的真实性，请运行 `dcv list-endpoints -j` 并对照其证书指纹检查输出。

您可以将默认 NICE DCV 证书及其密钥替换为您自己的证书和密钥。

在您生成自己的证书时，请选择满足您的特定需求的证书属性。大多数情况下，CN（Common Name）属性必须与主机的公有主机名匹配。您可能还希望指定 SAN（Subject Alternative Name）属性，并将其设置为主机的 IP 地址。

有关如何生成证书的说明，请参阅特定证书颁发机构的文档。

Important

如果您使用自己的证书和密钥，您必须将证书命名为 `dcv.pem`，并将密钥命名为 `dcv.key`。

Windows NICE DCV server

在 Windows 上更改服务器的 TLS 证书

- 将证书及其密钥放置在 Windows NICE DCV 服务器上的以下位置中：

```
C:\Windows\System32\config\systemprofile\AppData\Local\NICE\dcv\
```

Linux NICE DCV server

在 Linux 上更改服务器的 TLS 证书

1. 将证书及其密钥放置在 Linux NICE DCV 服务器上的以下位置中：

```
/etc/dcv/
```

2. 将这两个文件的所有权授予 `dcv` 用户，并将其权限更改为 `600`（只有所有者可以读取或写入它们）。

```
$ sudo chown dcv dcv.pem dcv.key
```

```
$ sudo chmod 600 dcv.pem dcv.key
```

Note

从 NICE DCV 2022.0 开始，如果您在 NICE DCV 服务器运行时更新证书文件，将自动重新加载新证书。对于以前版本的 NICE DCV，您需要手动[停止](#)并[重新启动](#) NICE DCV 服务器。

断开连接空闲的客户端

您可以配置 NICE DCV 以断开连接空闲的客户端。更具体地说，您可以为在特定时间段内未向 NICE DCV 服务器发送任何键盘或指针输入的客户端执行该操作。默认情况下，在 NICE DCV 客户端处于空闲状态 60 分钟（1 小时）后，NICE DCV 服务器将其断开连接。

某些操作会重置空闲断开连接超时时间。如果发生以下任何操作，则空闲超时时间将重置为其设定的时间范围：

- 移动鼠标
- 按下鼠标按钮或移动鼠标滚轮
- 按下键盘上的任意键
- 触摸触摸屏（如果启用）
- 使用触控笔（如果启用）
- 使用游戏手柄（如果启用）
- 使用网络摄像头进行流式传输（如果启用）
- 任何文件存储操作，例如上传文件、创建目录、下载文件或列出项目

Note

连接和使用任何音频设备都不会重置空闲超时时间。

您也可以将 NICE DCV 服务器配置为向空闲客户端发送通知。该通知旨在通知客户端，即将断开连接客户端的会话。仅 NICE DCV 服务器和客户端 2017.4 和更高版本支持超时通知。

您可以使用以下过程来指定自定义空闲超时期限。

Windows NICE DCV server

要更改 NICE DCV 服务器的空闲超时期限，您必须使用 Windows 注册表编辑器配置 `idle-timeout` 参数。

在 Windows 上更改空闲超时期限

1. 打开 Windows 注册表编辑器。
2. 导航到 `HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/connectivity/` 注册表项，并选择 `idle-timeout` 参数。

如果找不到该参数，请使用以下步骤创建该参数：

- a. 在导航窗格中，打开 `connectivity` 项的上下文（右键单击）菜单。然后，选择新建 > `DWORD (32 位) 值`。
 - b. 对于名称，输入 `idle-timeout` 并按 Enter。
3. 打开 `idle-timeout` 参数。对于值数据，输入空闲超时期限的值（以分钟为单位，十进制）。要避免断开空闲客户端的连接，请输入 `0`。
 4. 选择确定，并关闭 Windows 注册表编辑器。

（可选）将 NICE DCV 服务器配置为向空闲客户端发送超时通知

1. 导航到 `HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/connectivity/` 注册表项，并选择 `idle-timeout-warning` 参数。

如果找不到该参数，请使用以下步骤创建该参数：

- a. 在导航窗格中，打开 `connectivity` 项的上下文（右键单击）菜单。然后，选择新建 > `DWORD (32 位) 值`。
 - b. 对于名称，输入 `idle-timeout-warning` 并按 Enter。
2. 打开 `idle-timeout-warning` 参数。对于值数据，输入断开连接之前发送关联警告通知的秒数（十进制）。例如，如果您希望在达到空闲超时前两分钟发送通知，请输入 `120`。
 3. 选择确定，并关闭 Windows 注册表编辑器。

Linux NICE DCV server

要更改 NICE DCV 服务器的空闲超时期限，您必须在 `dcv.conf` 文件中配置 `idle-timeout` 参数。

在 Linux 上更改空闲超时期限

1. 使用首选文本编辑器打开 `/etc/dcv/dcv.conf`。
2. 在 `[connectivity]` 部分中找到 `idle-timeout` 参数。然后，将现有的超时期限替换为新的超时期限（以分钟为单位，十进制）。

如果在 `[connectivity]` 部分中没有 `idle-timeout` 参数，请按以下格式手动添加该参数：

```
[connectivity]
idle-timeout=timeout_in_minutes
```

要避免断开空闲客户端的连接，请输入 `0`。

3. 保存并关闭文件。

（可选）将 NICE DCV 服务器配置为向空闲客户端发送超时通知

1. 使用首选文本编辑器打开 `/etc/dcv/dcv.conf`。
2. 将 `idle-timeout-warning` 参数添加到 `[connectivity]` 部分中，并指定断开连接之前发送关联警告通知的秒数（十进制）。

```
idle-timeout-warning=seconds_before_idle_timeout
```

例如，如果您希望在达到空闲超时前的两分钟发送通知，请指定 `120`。

3. 保存并关闭文件。

在 Linux NICE DCV 服务器上启用 GPU 共享

通过使用 GPU 共享，您可以在多个 NICE DCV 虚拟会话之间共享一个或多个物理 GPU。有关会话的更多信息，请参阅[管理 NICE DCV 会话](#)。通过使用 GPU 共享，您可以使用单个 NICE DCV 服务器，并托管多个共享该服务器的物理 GPU 资源的虚拟会话。

Note

GPU 共享仅在 Linux NICE DCV 服务器上受支持。

先决条件

在开始之前，请满足以下先决条件：

- 在 Linux 服务器上安装 NICE DCV 服务器。
- 在服务器上安装 NICE DCV `dcv-g1` 和 `nice-Xdcv` 软件包。
- 确保服务器至少具有一个支持的 NVIDIA GPU。
- 在服务器上安装 NVIDIA GPU 驱动程序。需要具有正式的 NVIDIA 驱动程序。不支持开源 NVIDIA 驱动程序。
- 确保 NVIDIA GPU 驱动程序支持硬件加速的 OpenGL。
- 安装 X 服务器，并配置 `xorg.conf` 文件中的 `Device` 和 `Screen` 部分。

Note

您可以使用 `nvidia-xconfig` NVIDIA 实用程序自动创建 `xorg.conf` 文件，并为所有可用的 NVIDIA GPU 配置该文件。

- 确保 X 服务器正在运行。
- (可选) 运行 `dcvgldiag` 工具以验证 NICE DCV 服务器配置。有关更多信息，请参阅 [安装后检查](#)。

您也可以安装 `nice-dcv-gltest` 软件包，并运行 `dcvgltest` 测试应用程序以检查是否为服务器正确配置了 GPU 共享。

要启用 GPU 共享，您必须指定要由虚拟会话使用的 GPU 的列表。如果未指定 GPU，则仅使用标准 X 服务器使用的 GPU (显示名称为 `:0.0`)。

在满足本主题前面所述的先决条件后，在 `dcv.conf` 文件的 `gl-displays` 参数中指定 GPU。

在 Linux NICE DCV 服务器上启用 GPU 共享

1. 导航到 `/etc/dcv/`，并使用您的首选文本编辑器打开 `dcv.conf` 文件。
2. 添加 `[display/linux]` 部分和 `gl-displays` 参数。然后，按以下格式指定可用的 GPU：

```
[display/linux]
gl-displays =
  [':xserver_port.screen_number_1', ':xserver_port.screen_number_2', ...]
```

其中 `xserver_port` 是服务器，`screen_number` 是与 GPU 相关的屏幕的关联编号。`screen_number` 从 0 开始。

以下示例显示了在默认 X 服务器会话上运行的两个 GPU 的 `gl-displays` 参数：

```
[display/linux]
gl-displays = [':0.0', ':0.1']
```

3. 保存并关闭文件。
4. [停止并重新启动](#) NICE DCV 服务器。

启用触摸屏和触控笔支持

Note

触摸屏和触控笔设备不需要使用 USB 重定向。此外，不需要在 NICE DCV 服务器上安装任何供应商驱动程序。

NICE DCV 使用本机操作系统 API 支持触摸屏和触控笔。

Windows 使用 Windows Ink。

Linux 使用 X11 输入注入。

• Windows 服务器支持

在所有支持的 Windows 操作系统上支持触摸屏。从 Windows 10 和 Windows 2019 开始，在所有支持的 Windows 操作系统上支持触控笔，但在 Windows 2016、Windows 8.1 和更早版本上不支持触控笔。默认情况下，在 Windows NICE DCV 服务器上启用了这些功能。无需其他配置。

• Linux 服务器支持

在所有支持的 Linux 操作系统上支持触摸屏和触控笔。默认情况下，已在 Linux NICE DCV 服务器上托管的虚拟会话上启用这些功能。但是，需要一些额外配置才能在 Linux NICE DCV 服务器上托管的控制台会话上启用这些功能。

⚠ Important

如果在客户端和服务器上均支持触摸屏和触控笔功能，并且在服务器上启用了该功能，则允许在 NICE DCV 中使用触摸屏和触控笔。有关客户端支持的信息，请参阅《NICE DCV 用户指南》中的[客户端功能](#)。

为 Linux NICE DCV 服务器上托管的控制台会话启用触摸屏和触控笔支持

1. 使用所需的文本编辑器打开 `/etc/X11/xorg.conf`。
2. 将以下部分添加到此文件中。

```
Section "InputDevice"
    Identifier "DCV Stylus Pen"
    Driver "dcvinput"
EndSection

Section "InputDevice"
    Identifier "DCV Stylus Eraser"
    Driver "dcvinput"
EndSection

Section "InputDevice"
    Identifier "DCV Touchscreen"
    Driver "dcvinput"
EndSection
```

3. 将以下内容添加到 `ServerLayout` 部分的末尾。

```
InputDevice "DCV Stylus Pen"
InputDevice "DCV Stylus Eraser"
InputDevice "DCV Touchscreen"
```

例如：

```
Section "ServerLayout"
    ...existing content...
    InputDevice "DCV Stylus Pen"
    InputDevice "DCV Stylus Eraser"
    InputDevice "DCV Touchscreen"
```



```
EndSection
```

4. 保存更改并关闭该文件。
5. 重新启动 X 服务器。

- RHEL、Rocky、CentOS、亚马逊 Linux 2、Ubuntu 和 SUSE Linux Enterprise 12.x

```
$ sudo systemctl isolate multi-user.target
```

```
$ sudo systemctl isolate graphical.target
```

6. 要确保正确配置输入设备，请运行以下命令。

```
$ sudo DISPLAY=:0 xinput
```

DCV 触控笔、DCV 触控笔橡皮擦和 DCV 触摸屏显示在命令输出中。下面是示例输出。

```
| Virtual core pointer          id=2    [master pointer (3)]
|   | Virtual core XTEST pointer  id=4    [slave pointer (2)]
|   | dummy_mouse                 id=6    [slave pointer (2)]
|   | dummy_keyboard              id=7    [slave pointer (2)]
|   | DCV Stylus Pen                id=8    [slave pointer (2)]
|   | DCV Stylus Eraser             id=9    [slave pointer (2)]
|   | DCV Touchscreen              id=10   [slave pointer (2)]
| Virtual core keyboard         id=3    [master keyboard (2)]
|   | Virtual core XTEST keyboard id=5    [slave keyboard (3)]
```

配置触控笔压力范围

某些应用程序要求您将触控笔压力范围减小到 0 到 2048 之间。您可以在 `/etc/X11/xorg.conf` 文件中将 `Pressure2k` 选项设置为 `true` 以配置压力范围。

配置触控笔压力

1. 使用所需的文本编辑器打开 `/etc/X11/xorg.conf`。
2. 将以下部分添加到此文件中。

```
Section "InputDevice"
    Identifier "DCV Stylus Pen"
```

```
Driver "dcvinput"  
Option "Pressure2K" "true"  
EndSection  
  
Section "InputDevice"  
Identifier "DCV Stylus Eraser"  
Driver "dcvinput"  
Option "Pressure2K" "true"  
EndSection
```

3. 保存更改并关闭该文件。
4. 重新启动 X 服务器。

启用游戏手柄支持

从 NICE DCV Server 2022.0 开始，在连接到任何支持的 Windows 或 Linux 操作系统时，可以使用游戏手柄设备。

支持以下游戏手柄设备：

- Xbox 360 控制器
- DualShock 4 控制器

也可以使用与上面列出的设备兼容的其他设备，或者使用可配置为模拟支持的设备之一的其他设备。

Note

只有在使用 Windows 本机 NICE DCV 客户端时，才支持游戏手柄设备。确保您使用的是 NICE DCV Client 2022.0 或更高版本。

要启用游戏手柄支持，请确保您安装了最新版本的 NICE DCV 服务器并选择安装游戏手柄驱动程序。有关更多信息，请参阅 [在 Windows 上安装 NICE DCV 服务器](#)。在安装该驱动程序后，默认在 Windows NICE DCV 服务器上启用该功能。

支持 Xbox 360 控制器

Xbox 360 控制器要求安装其 Windows 驱动程序。不会自动在 Windows 上安装该驱动程序，需要从官方 Windows 更新网站中检索该驱动程序。

要下载并安装 Xbox 360 控制器驱动程序，请执行以下操作：

1. 在 Microsoft 更新目录页面上搜索该驱动程序：<https://www.catalog.update.microsoft.com/Search.aspx?q=game+devices+XBOX+360+Controller+For+Windows>。
2. 下载适用于您的操作系统的最新驱动程序版本。
3. 打开 .cab 文件并提取其内容：

```
expand filename.cab -F:* .
```

4. 使用以下命令安装该驱动程序的 .inf 文件：

```
pnputil /add-driver filename.inf /install
```

启用 USB 远程控制

对于 NICE DCV，客户端可以使用各种专用的 USB 设备，例如 3D 指点设备或身份验证设备。这些设备以物理方式连接到其计算机，以与 NICE DCV 服务器上运行的应用程序进行交互。

Important

NICE DCV 提供了一种重定向 USB 设备的通用机制。某些对网络延迟敏感的设备可能会出现問題。此外，由于驱动程序兼容性问题，某些设备可能无法正常工作。在部署到生产环境之前，确保您的设备正常工作。

Note

仅 Windows 客户端支持 USB 远程控制。可移植 Windows 客户端或 Web 浏览器客户端不支持该功能。可能需要在 NICE DCV 客户端上进行额外的配置。有关在客户端上安装 USB 远程控制功能的信息，请参阅《NICE DCV 用户指南》的[可安装的 Windows 客户端](#)中的可选步骤。

NICE DCV 服务器使用允许列表以确定允许客户端使用哪些 USB 设备。默认情况下，在允许列表中添加了一些常用的 USB 设备。这意味着客户端可以将这些 USB 设备连接到其计算机，并在服务器上使用这些设备，而无需进行任何额外的配置。有关更多信息，请参阅《NICE DCV 用户指南》中的[使用 USB 远程控制](#)。

但默认情况下，某些专用设备可能不会添加到允许列表中。必须将这些设备手动添加到 NICE DCV 服务器上的允许列表中，然后客户端才能使用这些设备。添加后，它们将出现在 Windows 客户端的设置菜单中。

Windows NICE DCV server

要将 USB 设备添加到允许列表，您必须从客户端获取 USB 设备的筛选条件字符串并将其添加到 `usb-devices.conf` 文件中。

将 USB 设备添加到 Windows NICE DCV 服务器上的允许列表

1. 确保您已安装最新版本的 NICE DCV 服务器，并选择安装 USB 远程控制驱动程序。有关更多信息，请参阅 [在 Windows 上安装 NICE DCV 服务器](#)。
2. 在 NICE DCV 服务器上安装 USB 设备的硬件驱动程序。
3. 在 Windows 客户端计算机上，在文件管理器中导航到 `C:\Program Files (x86)\NICE\DCV\Client\bin\`。
4. 运行 `dcvusblist.exe`。
5. 右键单击列表中的 USB 设备。
6. 从下拉菜单中选择 Copy filter string。
7. 在服务器上，使用常用的文本编辑器打开 `C:\Program Files\NICE\DCV\Server\conf\usb-devices.conf`，并将筛选条件字符串添加到文件底部的新行中。
8. 保存并关闭文件。
9. [停止并重新启动](#) NICE DCV 服务器。

Linux NICE DCV server

要将 USB 设备添加到允许列表中，请将 USB 设备的筛选条件字符串添加到 `usb-devices.conf` 文件中。

将 USB 设备添加到 Linux NICE DCV 服务器上的允许列表

1. 确保您已安装最新版本的 NICE DCV 服务器和 DCV USB 驱动程序。有关更多信息，请参阅 [在 Linux 上安装 NICE DCV 服务器](#)。
2. 在 NICE DCV 服务器上安装 USB 设备的硬件驱动程序。
3. 在 Windows 客户端计算机上，在文件管理器中导航到 `C:\Program Files (x86)\NICE\DCV\Client\bin\`。

4. 运行 `dcvusblis.exe`。
5. 右键单击列表中的 USB 设备。
6. 从下拉菜单中选择 Copy filter string。
7. 在服务器上，使用常用的文本编辑器打开 `/etc/dcv/usb-devices.conf`，并将筛选条件字符串添加到文件底部的新行中。
8. 保存并关闭文件。
9. [停止并重新启动](#) NICE DCV 服务器。

配置智能卡缓存

通过使用智能卡缓存功能，NICE DCV 服务器可以缓存智能卡值。如果启用了该功能，则 NICE DCV 服务器缓存最近调用客户端智能卡的结果。未来的调用直接从服务器的缓存中检索，而不是从客户端检索。这会减少在客户端和服务器之间传输的流量并提高性能。如果客户端的互联网连接速度较慢，这是尤其有用的。

默认情况下，将禁用智能卡缓存。客户可以通过设置 `DCV_PCSC_ENABLE_CACHE` 环境变量为他们运行的每个应用程序手动启用智能卡缓存。有关说明，请参阅《NICE DCV 用户指南》中的[使用智能卡](#)。或者，您可以将 NICE DCV 服务器配置为永久启用或禁用智能卡缓存，而无论为 `DCV_PCSC_ENABLE_CACHE` 环境变量指定的值如何。

Linux NICE DCV server

在 Linux NICE DCV 服务器上永久启用或禁用智能卡缓存

1. 导航到 `/etc/dcv/`，并使用您常用的文本编辑器打开 `dcv.conf`。
2. 在 `[smartcard]` 部分中找到 `enable-cache` 参数。要永久启用智能卡缓存，请输入 `'always-on'`。要永久禁用智能卡缓存，请输入 `'always-off'`。

如果在 `[smartcard]` 部分中没有 `enable-cache` 参数，请按以下格式手动添加该参数：

```
[smartcard]
enable-cache='always-on' | 'always-off'
```

3. 保存并关闭文件。
4. [停止并重新启动](#) NICE DCV 服务器。

Windows NICE DCV server

在 Windows NICE DCV 服务器上永久启用或禁用智能卡缓存

1. 打开 Windows 注册表编辑器。
2. 导航到 HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/smartcard/ 项，然后选择 enable-cache 参数。

如果该参数不存在，请使用以下步骤创建该参数：

- a. 在左侧窗格中，打开 smartcard 项的上下文（右键单击）菜单，然后选择新建 > 字符串值。
 - b. 对于名称，输入 enable-cache 并按 Enter。
3. 打开 enable-cache 参数。对于值数据，输入 always-on 以永久启用智能卡缓存，或者输入 always-off 以永久禁用智能卡缓存。
 4. 选择确定，并关闭 Windows 注册表编辑器。

配置 WebAuthn 重定向

从 NICE DCV Server 2023.1 开始，用户可以在远程会话中支持的浏览器中对使用 Web 身份验证 (WebAuthn) 标准的 Web 应用程序进行身份验证。这是通过将身份验证提示重定向到本地连接的 FIDO2 身份验证器（例如 Windows Hello 或 YubiKey 任何其他符合 FIDO2 标准的身份验证器）来完成的。

WebAuthn 重定向独立于 USB 重定向运行。无需在 NICE DCV 服务器上安装任何供应商特定的驱动程序。通过浏览器的本机 API 可以实现 WebAuthn 请求的重定向。

在使用之前 WebAuthn，请仔细检查 [“支持的功能”](#) 表，确保满足所有要求。

支持的浏览器：

- Google Chrome 116 和更高版本
- Microsoft Edge 116 和更高版本

WebAuthn 可以使用 webauthn-redirect 权限启用或禁用重定向。有关更多信息，请参阅 [使用权限文件](#)。

WebAuthn 重定向需要在远程服务器上安装浏览器扩展程序。启用该功能并安装浏览器扩展程序后，会话中浏览器中运行的 Web 应用程序发起的任何 WebAuthn 请求都将无缝重定向到本地客户端。然后，用户可以使用诸如 Windows Hello 之类 YubiKey 的设备或完成身份验证。

Note

虽然此功能允许 WebAuthn 在远程会话期间在浏览器中使用，但它不支持使用 WebAuthn 身份验证器进行 DCV 会话身份验证。

设置 WebAuthn 重定向浏览器扩展

首次启动浏览器时自动提示

安装启用 WebAuthn 重定向功能的 NICE DCV Server 2023.1 后，用户在首次启动浏览器时会被提示启用浏览器扩展程序。如果他们选择不安装扩展程序或稍后将其卸载，则 WebAuthn 重定向将不起作用。管理员可以使用组策略强制安装。

使用组策略进行安装

对于希望更广泛地部署扩展程序的组织，您可以使用组策略。

使用 Microsoft Edge：

1. 下载并安装 [Microsoft Edge 管理模板](#)。
2. 启动组策略管理工具 (gpmc.msc)。
3. 导航：“林”>“域”>“您的 FQDN” (例如 example.com) >“组策略对象”。
4. 选择所需的策略或创建新的策略，然后右键单击该策略并选择“编辑”。
5. 采用以下路径：“计算机配置”>“管理模板”>“Microsoft Edge”>“扩展”。
6. 访问“配置扩展管理设置”，并将其设置为“已启用”。
7. 在“配置扩展管理设置”字段中，输入以下内容：

```
{"ihejeaahjpbegmaaegiikmlphghlfmeh":  
{"installation_mode":"force_installed","update_url":"https://edge.microsoft.com/  
extensionwebstorebase/v1/crx"}}
```

8. 保存更改并重新引导服务器。

使用 Google Chrome :

1. 获取并实施 [Google Chrome 管理模板](#)。
2. 与 Microsoft Edge 的步骤类似，通过组策略管理工具进行导航。
3. 转到：“计算机配置”>“管理模板”>“Google Chrome”>“扩展”。
4. 访问“配置扩展管理设置”，并将其设置为“已启用”。
5. 在“配置扩展管理设置”字段中，输入以下内容：

```
{"mmiioagbgnbojdbcjoddlfahmcocfpmn":  
{ "installation_mode":"force_installed","update_url":"https://clients2.google.com/  
service/update2/crx"}}
```

6. 保存更改并重新引导服务器。

手动安装

可以从相应的浏览器商店中获取扩展：

- [Microsoft Edge 加载项](#)
- [Chrome 网上商店](#)

对于手动安装：

1. 连接到你的 NICE DCV 会话。
2. 打开您的首选浏览器，然后导航到相关的浏览器商店（上面的链接）。
3. 选择“获取”（Microsoft Edge）或“添加到 Chrome”（Google Chrome）以继续。
4. 按照屏幕上的说明进行操作。在成功添加扩展后，将显示确认消息。

在隐身模式下使用 WebAuthn 重定向（仅限 Chrome）

使用隐身模式时，需要特别允许 Amazon DCV WebAuthn 重定向扩展在其中运行，否则将不会发生 WebAuthn 重定向。要实现此目的，应按照以下步骤进行：

1. 打开分机设置。
2. 在详细信息中找到“隐身模式”中的允许。
3. 将开关切换到“开”。

启用会话存储

会话存储是 NICE DCV 服务器上的一个文件夹，客户端可以在连接到特定 NICE DCV 会话时访问该文件夹。当您启用会话的会话存储时，客户端既可从指定文件夹下载文件，又可将文件上传到指定文件夹。利用此功能，客户端可在连接到会话时共享文件。

主题

- [在 Windows NICE DCV 服务器上启用会话存储](#)
- [在 Linux NICE DCV 服务器上启用会话存储](#)

在 Windows NICE DCV 服务器上启用会话存储

要启用会话存储，请先创建要用于会话存储的文件夹。然后，使用 Windows 注册表编辑器配置 `storage-root` 参数。

在 Windows 上启用会话存储

1. 创建要用于会话存储的文件夹（例如 `c:\session-storage`）。
2. 配置 `storage-root` 参数。
 - a. 打开 Windows 注册表编辑器。
 - b. 导航到 `HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/session-management/automatic-console-session` 注册表项并选择 `storage-root` 参数。

如果在注册表项中没有 `storage-root` 参数，请按以下方式创建该参数：

- i. 在导航窗格中，打开 `session-management/automatic-console-session` 项的上下文（右键单击）菜单。然后，选择新建 > 字符串。
 - ii. 对于名称，输入 `storage-root` 并按 Enter。
- c. 打开 `storage-root` 参数。对于值数据，输入在步骤 1 中创建的文件夹的完整路径。

您也可以在路径中使用 `%home%` 以指定当前登录用户的主目录。例如，以下路径使用 `c:\Users\username\storage\` 作为会话存储目录。

```
%home%/storage/
```

Note

如果指定的子目录不存在，则会禁用会话存储。

- d. 选择确定，并关闭 Windows 注册表编辑器。
 - e. [停止并重新启动](#) NICE DCV 服务器。
3. 启动会话并指定 `--storage-root` 选项。有关更多信息，请参阅 [启动 NICE DCV 会话](#)。

在 Linux NICE DCV 服务器上启用会话存储

要启用会话存储，请创建用于会话存储的文件夹，然后在 `dcv.conf` 文件中配置 `storage-root` 参数。

在 Linux 上启用会话存储

1. 创建要用于会话存储的文件夹（例如 `/opt/session-storage/`）。
2. 配置 `storage-root` 参数。
 - a. 导航到 `/etc/dcv/`，并使用您常用的文本编辑器打开 `dcv.conf`。
 - b. 在 `[session-management/automatic-console-session]` 部分中找到 `storage-root` 参数。将现有路径替换为步骤 1 中创建的文件夹的完整路径。

如果在 `[session-management/automatic-console-session]` 部分中没有 `storage-root` 参数，请按以下格式手动添加该参数。

```
[session-management/automatic-console-session]
storage-root="/opt/session-storage/"
```

您也可以在路径中使用 `%home%` 以指定当前登录用户的主目录。例如，以下参数使用会话存储的 `$HOME/storage/` 目录。

```
[session-management/automatic-console-session]
storage-root="%home%/storage/"
```

Note

如果指定的子目录不存在，则会禁用会话存储。

3. 保存并关闭文件。
4. [停止并重新启动](#) NICE DCV 服务器。
5. 启动会话并指定 `--storage-root` 选项。有关更多信息，请参阅 [启动 NICE DCV 会话](#)。

在 Linux NICE DCV 服务器上配置打印机

NICE DCV 允许您打印到本地重定向的打印机或虚拟 NICE DCV 打印机。

如果您使用支持的 Linux 发行版，您必须配置 NICE DCV 服务器以支持打印。

如果您使用 Windows NICE DCV 服务器，则无需进行额外的配置。

在 Linux NICE DCV 服务器上启用打印机重定向

1. 在您的服务器上安装 CUPS 服务。
 - Amazon Linux 2、RHEL 和 CentOS

```
$ sudo yum install cups
```

- Ubuntu

```
$ sudo apt-get install cups
```

- SUSE Linux Enterprise

```
$ sudo zypper install cups
```

2. 将 `dcv` 用户添加到打印机管理员组中。打印机管理员组的名称可能因操作系统而异。例如，如果您的打印机管理员组命名为 `lpadmin`，请运行以下命令：

```
$ usermod -a -G lpadmin dcv
```

3. 确保在 cups 配置文件的 SystemGroup 参数中引用了打印机管理员组。例如，如果您的打印机管理员组命名为 lpadmin，请使用文本编辑器打开 /etc/cups/cups-files.conf 并查找以下行。

```
SystemGroup lpadmin
```

如果在配置文件中出现该行，则说明安装已完成。继续下一步。

如果在配置文件中未出现该行，请按以下格式手动添加该行，然后保存并关闭该文件。

```
SystemGroup printer_admin_groupname
```

4. (仅限 SUSE Linux Enterprise) 确保打印机管理员组有权读取 cups 本地证书。该证书位于以下目录中：/var/run/cups/certs/。例如，如果您的打印机管理员组命名为 lpadmin，请运行以下命令：

```
$ sudo chgrp -R lpadmin /var/run/cups/certs/ && chmod g+x /var/run/cups/certs
```

5. 重新启动 cups 服务。

```
$ sudo systemctl restart cups
```

6. [停止](#)并[重新启动](#) NICE DCV 服务器。

解决打印机问题

SUSE Linux Enterprise 和 RHEL 8 可能会禁止连接到打印机插槽。如果您运行这两种操作系统之一并遇到打印问题，请检查日志文件以确定这是否为问题原因。

使用文本编辑器打开 /var/log/audit/audit.log，并检查日志是否具有类似下面的行：

```
type=AVC msg=audit(1617716179.487:504): avc: denied { connectto } for pid=33933
comm="dcvcupsbackend"
path=002F636F6D2F6E696365736F6674776172652F6463762F637570732F636F6E736F6C65
scontext=system_u:system_r:cupsd_t:s0-s0:c0.c1023
tcontext=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
tclass=unix_stream_socket permissive=0
```

如果在日志文件中出现类似的行，则操作系统禁止访问打印机插槽。

要解决该问题，您必须创建一个允许访问打印机插槽的 CUPS 策略。为此，请执行以下步骤：

1. 创建所需的策略文件。使用常用的文本编辑器创建一个名为 `cupsd_policy` 的新文件，并添加以下内容。

```
#===== cupsd_t =====  
allow cupsd_t unconfined_t:unix_stream_socket connectto;
```

2. 安装该策略。

```
$ ausearch -c 'dcvcupsbackend' --raw | audit2allow -M dcv-printer-policy
```

```
$ semodule -X 300 -i dcv-printer-policy.pp
```

在 Linux NICE DCV 服务器上配置剪贴板

Linux 操作系统具有两个可用于复制和粘贴内容的缓冲区。这两个缓冲区是主选区和剪贴板。要将内容复制到主选区中，请拖动指针以突出显示内容。要从主选区中粘贴内容，请使用指针或 `Shift+Insert` 键盘快捷键。要将内容复制到剪贴板，请突出显示内容并从上下文（右键单击）菜单中选择复制。要从剪贴板粘贴内容，请从上下文（右键单击）菜单中选择粘贴。

在 Linux NICE DCV 服务器上，您可以将服务器配置为在客户端和服务器之间执行复制和粘贴操作时使用主选区或剪贴板。

主题

- [将客户端剪贴板内容粘贴到主选区](#)
- [将主选区内容复制到客户端剪贴板](#)

将客户端剪贴板内容粘贴到主选区

默认情况下，在客户端中复制的内容将放置在剪贴板中。要将此内容粘贴到服务器上，必须使用上下文（右键单击）菜单从剪贴板粘贴内容。

可以将服务器配置为将剪贴板内容放入主选区中。这样，用户就可以使用上下文（右键单击）菜单从剪贴板中粘贴复制的内容。或者，他们也可以使用鼠标中键或 `Shift+Insert` 键盘快捷键粘贴从主选区中复制的内容。

将服务器配置为将剪贴板内容放入主选区中

1. 导航到 `/etc/dcv/`，并使用您常用的文本编辑器打开 `dcv.conf`。
2. 找到 `[clipboard]` 部分中的 `primary-selection-paste` 参数并将值设置为 `true`。

如果在 `[clipboard]` 部分中没有 `primary-selection-paste` 参数，请按以下格式手动添加该参数：

```
[clipboard]
primary-selection-paste=true
```

3. 保存并关闭文件。
4. [停止并重新启动](#) NICE DCV 会话。

将主选区内容复制到客户端剪贴板

默认情况下，用户只能使用剪贴板将内容从服务器复制到客户端。这意味着无法将复制到主选区的内容粘贴到客户端上。

可以将服务器配置为将主选区内容放入剪贴板中。这意味着当用户将内容复制到服务器上的主选区时，内容也会复制到剪贴板中。这也意味着用户可以将剪贴板中的内容粘贴到客户端中。

将服务器配置为将主选区内容放入剪贴板中

1. 导航到 `/etc/dcv/`，并使用您常用的文本编辑器打开 `dcv.conf`。
2. 找到 `[clipboard]` 部分中的 `primary-selection-copy` 参数并将值设置为 `true`。

如果在 `[clipboard]` 部分中没有 `primary-selection-copy` 参数，请按以下格式手动添加该参数：

```
[clipboard]
primary-selection-copy=true
```

3. 保存并关闭文件。
4. [停止并重新启动](#) NICE DCV 会话。

配置多声道音频

在使用 NICE DCV 本机客户端时，NICE DCV 最多支持 7.1 音频声道。Web 浏览器客户端仅支持立体声 2.0 音频声道。

NICE DCV 支持以下多声道音频配置：

- 立体声 2.0 (两个声道)
- 四声道 4.0 (四个声道)
- 环绕声 5.1 (6 个声道)
- 环绕声 7.1 (8 个声道) - 仅限 Windows NICE DCV 服务器



如果客户端请求的音频声道数量少于服务器提供的声道数量，则服务器缩混声道数。这是为了与客户端请求的声道数匹配。例如，假设客户端请求环绕声 5.1，而服务器最多支持环绕声 7.1。服务器将音频缩混为 5.1。

服务器不会自动缩混音频以与源应用程序的音频输出匹配。例如，假设源应用程序提供环绕声 7.1，而客户端仅支持立体声 2.0。仅左前和右前音频声道流式传输到客户端。其余声道丢失。如果是这种情况，为了防止音频声道丢失，请将 NICE DCV 服务器配置为缩混音频声道。

主题

- [在 Windows NICE DCV 服务器上配置音频声道](#)

- [在 Linux NICE DCV 服务器上配置音频声道](#)

在 Windows NICE DCV 服务器上配置音频声道

Windows 服务器支持环绕声 7.1 (8 个音频声道)。默认配置是立体声。不过，您可以将服务器配置为使用不同的配置。

在 Windows 服务器上配置音频声道：

1. 打开“声音”控制面板。从桌面的任务栏中，右键单击扬声器图标，然后选择声音。
2. 打开“播放”选项卡，然后选择 NICE DCV 扬声器。
3. 选择配置。
4. 选择常用的声道配置。
5. 选择确定。

在 Linux NICE DCV 服务器上配置音频声道

Linux 服务器默认支持立体声 2.0 (两个音频声道)，并需要进行一些额外的配置以支持多声道音频。

您需要执行以下操作：

1. 配置 PulseAudio 声音服务器。
2. 将 NICE DCV 服务器配置为使用该设备。PulseAudio
3. 配置要使用的声道数。

配置 PulseAudio 声音服务器

1. 使用首选文本编辑器打开 `/etc/pulse/default.pa`。
2. 将以下行添加到文件的末尾。

```
load-module module-null-sink sink_name=dcv format=s16be channels=6
channel_map=front-left,front-right,rear-left,rear-right,front-center,lfe
rate=48000 sink_properties="device.description='DCV Audio Speakers'"
```

3. 保存并关闭文件。

配置 PulseAudio 声音服务器后，必须将 NICE DCV 服务器配置为从 PulseAudio 声音服务器捕获音频。

将 NICE DCV 服务器配置为使用该设备 PulseAudio

1. 使用首选文本编辑器打开 `/etc/dcv/dcv.conf`。
2. 在 `[audio]` 部分中找到 `grab-device` 参数。然后，将现有的值替换为您在上一步中检索的设备名称。

如果在 `[audio]` 部分中没有 `grab-device` 参数，请按以下格式手动添加该参数：

```
[audio]
grab-device="DCV Audio Speakers"
```

3. 保存并关闭文件。

将 NICE DCV 服务器配置为从 PulseAudio 声音服务器捕获音频后，可以指定要使用的声道数。

配置要使用的声道数

1. 使用首选文本编辑器打开 `/etc/dcv/dcv.conf`。
2. 在 `[audio]` 部分中找到 `source-channels` 参数。然后，将现有声道数替换为以下值之一：2.0 为 2，4.0 为 4，5.1 为 6。

如果在 `[audio]` 部分中没有 `source-channels` 参数，请按以下格式手动添加该参数：

```
[audio]
source-channels=channels
```

3. 保存并关闭文件。
4. [停止并重新启动](#) NICE DCV 服务器。

配置 HTTP 标头

您可以配置 NICE DCV 服务器，在用户使用 Web 浏览器客户端连接到会话时，向 NICE DCV 客户端发送额外的 HTTP 响应标头。响应标头可以提供有关用户连接到的 NICE DCV 服务器的额外信息。

主题

- [在 Windows NICE DCV 服务器上配置 HTTP 标头](#)

- [在 Linux NICE DCV 服务器上配置 HTTP 标头](#)

在 Windows NICE DCV 服务器上配置 HTTP 标头

要在 Windows 上配置 HTTP 标头，请使用 Windows 注册表编辑器配置 `web-extra-http-headers` 参数。

在 Windows 上配置 HTTP 标头

1. 打开 Windows 注册表编辑器。
2. 导航到 `HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/connectivity/` 注册表项。
3. 在导航窗格中，打开 `connectivity` 项的上下文（右键单击）菜单。然后，选择新建 > 字符串。
4. 对于名称，输入 `web-extra-http-headers` 并按 Enter。
5. 打开 `web-extra-http-headers` 参数。对于值数据，按以下格式输入 HTTP 标头名称和值。

```
[("header-name", "header-value")]
```

要指定多个标头，请将它们添加到以逗号分隔的列表中。

```
[("header1-name", "header1-value"), ("header2-name", "header2-value")]
```

6. 选择确定，并关闭 Windows 注册表编辑器。
7. [停止并重新启动](#) NICE DCV 服务器。

在 Linux NICE DCV 服务器上配置 HTTP 标头

要在 Linux 上配置 HTTP 标头，请配置 `dcv.conf` 文件中的 `web-extra-http-headers` 参数。

在 Linux 上配置 HTTP 标头

1. 使用首选文本编辑器打开 `/etc/dcv/dcv.conf`。
2. 找到 `[connectivity]` 部分。按以下格式指定 HTTP 标头名称和值。

```
[connectivity]
web-extra-http-headers=[("header-name", "header-value")]
```

要指定多个标头，请将它们添加到以逗号分隔的列表中。

```
[connectivity]
web-extra-http-headers=[("header1-name", "header1-value"), ("header2-name",
"header2-value")]
```

3. 保存并关闭文件。
4. [停止](#)并[重新启动](#) NICE DCV 服务器。

配置 NICE DCV 身份验证

默认情况下，在连接到 NICE DCV 会话之前，客户端需要在托管 NICE DCV 的服务器中进行身份验证。如果客户端无法进行身份验证，这可能是由于禁止客户端连接到会话。可禁用客户端身份验证要求以允许客户端在不针对服务器进行身份验证的情况下连接到会话。

NICE DCV 支持以下身份验证方法：

- **system** - 这是默认身份验证方法。客户端身份验证工作委派给基础操作系统。对于 Windows NICE DCV 服务器，身份验证委托给 WinLogon 对于 Linux NICE DCV 服务器，身份验证工作委派给 PAM。在连接到 NICE DCV 会话时，客户端提供其系统凭证。验证您的客户端是否具有 NICE DCV 服务器的相应登录凭证。
- **none** - 在连接到 NICE DCV 会话时，不需要进行客户端身份验证。NICE DCV 服务器为尝试连接到会话的所有客户端授予访问权限。

确保您的客户端了解 NICE DCV 服务器使用的身份验证方法。客户端还应该确保具有连接到会话所需的信息。

主题

- [在 Windows 上配置身份验证](#)
- [在 Linux 上配置身份验证](#)
- [配置使用外部身份验证器的身份验证](#)

在 Windows 上配置身份验证

要更改 NICE DCV 服务器的身份验证方法，您必须使用 Windows 注册表编辑器配置 `authentication` 参数。

在 Windows 上更改身份验证方法

1. 打开 Windows 注册表编辑器。
2. 导航到 HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/security/ 注册表项，并选择 authentication 参数。

如果在注册表项中没有 authentication 参数，请创建该参数：

- a. 在导航窗格中，打开 authentication 项的上下文（右键单击）菜单。然后，选择新建 > 字符串值。
 - b. 对于名称，输入 authentication 并按 Enter。
3. 打开 authentication 参数。对于数值数据，输入 system 或 none。
 4. 选择确定，并关闭 Windows 注册表编辑器。

Windows 凭证提供程序

在使用 Windows 凭证提供程序时，如果用户可以在 DCV 服务器中进行身份验证，则他们可以绕过 Windows 登录。

仅当 DCV authentication 参数设置为 system 时，才支持 Windows 凭证提供程序。如果 DCV authentication 参数设置为 none，则用户必须在针对 DCV 服务器自动进行身份验证后手动登录到 Windows。

默认情况下，在安装 NICE DCV 服务器时，将启用 Windows 凭证提供程序。

禁用 Windows 凭证提供程序

1. 打开 Windows 注册表编辑器。
2. 导航到 HKEY_LOCAL_MACHINE\ SOFTWARE\ Microsoft\ Windows\ AuthenticationCurrentVersion\ 凭证提供者\ {8A2C93D0-D55F-4045-99D7-B27F5E263407} 密钥。
3. 依次选择编辑、新建、DWORD 值。
4. 对于名称，输入 **Disabled**。
5. 打开该值。对于值数据，输入 1 并选择确定。
6. 关闭 Windows 注册表编辑器。

重新启用 Windows 凭证提供程序

1. 打开 Windows 注册表编辑器。
2. 导航到 HKEY_LOCAL_MACHINE\ SOFTWARE\ Microsoft\ Windows\ AuthenticationCurrentVersion\ 凭证提供者\ {8A2C93D0-D55F-4045-99D7-B27F5E263407} 密钥。
3. 打开已禁用值。对于值数据，输入 0 并选择确定。
4. 关闭 Windows 注册表编辑器。

在 Linux 上配置身份验证

要更改 NICE DCV 服务器的身份验证方法，您必须配置 `dcv.conf` 文件中的 `authentication` 参数。

在 Linux 上更改身份验证方法

1. 导航到 `/etc/dcv/`，并使用您常用的文本编辑器打开 `dcv.conf`。
2. 在 `[security]` 部分中找到 `authentication` 参数。然后，将现有的值替换为 `system` 或 `none`。

如果在 `[security]` 部分中没有 `authentication` 参数，请按以下格式添加该参数。

```
[security]
authentication=method
```

3. 保存并关闭文件。

PAM 服务

在 Linux 上，在 NICE DCV 参数 `authentication` 设置为 `system` 时，将执行 PAM 服务以进行身份验证。

默认情况下，NICE DCV 服务器执行的特权访问管理 (PAM) 服务为 `/etc/pam.d/dcv`。

如果要更改通过 NICE DCV 验证用户身份时 PAM 执行的步骤，您可以在 `dcv.conf` 的 `authentication` 部分中设置 `pam-service` 参数。

更改 PAM 服务

1. 以根用户身份导航到 `/etc/pam.d` 目录，并创建一个新文件，例如 `dcv-custom`。
2. 使用常用的文本编辑器编辑 `dcv-custom` 文件。有关 PAM 服务文件的语法，请参阅系统文档。
3. 导航到 `/etc/dcv/`，并使用您常用的文本编辑器打开 `dcv.conf`。
4. 在 `[authentication]` 部分中找到 `pam-service` 参数。然后，将现有服务名称替换为新的 PAM 服务名称。

如果在 `[authentication]` 部分中没有 `pam-service` 参数，请按以下格式手动添加该参数：

```
[authentication]
pam-service=service_name
```

Note

PAM 服务名称必须与您在 `/etc/pam.d` 中创建的文件的名称匹配。

5. 保存并关闭文件。

配置使用外部身份验证器的身份验证

可以将 DCV 配置为使用外部身份验证器。有关该过程及其要求的更多信息，请参阅[使用外部身份验证](#)。

配置 NICE DCV 授权

授权用于为 NICE DCV 客户端授予或拒绝特定 NICE DCV 功能的权限。在 NICE DCV 中，授权是使用权限文件配置的。权限文件定义特定用户在连接到会话时可以使用的特定 NICE DCV 功能。

NICE DCV 支持两种类型的权限文件：

主题

- [默认权限文件](#)
- [自定义权限文件](#)
- [使用权限文件](#)

默认权限文件

如果您在创建会话时未指定自定义权限文件，则将默认权限文件用于所有会话。默认权限文件仅向会话拥有者授予对所有功能的完全访问权限。

可以自定义默认权限文件以包含自定义授权。默认权限文件位于 `n C:\Program Files\NICE\DCV\Server\conf\default.perm` (在 Windows NICE DCV 服务器上) 和 `/etc/dcv/default.perm` (在 Linux NICE DCV 服务器上)。

有关自定义默认权限文件的信息，请参阅[使用权限文件](#)。

自定义权限文件

您可以使用自定义权限文件定义特定用户或组在连接到 NICE DCV 会话时有权访问的功能。在使用自定义权限文件时，将覆盖默认权限文件。

要使用自定义权限文件，您必须先创建该权限文件。接下来，在将 `--permissions-file` 选项与 `dcv create-session` 命令结合使用来启动会话时指定它。有关启动会话的更多信息，请参阅[启动 NICE DCV 会话](#)。

有关创建自定义权限文件的信息，请参阅[使用权限文件](#)。

使用权限文件

可以使用首选文本编辑器创建自定义权限文件或更新现有权限文件。权限文件通常采用以下格式：

```
#import file_to_import

[groups]
group_definitions

[aliases]
alias_definitions

[permissions]
user_permissions
```

以下部分说明如何在更新或创建权限文件时填充这些部分。

目录

- [导入权限文件](#)
- [创建组](#)
- [创建别名](#)
- [添加权限](#)

导入权限文件

`imports` 部分通常是权限文件的第一个部分。您可以使用该部分引用和包含现有的权限文件。您还可以使用该部分将以前定义的 NICE DCV 权限合并到您的权限文件中。

权限文件可包含多个导入。导入的权限文件可能导入其他权限文件。

将一个权限文件导入您的权限文件中

- 使用 `#import` 语句并使用绝对或相对路径指定该文件的位置
 - Windows NICE DCV 服务器：

```
#import ..\file_path\file
```

- Linux NICE DCV 服务器：

```
#import ../file_path/file
```

示例

以下语句使用绝对路径导入名为 `dcv-permissions.file` 的权限文件。它位于 Windows NICE DCV 服务器上的 NICE DCV 安装文件夹中。

```
#import c:\Program Files\NICE\DCV\dcv-permissions.file
```

创建组

您可以使用权限文件的 `[groups]` 部分，为具有类似使用案例或权限要求的用户定义用户组。可以为组分配特定的权限。分配给组的权限适用于组中包含的所有用户。

要在权限文件中创建组，您必须先将组部分标题添加到文件中。

```
[groups]
```

然后，您可以在部分标题的下方创建您的组。要创建组，请提供组名称，然后在逗号分隔的列表中指定组成员。组成员可以是单个用户、其他组和操作系统用户组。

```
group_name=member_1, member_2, member_3
```

如何为群组添加用户

指定用户名。

Note

您可以为用户名添加前缀 `user:`。Windows 域用户名可以包含域名。

```
group_name=user_1, user:user_2, domain_name\user_3
```

将现有组添加到一个组

指定前缀为 `group:` 的组名

```
group_name=group:group_1, group:group_2
```

将操作系统用户组添加到一个组 (仅限 Linux NICE DCV 服务器)

指定前缀为 `osgroup:` 的组名

```
group_name=osgroup:os_group_1, osgroup:os_group2
```

示例

以下示例添加 `groups` 部分标题，并创建一个名为 `my-group` 的组。该组包含各个用户。用户的名称为 `john` 和 `jane`。其中的一个组是名为 `observers` 的现有组。另一个组是名为 `guests` 的操作系统用户组：

```
[groups]  
my-group=john, user:jane, group:observers, osgroup:guests
```

创建别名

您可以使用权限文件的 [aliases] 部分创建 NICE DCV 功能集。在定义别名后，您可以为组或各个用户授予或拒绝使用该别名的权限。授予或拒绝对别名的权限将授予或拒绝对其中包含的所有功能的权限。

要在权限文件中创建别名，您必须先将别名部分标题添加到文件中。

```
[aliases]
```

然后，您可以在部分标题的下方创建您的别名。要创建别名，请提供别名名称，然后在逗号分隔的列表中指定别名成员。别名成员可以是单独的 NICE DCV 功能或其他别名。

```
alias_name=member_1, member_2, member_3
```

示例

以下示例添加 aliases 部分标题，并创建一个名为 file-management 的别名。它包括 file-upload 和 file-download 功能以及名为 clipboard-management 的现有别名。

```
[aliases]  
file-management=file-upload, file-download, clipboard-management
```

添加权限

权限文件的 [permissions] 部分可让您控制用户和组对特定功能或别名的访问权。

要将权限添加到权限文件，请先将权限部分标题添加到文件中。

```
[permissions]
```

然后，您可以在部分标题的下方添加您的权限。要添加一个权限，请指定该权限控制的角色、要应用的规则 and 该权限应用于的功能。

```
actor rule features
```

角色可以是用户、组或操作系统组。组的前缀必须为 group:。操作系统组必须以 osgroup: 为前缀。NICE DCV 包含一个内置 %owner% 引用，可用于引用会话所有者。它还用于引用内置的 %any % 引用，后者可用于引用任何用户。

以下规则可用于权限语句：

- allow - 授予功能的访问权限。
- disallow - 拒绝功能的访问权限，但可以被后续权限覆盖。
- deny - 拒绝功能的访问权限，但不能被后续权限覆盖。

这些功能可以包括单独的 NICE DCV 功能、别名或两者的组合。功能列表必须以空格分隔。NICE DCV 包含一个内置的 `builtin` 别名，其中包含所有 NICE DCV 功能。

可在权限文件中引用以下功能：

- audio-in - 将音频从客户端插入到 NICE DCV 服务器中。
- audio-out - 在客户端上播放 NICE DCV 服务器音频。
- builtin - 所有功能。
- clipboard-copy - 将数据从 NICE DCV 服务器复制到客户端剪贴板。
- clipboard-paste - 将数据从客户端剪贴板粘贴到 NICE DCV 服务器。
- display - 从 NICE DCV 服务器接收视觉数据。
- extensions-client - 允许启动 NICE DCV 客户端上安装的扩展。
- extensions-server - 允许启动 NICE DCV 服务器上安装的扩展。
- file-download - 从会话存储中下载文件。
- file-upload - 将文件上传到会话存储中。
- gamepad - 在会话中使用连接到客户端计算机的游戏手柄。在 NICE DCV 2022.0 和更高版本上支持。
- keyboard - 从客户端键盘输入到 NICE DCV 服务器。
- keyboard-sas - 使用安全注意序列 (CTRL+Alt+Del)。需要 keyboard 功能。在 NICE DCV 2017.3 和更高版本上支持。
- mouse - 从客户端指针输入到 NICE DCV 服务器。
- pointer - 查看 NICE DCV 服务器鼠标位置事件和指针形状。在 NICE DCV 2017.3 和更高版本上支持。
- printer - 创建从 NICE DCV 服务器发送到客户端的 PDF 或 XPS 文件。
- screenshot - 保存远程桌面的屏幕截图。在 NICE DCV 2021.2 和更高版本上支持该功能。

在删除 `screenshot` 授权时，我们建议您禁用 `clipboard-copy` 权限。这会禁止用户在服务器剪贴板上捕获屏幕截图并将其粘贴到客户端上。在拒绝 `screenshot` 授权时，Windows 和 macOS 还

会禁止外部工具捕获客户端的屏幕截图。例如，在 NICE DCV 客户端窗口上使用 Windows 截图工具将导致黑色图像。

- smartcard - 从客户端中读取智能卡。
- stylus - 从专用的 USB 设备中输入，例如 3D 指点设备或绘图板。
- touch - 使用本机触摸事件。在 DCV 2017.3 版及更高版本上受支持。
- unsupervised-access - 用于为协作会话中的用户设置无所有者访问权限。
- usb - 使用客户端中的 USB 设备。
- webcam - 在会话中使用连接到客户端计算机的网络摄像头。在 NICE DCV 2021.0 和更高版本上支持。
- webauthn-redirect - 将 WebAuthn 请求从远程浏览器重定向到本地客户端。在 NICE DCV 2023.1 和更高版本上支持。

示例

以下示例添加权限部分标头和 4 项权限。第一项权限向名为 john 的用户授予对 display、file-upload 和 file-download 功能的访问权。第二项权限拒绝 observers 组对 audio-in 和 audio-out 功能以及别名为 clipboard-management 的功能的访问权。第三项权限向 guests 操作系统组授予对 clipboard-management 和 file-management 别名的访问权。第四项权限向会话所有者授予对所有功能的访问权。

```
[permissions]
john allow display file-upload file-download
group:observers deny audio-in audio-out clipboard-management
osgroup:guests allow clipboard-management file-management
%owner% allow builtin
```

管理 NICE DCV 会话

您必须先要在 NICE DCV 服务器上创建 NICE DCV 会话，然后您的客户端才能连接到会话。只有在存在活动会话时，客户端才能连接到 NICE DCV 服务器。

每个 NICE DCV 会话具有以下属性：

- 会话 ID - 用于标识 NICE DCV 服务器上的特定会话。
- 所有者 - 创建会话的 NICE DCV 用户。默认情况下，仅所有者可以连接到会话。

NICE DCV 客户端需要此信息来连接到会话。

主题

- [NICE DCV 会话简介](#)
- [使用命令行工具管理 NICE DCV 会话](#)
- [启动 NICE DCV 会话](#)
- [停止 NICE DCV 会话](#)
- [管理运行的 NICE DCV 会话](#)
- [管理会话时区](#)
- [查看 NICE DCV 会话](#)
- [获取 NICE DCV 会话屏幕截图](#)

NICE DCV 会话简介

NICE DCV 提供两种类型的会话：控制台会话和虚拟会话。下表简要说明了两种类型的会话之间的差异。

会话类型	支持	多个会话	所需的权限	直接捕获屏幕	GPU 加速的 OpenGL 支持
控制台	Linux 和 Windows NICE DCV 服务器	否，在每个服务器上仅允许一个控制台会话	仅管理员用户可以启动和关闭会话	是	是，无需额外的软件
虚拟	仅 Linux NICE DCV 服务器	是，在单个服务器上允许多个虚拟会话	任何用户都可以启动和关闭会话	否，为每个虚拟会话运行一个专用的 X 服务器 (Xdcv)。屏幕是从 X 服务器中捕获的。	是，但需要使用 DCV-GL 软件包

Note

您无法同时在同一 NICE DCV 服务器上运行控制台会话和虚拟会话。

控制台会话

Windows 和 Linux NICE DCV 服务器上支持控制台会话。如果您使用 Windows NICE DCV 服务器，则只能使用控制台会话。

每次只能在 NICE DCV 服务器上托管一个控制台会话。控制台会话由管理员（在 Windows NICE DCV 服务器上）和根用户（在 Linux NICE DCV 服务器上）创建和管理。

对于控制台会话，NICE DCV 直接捕获桌面屏幕的内容。如果服务器配置了 GPU，NICE DCV 控制台会话可以直接访问该 GPU。

虚拟会话

仅 Linux NICE DCV 服务器上支持虚拟会话。

您可以同时在同一 NICE DCV 服务器上托管多个虚拟会话。虚拟会话是由 NICE DCV 用户创建和管理的。NICE DCV 用户只能管理他们创建的会话。根用户可以管理当前在 NICE DCV 服务器上运行的所有虚拟会话。

对于虚拟会话，NICE DCV 启动 X 服务器实例 `Xdcv`，并在 X 服务器中运行桌面环境。NICE DCV 为每个虚拟会话启动一个新的专用 X 服务器实例。每个虚拟会话均使用其 X 服务器实例提供的显示内容。

Note

虽然 NICE DCV 确保每个虚拟会话具有独立的 `Xdcv` 显示器，但很多其他系统资源（包括用户主文件夹中的文件、D-Bus 服务和设备）是按用户分配的，因此，可以在同一用户的多个虚拟会话之间共享和访问这些资源。

您不应同时在同一 NICE DCV 服务器上为同一用户运行多个虚拟会话，除非您已设置操作系统以减轻对共享资源的可能担忧。

如果已安装 `dcv-gl` 软件包并获得许可，NICE DCV 虚拟会话将共享对服务器 GPU 的访问。要在多个虚拟会话之间共享基于硬件的 OpenGL，必须通过配置 `dcv-gl.conf` 文件将虚拟 X 服务器实例连接到 GPU。

使用命令行工具管理 NICE DCV 会话

NICE DCV 服务器包含一个命令行工具，可用于启动、停止和查看 NICE DCV 会话。

在 Windows NICE DCV 服务器上使用命令行工具

要在 Windows NICE DCV 服务器上使用命令行工具，请从 NICE DCV 安装目录中运行命令或将 NICE DCV 目录添加到 PATH 环境变量中。如果将 NICE DCV 目录添加到 PATH 环境变量中，您可以从任何目录中使用命令。

从 NICE DCV 安装目录中使用命令行工具

导航到 `dcv.exe` 文件所在的文件夹（默认为 `C:\Program Files\NICE\DCV\Server\bin\`），然后打开命令提示符窗口。

或者，您可以在从其他目录运行命令时指定完整路径。

```
"C:\> Program Files\NICE\DCV\Server\bin\dcv.exe" list-sessions
```


将 NICE DCV 目录添加到 PATH 环境变量

1. 在文件资源管理器中，右键单击此 PC，然后选择属性。
2. 选择高级系统设置。
3. 在高级选项卡上，选择环境变量。
4. 在系统变量部分中，选择 Path 变量，然后选择编辑。
5. 选择新建并指定 NICE DCV 安装目录中的 bin 文件夹的完整路径（例如 C:\Program Files\NICE\DCV\Server\bin\）。
6. 选择确定并关闭“环境变量”窗口。

在 Linux NICE DCV 服务器上使用命令行工具

在 Linux NICE DCV 服务器上，在 \$PATH 环境变量中自动配置命令行工具。您可以从任何文件夹中使用该工具。打开终端窗口并输入要运行的命令。

命令行工具的使用

下表涵盖了可用的命令行工具选项。此列表可以通过在调用--help时使用来检索dcv。有关如何使用每条命令的更多信息，请在需要使用信息的命令--help后面传入。例如：`dcv create-session --help`。

命令	描述
<code>create-session</code>	创建新的 DCV 会话
<code>close-session</code>	关闭活动的 DCV 会话
<code>describe-session</code>	描述 DCV 会话
<code>list-sessions</code>	列出活跃的 DCV 会话
<code>list-connections</code>	列出 DCV 会话的客户端连接
	关闭活跃的客户端连接

命令	描述
close-connection	
get-screenshot	获取 DCV 控制台的屏幕截图
set-display-layout	设置活动 DCV 会话的显示布局
set-name	为 DCV 会话设置名称
set-permissions	设置活动 DCV 会话的权限
set-storage-root	设置活动 DCV 会话的存储根
reload-licenses	强制重新加载所有正在运行的会话的许可证
get-config	获取服务器配置
list-endpoints	列出 DCV 端点
set-config	设置服务器配置
version	显示 DCV 的版本
help	显示帮助

启动 NICE DCV 会话

在您使用默认设置[安装 Windows NICE DCV 服务器](#)时，将在安装服务器后自动创建并激活一个[控制台会话](#)。默认控制台会话由 Administrator 拥有，其默认会话 ID 为 console。您可以使用该会话，也可以[关闭该会话](#)并创建一个新会话。

如果您在安装 NICE DCV 服务器时选择不自动创建控制台会话，您必须手动创建一个会话。在安装 NICE DCV 服务器后，您可以随时启用或禁用[自动创建控制台会话](#)的功能。

Note

Linux NICE DCV 服务器不会在安装后获得默认控制台会话。

假设您在本地或其他基于云的服务器上使用浮动许可证，并且超过了许可证支持的最大并发会话数，您可能会出现 no licenses 错误。如果您收到此错误，请停止一个未使用的会话以释放许可证并重试。

NICE DCV 服务器必须正在运行才能启动会话。有关更多信息，请参阅[启动 NICE DCV 服务器](#)。

主题

- [手动启动控制台会话和虚拟会话](#)
- [启用自动控制台会话](#)

手动启动控制台会话和虚拟会话

您可以随时启动 NICE DCV 会话。一次只能运行一个控制台会话。如果使用 Linux NICE DCV 服务器，您可以同时运行多个虚拟会话。

最好在创建会话之前运行 `dcv list-sessions`，尤其是在使用 Windows NICE DCV 服务器时。

要在 Windows 或 Linux NICE DCV 服务器上创建控制台会话或虚拟会话，请使用 `dcv create-session` 命令。

主题

- [语法](#)
- [Options](#)
- [示例](#)

语法

启动会话的命令的最简单语法是：

```
dcv create-session session_ID
```

包含所有选项的完整语法是：

```
dcv create-session \  
  --type console|virtual \  
  --name session_name \  
  --user username \  
  --owner owner_name \  
  --permissions-file /path_to/permissions_file \  
  --storage-root /path_to/storage_directory \  
  --gl on/off \  
  --max-concurrent-clients number_of_clients \  
  --init /path_to/init_script \  
  session_ID
```

Note

\ 符号表示将命令拆分为多行的语法。

您也可以使用 `dcv create-session --help` 显示语法的快速参考。

Options

以下选项可以与 `dcv create-session` 命令一起使用：

--type

仅 Linux NICE DCV 服务器上支持此选项。它指定要创建的会话的类型，可以是 `console` 或 `virtual`。

类型：字符串

允许的值：`console | virtual`

必需：否

--name

指定会话的名称。会话名称可以是最多 256 个字符的任意字符串。如果字符串超过 256 个字符，命令将失败。会话名称在运行的会话中不需要是唯一的。

您可以随时使用 `dcv set-name` 命令更改会话的名称。有关更多信息，请参阅[管理会话名称](#)。

类型：字符串

必需：是

--user

仅在 Linux NICE DCV 会话上支持将此选项与虚拟会话一起使用。此值是要用于创建会话的用户。只有根用户可以模拟其他用户。

类型：字符串

必需：否

--owner

指定会话所有者。如果省略，则默认为当前已登录的用户。

类型：字符串

必需：否

--permissions-file

指定自定义权限文件的路径。如果省略，则默认为服务器默认值。

类型：字符串

必需：否

--storage-root

指定用于会话存储的文件夹的路径。

您可以使用 `%home%` 指定当前已登录的用户的主目录。例如，以下内容将会话存储的目录设置为 `c:\Users\username\storage\`（对于 Windows 服务器）或 `$HOME/storage/`（对于 Linux 服务器）。

```
--storage-root %home%/storage/
```

Note

如果指定的子目录不存在，则会禁用会话存储。

类型：字符串

必需：否

--gl

仅在 Linux NICE DCV 会话上支持将此选项与虚拟会话一起使用。它覆盖默认的 `dcv-gl` 状态，并且可以是 `on` 或 `off`。

类型：字符串

允许的值：`on` | `off`

必需：否

--max-concurrent-clients

指定允许连接到会话的 NICE DCV 客户端的最大数量。如果省略，则默认为无限连接。

类型：整数

必需：否

--init

仅在 Linux NICE DCV 服务器上支持将此选项与虚拟会话一起使用。它指定自定义 `init` 脚本的路径。该脚本可用于启动特定桌面环境以及在会话开始时自动启动特定应用程序。该脚本必须是可执行的。如果省略，则默认为启动默认桌面环境的脚本。

类型：字符串

必需：否

session ID

在命令末尾提供会话的 ID。

类型：字符串

必需：是

示例

示例 1 – 控制台会话

以下命令创建一个由 `dcv-user` 拥有的控制台会话，它具有唯一会话 ID `my-session` 和会话名称 `my graphics session`。它还指定一个名为 `perm-file.txt` 的权限文件。

- Windows NICE DCV 服务器

```
C:\> dcv create-session^
--owner dcv-user^
--name "my graphics session"^
--permissions-file perm-file.txt^
my-session
```

- Linux NICE DCV 服务器

```
$ sudo dcv create-session \
--type=console \
--owner dcv-user \
--name "my graphics session" \
--permissions-file perm-file.txt \
my-session
```

示例 2 - 虚拟会话 (仅限 Linux NICE DCV 服务器)

以下命令使用根用户模拟预期会话所有者 `dcv-user` 以创建一个虚拟会话。即使会话是由根用户创建的，该会话仍由 `dcv-user` 拥有：

```
$ sudo dcv create-session \
--owner dcv-user \
--user dcv-user \
my-session
```

示例 3 - 虚拟会话 (仅限 Linux NICE DCV 服务器)

以下命令创建一个虚拟会话，该会话由创建它的用户拥有：

```
$ dcv create-session my-session
```

启用自动控制台会话

启用自动控制台会话可以确保，每次 NICE DCV 服务器启动时自动创建控制台会话。自动控制台会话由 `owner` 配置参数指定的 NICE DCV 用户拥有。它的会话 ID 始终为 `console`。

影响自动控制台会话的其他参数是 `max-concurrent-clients`、`permissions-file` 和 `storage-root`。有关这些参数的更多信息，请参阅 [session-management/automatic-console-session](#) 参数。

Note

NICE DCV 不支持自动虚拟会话。

Windows NICE DCV server

在 Windows NICE DCV 服务器上启用自动控制台会话

1. 打开 Windows 注册表编辑器。
2. 导航到 HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/session-management 注册表项。
3. 创建 create-session 参数：
 - a. 在导航窗格中，打开 session-management 项的上下文（右键单击）菜单，然后选择新建 > DWORD（32 位）值。
 - b. 对于名称，输入 create-session 并按 Enter。
 - c. 打开 create-session 参数。对于值数据，输入 1 并选择确定。
4. 导航到 HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/session-management/automatic-console-session 注册表项。
5. 创建 owner 参数：
 - a. 在导航窗格中，打开 automatic-console-session 密钥的上下文（右键单击）菜单，然后选择“新建，字符串值”。
 - b. 对于名称，输入 owner 并按 Enter。
 - c. 打开 owner 参数。对于值数据，输入会话拥有者的名称并选择确定。
6. 选择确定，并关闭 Windows 注册表编辑器。
7. [停止并重新启动](#) NICE DCV 服务器。

Linux NICE DCV server

在 Linux NICE DCV 服务器上启用自动控制台会话

1. 导航到 /etc/dcv/，并使用您常用的文本编辑器打开 dcv.conf。
2. 使用以下格式将 create-session 和 owner 参数添加到 [session-management/automatic-console-session] 段：


```
[session-management]
create-session = true

[session-management/automatic-console-session]
owner="session-owner"
```

3. 保存并关闭文件。
4. [停止并重新启动](#) NICE DCV 服务器。

停止 NICE DCV 会话

控制台会话只能由管理员（在 Windows NICE DCV 服务器上）和根用户（在 Linux NICE DCV 服务器上）停止。Linux NICE DCV 服务器上的虚拟会话只能由根用户或创建它的 NICE DCV 用户停止。

Note

停止会话将关闭会话中正在运行的所有应用程序。

要停止 Windows 或 Linux NICE DCV 服务器上的控制台会话或虚拟会话，请使用 `dcv close-session` 命令并指定唯一的会话 ID。

主题

- [语法](#)
- [示例](#)

语法

```
dcv close-session session-id
```

示例

例如，以下命令停止具有唯一 ID `my-session` 的会话。

```
dcv close-session my-session
```

管理运行的 NICE DCV 会话

下一节提供了有关管理运行的 NICE DCV 会话的信息。

主题

- [管理 NICE DCV 会话存储](#)
- [管理 NICE DCV 会话授权](#)
- [管理 NICE DCV 会话显示布局](#)
- [管理会话名称](#)

管理 NICE DCV 会话存储

会话存储是 NICE DCV 服务器上的一个目录，客户端可以在连接到 NICE DCV 会话时访问该目录。

如果在 NICE DCV 服务器上启用了会话存储，您可以使用 `dcv set-storage-root` 命令指定服务器上用于会话存储的目录。有关在 NICE DCV 服务器上启用会话存储的更多信息，请参阅[启用会话存储](#)。

要设置会话存储路径，请使用 `dcv set-storage-root` 命令并指定会话 ID 和要使用的目录的路径。

主题

- [语法](#)
- [Options](#)
- [示例](#)

语法

```
dcv set-storage-root --session session_id /path_to/directory
```

对于目录路径，您可以使用 `%home%` 指定当前登录用户的主目录。例如，在 Windows 服务器上，`%home%/storage/` 路径解析为 `c:\Users\username\storage\`。在 Linux 服务器上，该路径解析为 `$HOME/storage/`。

Options

可以将以下选项与 `dcv set-storage-root` 命令一起使用：

--session

要指定存储目录的会话 ID。

类型：字符串

必需：是

示例

Windows NICE DCV 服务器示例

以下示例将会话 ID 为 `my-session` 的会话的存储路径设置为 `c:\session-storage`。

```
C:\> dcv set-storage-root --session my-session c:\session-storage
```

Linux NICE DCV 服务器示例

以下示例设置为当前用户的主目录中名为 `session-storage` 的目录的存储路径（对于会话 ID 为 `my-session` 的会话）。

```
$ dcv set-storage-root --session my-session %home%/session-storage/
```

管理 NICE DCV 会话授权

授权用于为 NICE DCV 客户端授予或拒绝特定 NICE DCV 功能的权限。通常，授权是在启动 NICE DCV 会话时配置的。不过，可以编辑运行的会话的权限。有关 NICE DCV 授权的更多信息，请参阅[配置 NICE DCV 授权](#)。

要修改运行的会话的权限，请使用 `dcv set-permissions` 命令。

主题

- [语法](#)
- [Options](#)

- [示例](#)

语法

```
dcv set-permissions --session session-id --none | --reset-builtin | --file /path_to/permissions_file
```

您必须指定 `--none`、`--reset-builtin` 或 `--file`。

Options

以下选项可以与 `dcv set-permissions` 命令一起使用。

`--session`

指定要设置权限的会话的 ID。

`--reset-builtin`

将会话的权限重置为默认会话权限。默认权限仅向会话拥有者授予对所有功能的完全访问权限。

`--none`

撤消对会话的所有权限。

`--file`

指定自定义权限文件的路径。如果指定的文件为空，则撤消所有权限。有关创建自定义权限文件的更多信息，请参阅[使用权限文件](#)。

示例

示例 1 - 撤销所有权限

以下示例撤消 ID 为 `my-session` 的会话的所有客户端权限。

```
C:\> dcv set-permissions --session my-session --none
```

示例 2 - 指定自定义权限

以下示例为具有 ID `my-session` 的会话指定名为 `perm-file.txt` 的自定义权限文件。此文件位于 `c:\dcv\` 目录中。

```
C:\> dcv set-permissions --session my-session --file c:\dcv\perm-file.txt
```

示例 3 - 重置权限

以下示例将 ID 为 `my-session` 的会话的权限重置为默认值。

```
C:\> dcv set-permissions --session my-session --reset-builtin
```

管理 NICE DCV 会话显示布局

您可以为运行的 NICE DCV 会话设置显示布局。显示布局指定客户端连接到会话时使用的默认配置。不过，客户端可以使用 NICE DCV 客户端设置或本机操作系统显示设置手动覆盖布局。

如果主机服务器的硬件和软件配置不支持指定的分辨率或屏幕数量，则 NICE DCV 服务器不会应用指定的显示布局。

NICE DCV 可以根据设置和服务器系统配置配置分辨率。

- 默认情况下，Web 客户端分辨率限制为 1920x1080（来自服务器 `web-client-max-head` 分辨率设置）。
- 默认情况下，本机客户端限制为 4096x2160（来自 `max-head-resolution`）。

请注意，可用的分辨率和显示器数量取决于服务器的配置，请务必按照[先决条件指南](#)正确设置系统环境和驱动程序以获得最佳性能。

Note

对于本机客户端，最多可以使用四台显示器。
对于 Web 客户端，最多可以使用两台显示器。
任何配置都不支持更高的分辨率或超过最大显示器数量。

主题

- [限制显示布局](#)
- [指定显示布局](#)
- [查看显示布局](#)

限制显示布局

您可以配置 NICE DCV 服务器，以防止客户端请求超出指定范围的显示布局。要限制显示布局更改，请配置以下 NICE DCV 服务器参数。

- [enable-client-resize](#) - 要禁止客户端更改显示布局，请将该参数设置为 `false`。
- [min-head-resolution](#) 和 [max-head-resolution](#) - 分别指定允许的最小和最大分辨率。
- [web-client-max-head-resolution](#) - 指定 Web 浏览器客户端允许的最大分辨率。`max-head-resolution` 限制基于 `web-client-max-head-resolution` 限制应用。默认情况下，Web 浏览器客户端的最大分辨率为 1920x1080。指定更高分辨率可能会导致性能问题，具体取决于 Web 浏览器和客户端计算机规范。
- [max-num-heads](#) - 指定最大显示器数。
- [max-layout-area](#) - 指定屏幕区域允许的最大像素数。将忽略总屏幕区域（以像素表示）超过指定值的请求。

有关这些参数的更多信息，请参阅“参数参考”中的[display 参数](#)。

指定显示布局

为运行的 NICE DCV 会话配置显示布局

使用 `dcv set-display-layout` 命令并指定要设置显示布局和显示布局描述符的会话。

```
dcv set-display-layout --session session-id display-layout-descriptor
```

显示布局描述符指定显示数以及每个显示的分辨率和位置偏移。必须按以下格式指定描述：

```
widthxheight+|-x-position-offset+|-y-position-offset
```

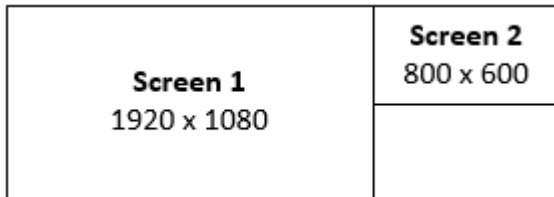
如果指定多个屏幕，请用逗号将屏幕描述符分隔开来。屏幕位置偏移指定屏幕左上角相对于屏幕 1 的位置。如果您没有指定屏幕的位置偏移，则默认为 `x=0` 和 `y=0`。

Important

如果要指定多个屏幕，请确保正确设置每个屏幕的位置偏移，以避免屏幕重叠。

例如，以下显示布局描述符指定两个屏幕：

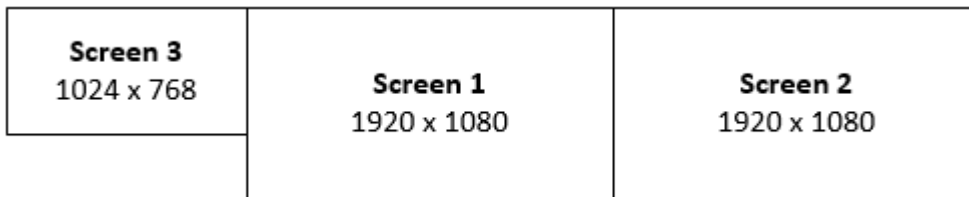
- 屏幕 1：1920x1080 分辨率，偏移为 $x=0$ ， $y=0$
- 屏幕 2：800x600 分辨率，偏移为 $x=1920$ ， $y=0$ ，以便它显示在屏幕 1 的右侧。



$1920 \times 1080 + 0 + 0, 800 \times 600 + 1920 + 0$

以下显示布局描述符指定三个屏幕。

- 屏幕 1：1920x1080 分辨率，偏移为 $x=0$ ， $y=0$
- 屏幕 2：1920x1080 分辨率，偏移为 $x=1920$ ， $y=0$ ，以便它显示在屏幕 1 的右侧。
- 屏幕 3：1024x768 分辨率，偏移为 $x=-1024$ ， $y=0$ ，以便它显示在屏幕 1 的左侧。



$1920 \times 1080 + 0 + 0, 1920 \times 1080 + 1920 + 0, 1024 \times 768 - 1024 + 0$

查看显示布局

查看会话的显示布局

使用 `dcv describe-session` 命令并查看输出中的 `display layout` 元素。有关更多信息，请参阅[查看 NICE DCV 会话](#)。

管理会话名称

您可以随时更改运行的会话的名称。您可以使用会话的具体名称，以根据会话名称快速找到会话。会话名称在运行的会话中不需要是唯一的。

要更改运行的会话的名称，请使用 `dcv set-name` 命令。

主题

- [语法](#)
- [Options](#)
- [示例](#)

语法

```
$ dcv set-name --session session_id --none | --name "session-name"
```

您必须指定 `--name` 或 `--none`。

Options

以下选项可以与 `dset-name` 命令一起使用。

--session

要设置名称的会话的 ID。

类型：字符串

必需：是

--name

要为会话分配的名称。只有在您要为会话分配名称时，才需要指定该选项。如果要删除名称，请省略该参数。会话名称长度最多为 256 个字符。它可以由字母、数字和特殊字符组成。如果指定的字符串超过 256 个字符，命令将失败。

类型：字符串

必需：否

--none

指定该参数以从会话中删除现有的名称。如果您不希望删除会话名称，请省略该选项。

必需：否

示例

示例 1 - 更改会话的名称

以下示例将具有 ID `my-session` 的会话的名称设置为 `my graphics session`。

```
$ dcv set-name --session my-session --name "my graphics sessions"
```

示例 2 - 删除会话的名称

以下示例删除具有 ID `my-session` 的会话的名称。

```
$ dcv set-name --session my-session --none
```

管理会话时区

DCV 允许会话所有者和用户设置其会话的时区，以反映 DCV 服务器位置或他们的当前位置。

启用时区重定向

您可以为特定会话上的所有用户启用和禁用该功能。

1. 将 [enable-timezone-redirection](#) 参数修改为以下值之一：

- `always-on`：始终启用时区重定向。

将开启该功能，并且会话显示客户端的时区信息。用户无法关闭该功能。

- `always-off`：始终禁用时区重定向。

将关闭该功能，并且会话显示自己的时区信息。用户无法开启该功能。

- `client-decides`：默认开启时区重定向。

会话将启用该功能，显示客户端时区，并且用户可以选择禁用该功能以允许显示服务器时区。

Note

该设置是标准默认设置。

Note

如果仅会话中的各个用户需要具有该功能，您需要先为所有用户设置集中参数，然后在 [添加权限](#) 中创建自定义权限文件以单独调整各个用户的权限。

2. 重新启动任何受影响的会话以使更改生效。

查看 NICE DCV 会话

Windows NICE DCV 服务器上的管理员或 Linux NICE DCV 服务器上的根用户可以查看服务器上运行的所有活动会话。NICE DCV 用户只能查看他们创建的会话。

主题

- [列出所有活动会话](#)
- [查看有关特定会话的信息](#)

列出所有活动会话

要列出 Windows 或 Linux NICE DCV 服务器上的活动控制台会话或虚拟会话，请使用 `dcv list-sessions` 命令。

主题

- [语法](#)
- [输出](#)

语法

```
dcv list-sessions
```

输出

该命令使用以下格式返回活动会话列表。

```
Session: session-id (owner:session-owner type:virtual/console name:'my session')
```

查看有关特定会话的信息

要查看有关会话的信息，请使用 `dcv describe-session` 命令并指定唯一的会话 ID。

主题

- [语法](#)

- [输出](#)

语法

```
$ dcv describe-session session_id
```

输出

在以下示例输出中，`display-layout` 元素指示会话的显示布局设置为使用两个 800x600 屏幕。其中的第二个屏幕从第一个屏幕偏移 `x=800`（向右）。

```
Session: test
  owner: session-id
  name: session-name
  x display: :1
  x authority: /run/user/1009/dcv/test.xauth
  display layout: 800x600+0+0,800x600+800+0
```

您还可以包含 `--json`（或 `-j`）选项，以强制命令以 JSON 格式返回输出。JSON 输出提供了有关会话的其他详细信息。

```
$ dcv describe-session session-id --json
```

以下是示例 JSON 输出。

```
{
  "id" : "session-id",
  "owner" : "dcvuser",
  "name" : "session-name",
  "num-of-connections" : 0,
  "creation-time" : "2020-03-02T16:08:50Z",
  "last-disconnection-time" : "",
  "licenses" : [
    {
      "product" : "dcv",
      "status" : "licensed",
      "check-timestamp" : "2020-03-02T16:08:50Z",
      "expiration-date" : "2020-03-29T00:00:00Z"
    },
    {
```

```
    "product" : "dcv-gl",
    "status" : "licensed",
    "check-timestamp" : "2020-03-02T16:08:50Z",
    "expiration-date" : "2020-03-29T00:00:00Z"
  }
],
"storage-root" : "",
"type" : "virtual",
"x11-display" : ":2",
"x11-authority" : "/run/user/1009/dcv/vsession.xauth",
"display-layout" : [
  {
    "width" : 800,
    "height" : 600,
    "x" : 0,
    "y" : 0
  },
  {
    "width" : 800,
    "height" : 600,
    "x" : 800,
    "y" : 0
  }
]
}
```

获取 NICE DCV 会话屏幕截图

您可以使用 `dcv get-screenshot` 命令获取运行的会话的桌面屏幕截图。

语法

```
dcv get-screenshot --max-width pixels --max-height pixels --format JPEG/PNG --primary
--json --output /path_to/destination session_name
```

Options

--max-width

指定屏幕截图的最大宽度（以像素为单位）。如果未指定宽度或高度，屏幕截图将使用会话的显示分辨率。如果仅指定高度，则自动扩缩宽度以保持纵横比。

类型：整数

必需：否

--max-height

指定屏幕截图的最大高度（以像素为单位）。如果未指定宽度或高度，屏幕截图将使用会话的显示分辨率。如果仅指定宽度，则自动扩缩高度以保持纵横比。

类型：整数

必需：否

--format

屏幕截图的文件格式。目前，仅支持 JPEG 和 PNG 格式。如果为 `--format` 和 `--output` 选项指定的文件类型发生冲突，则优先使用为 `--format` 指定的值。例如，如果您指定 `--format JPEG` 和 `--output myfile.png`，则 NICE DCV 创建 JPEG 图像文件。

类型：字符串

允许的值：JPEG | PNG

必需：否

--primary

指示是否仅获取主显示器的屏幕截图。要仅获取主显示器的屏幕截图，请指定 `--primary`。要获取所有显示器的屏幕截图，请省略该选项。如果您选择获取所有显示器的屏幕截图，则所有显示器将合并为单个屏幕截图。

必需：否

--json, -j

指示是否使用 JSON 格式提供 Base64 编码的输出。要获取 JSON 输出，请指定 `--json`。否则，省略该选项。

必需：否

--output, -o

指定屏幕截图的目标路径、文件名和文件类型。例如，对于 Windows，指定 `c:\directory\filename.format`；对于 Linux，指定 `/directory/filename.format`。格式必须是 `.png` 或 `.jpeg`。如果为 `--format` 和 `--output` 选项指定的文件类型发生冲突，则优先使用为 `--`

`format` 指定的值。例如，如果您指定 `--format JPEG` 和 `--output myfile.png`，则 NICE DCV 创建 JPEG 图像文件。

类型：字符串

必需：否

示例

示例 1

以下示例命令获取名为 `my-session` 的会话的屏幕截图。屏幕截图使用服务器的分辨率。

```
dcv get-screenshot --output myscreenshot.png my-session
```

示例 2

以下示例命令获取宽度为 200 像素且高度为 100 像素的屏幕截图。该示例从名为 `my-session` 的会话中获取屏幕截图。它将屏幕截图保存在当前目录中，文件名为 `myscreenshot.png`。

```
dcv get-screenshot --max-width 200 --max-height 100 --output myscreenshot.png my-session
```

示例 3

以下示例命令获取名为 `my-session` 的会话的屏幕截图。屏幕截图仅是主显示器的屏幕截图。它将文件保存在当前目录中，并将屏幕截图命名为 `myscreenshot.png`。

```
dcv get-screenshot --primary --output myscreenshot.jpeg my-session
```

示例 4

以下示例命令获取名为 `my-session` 的会话的屏幕截图。该命令使用 JSON 格式输出 Base64 编码的文件。

```
dcv get-screenshot --json --format png my-session
```

如何...

主题

- [使用外部身份验证](#)
- [查找并停止空闲会话](#)
- [启用到 X 服务器的远程 X 连接](#)
- [将 NICE DCV Web 浏览器客户端嵌入到 iFrame 中](#)

使用外部身份验证

默认情况下，NICE DCV 客户端身份验证委派给底层操作系统。对于 Windows NICE DCV 服务器，将身份验证委托给 WinLogon。对于 Linux NICE DCV 服务器，身份验证委派给 Linux PAM。

您可以将 NICE DCV 配置为使用外部身份验证服务器对客户端进行身份验证。这样，您就可以使用现有的身份验证系统了。对于外部身份验证，NICE DCV 利用您的现有登录机制，并将身份验证委派给外部身份验证服务器。

外部身份验证验证用户是否具有 DCV 服务器访问权限，从而决定是否允许使用会话创建功能。与系统身份验证不同，外部身份验证不会根据底层操作系统验证您的用户身份，除非您将自己的外部身份验证器设置为这样做。

[DCV Session Manager](#) 内置一个外部身份验证器。要使用此功能，您的 DCV 服务器需要使用会话管理器地址设置 [auth-token-verifier](#) 参数。

要使用外部身份验证服务器，您必须具备以下条件：

- 登录机制 - 这是用户登录时使用的前端机制。它应该能够使用现有的凭证验证系统验证您的用户，并且应该能够生成令牌并将其提供给 NICE DCV 服务器。有关更多信息，请参阅[使用令牌](#)。
- 身份验证服务器 - 这是对登录机制生成的令牌进行身份验证的服务器。该服务器应该能够从 NICE DCV 服务器接收包含令牌的 HTTP (S) POST 请求，执行必要的身份验证，然后将响应发回到 NICE DCV 服务器。有关实施身份验证服务器的更多信息，请参阅[身份验证服务要求](#)。
- NICE DCV 服务器配置 - NICE DCV 服务器必须配置为使用外部身份验证服务器。有关更多信息，请参阅[NICE DCV 服务器配置](#)。

主题

- [NICE DCV 服务器配置](#)
- [使用令牌](#)
- [身份验证服务要求](#)

NICE DCV 服务器配置

您必须将 NICE DCV 服务器配置为使用外部身份验证服务。

Linux NICE DCV server

在 Linux 上指定外部身份验证服务器

1. 导航到 `/etc/dcv/`，并使用您常用的文本编辑器打开 `dcv.conf`。
2. 找到 `[security]` 部分中的 `auth-token-verifier` 参数，并将现有值替换为外部身份验证服务器的 URL 和用于通信的端口，格式如下所示：`url:port`。例如，如果您使用的是 `DcvSimpleExternalAuthenticator`，请指定以下内容：`http://127.0.0.1:8444`。

如果 `auth-token-verifier` 部分中没有 `[security]` 参数，则使用以下格式手动添加此参数：

```
[security] auth-token-verifier=url:port
```

3. 保存并关闭文件。

Windows NICE DCV server

在 Windows 上指定外部身份验证服务器

1. 打开 Windows 注册表编辑器。
2. 导航到 `HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/` 项。
3. 在[安全auth-token-verifier参数中找到该参数](#)。
4. 请执行以下操作之一：
 - 对于值数据，输入外部身份验证服务器的 URL 和用于通信的端口，格式如下所示：`url:port`。

Example

例如，如果您使用的是 `DcvSimpleExternalAuthenticator`，请指定以下内容：`http://127.0.0.1:8444`。

- 如果安全部分中没有 `auth-token-verifier` 参数，请将其添加到 PowerShell。请参阅 [修改配置参数](#)。
5. 关闭 Windows 注册表编辑器。
 6. [停止并重新启动](#) NICE DCV 服务器。

使用令牌

在生成令牌后，您必须能够将其发送到 NICE DCV 服务器。对于 Web 浏览器客户端，将令牌附加到连接 URL 中，如下所示：

```
https://server_hostname_or_IP:port?authToken=token#session_id
```

例如：

```
https://my-dcv-server.com:8443/?authToken=1234567890abcdef#my-session
```

身份验证服务要求

您的自定义身份验证服务可以在 NICE DCV 服务器的同一主机上运行，也可以在单独的主机上运行。身份验证服务必须侦听来自 NICE DCV 服务器的 HTTP (S) POST 请求。

下面显示了 NICE DCV 服务器使用的 POST 请求格式。

```
POST / HTTP/1.1
Content-Type: application/x-www-form-urlencoded
sessionId=session_id&authenticationToken=token&clientAddress=client_address
```

您的身份验证服务负责确定提供的令牌是否有效。

在验证令牌后，身份验证服务器必须将响应返回到 NICE DCV 服务器。响应正文必须包含以下内容之一，具体取决于身份验证过程的结果：

- 如果身份验证成功，身份验证服务将返回结果 `yes` 和用户标识符。例如：

```
<auth result="yes"><username>username</username></auth>
```

- 如果身份验证失败，身份验证服务将返回结果 no。例如：

```
<auth result="no"><message>message</message></auth>
```

DcvSimpleExternalAuthenticator

NICE DCV 附带一个名为 `DcvSimpleExternalAuthenticator` 的参考外部身份验证服务器。 `DcvSimpleExternalAuthenticator` 是一个 Python 脚本，您可以使用它作为创建自己的自定义身份验证服务器的起点。

`DcvSimpleExternalAuthenticator` 服务器支持 HTTP 和 HTTPS，并且必须运行在安装了 NICE DCV 服务器的同一台服务器上。默认情况下，`DcvSimpleExternalAuthenticator` 监听端口 8444 上的请求。如果需要，您可以更改端口。为此，请使用常用的文本编辑器打开 `/etc/dcv/simpleextauth.conf`，找到 `EXTAUTH_PORT` 参数，然后将现有值替换为所需的端口号。

要使用 `DcvSimpleExternalAuthenticator`，必须安装 `nice-dcv-simple-external-authenticator` 软件包。有关更多信息，请参阅 [安装 NICE DCV 服务器](#)。

使用简单外部身份验证器

1. 导航到您的身份验证目录。

```
sudo mkdir -p /var/run/dcvsimpleextauth
```

2. 生成您的身份验证令牌。

Example

在该示例中，123456 是示例身份验证器令牌，`session-123` 是示例会话 ID，`username` 是用户。

```
echo "123456" | sudo dcvsimpleextauth add-user --session session-123 --auth-dir /var/run/dcvsimpleextauth/ --user username -append
```

3. 启动您的服务器。

```
sudo dcvsimpleextauth --port 8444 --auth-dir /var/run/dcvsimpleextauth/ start-server
```

4. 在服务器运行后，测试配置以进行验证。

Example

再次使用该示例，测试将按以下方式运行：

```
curl -k http://localhost:8444 -d sessionId=session-123 -d authenticationToken=123456
```

如果成功，您将收到身份验证结果 `yes`。

查找并停止空闲会话

您可以使用具有 `-j` 命令选项的 `dcv describe-sessions` CLI 命令确定空闲的 NICE DCV 会话。指定 `-j` 选项会将此命令配置为返回 JSON 格式的输出，其中包含有关会话的其他详细信息。

例如，以下命令返回有关名为 `my-session` 的会话的信息。

```
$ dcv describe-session my-session -j
```

输出：

```
{
  "id" : "my-session",
  "owner" : "dcvuser",
  "x11-display" : ":1",
  "x11-authority" : "/run/user/1009/dcv/test3.xauth",
  "num-of-connections" : 1,
  "creation-time" : "2019-05-13T13:21:19.262883Z",
  "last-disconnection-time" : "2019-05-14T12:32:14.357567Z",
  "licensing-mode" : "DEMO",
  "licenses" : [
    {
      "product" : "dcv",
      "status" : "LICENSED",
      "check-timestamp" : "2019-05-14T12:35:40Z",
      "expiration-date" : "2019-05-29T00:00:00Z"
    },
    {
      "product" : "dcv-gl",
      "status" : "LICENSED",
```

```
        "check-timestamp" : "2019-05-14T12:35:40Z",
        "expiration-date" : "2019-05-29T00:00:00Z"
    }
]
}
```

在命令输出中，`num-of-connections` 参数表示活动客户端连接的数量。值 `0` 表示没有活动的客户端连接，并且会话当前处于空闲状态。您还可以使用 `last-disconnection-time` 参数确定会话上次具有活动客户端连接的时间。

您可以创建脚本或 `cron` 作业来使用此信息标识空闲会话。然后，您可以使用 [dcv close-session](#) 命令停止使用这些会话。

Note

停止会话将关闭会话中正在运行的所有应用程序。

启用到 X 服务器的远程 X 连接

默认情况下，由于存在固有的安全风险，`Xdcv` 禁止使用 X 转发。NICE DCV 从较新版本的 Xorg 服务器继承该行为。NICE DCV 服务器实施以下默认缓解措施以最大限度降低安全风险：

- X 服务器禁止来自网络的 X 连接。X 服务器配置为使用 `-nolisten tcp` 命令行选项启动。但是，可以更改默认行为，以启用到 X 服务器的远程 X 连接。有关变通方法的更多信息，请参阅[启用到 X 服务器的远程 X 连接](#)。
- X 服务器会禁用 GLX 间接上下文。由于与 DCV-GL 存在冲突，因此，目前没有变通办法来启用 GLX 间接上下文。

有关安全风险以及缓解措施的更多信息，请参阅[X.Org 安全公告](#)。

启用到 X 服务器的远程 X 连接

默认情况下，`Xdcv` 配置为通过 `-nolisten tcp` 命令行选项来启动以降低安全风险。但是，可以更改默认行为，以启用 X 转发。

启用 X 转发

使用所需的文本编辑器打开 `/etc/dcv/dcv.conf`。将以下内容添加到该文件的末尾：

- 通过 IPv4 和 IPv6 启用 X 转发

```
[session-management]
virtual-session-xdcv-args="-listen tcp"
```

- 仅通过 IPv4 启用 X 转发

```
[session-management]
virtual-session-xdcv-args="-listen tcp -nolisten inet6"
```

Note

启用 X 转发不会影响现有会话，而只会影响在启用它后启动的新会话。

测试 X 转发

1. 连接 NICE DCV 会话。
2. 确认 NICE DCV 服务器正在侦听 6000-6063 范围内的端口。

```
$ netstat -punta | grep 600
```

3. 将远程服务器添加到 NICE DCV 服务器主机访问列表中。

```
$ xhost +remote_server
```

4. 检索 NICE DCV 会话显示编号。

```
$ dcv describe-session session_name | grep display
```

5. 通过 SSH 连接到托管该应用程序的远程服务器。

```
$ ssh user@remote_server
```

6. 从远程服务器中，导出 display 环境变量以指向 NICE DCV 会话的 X 服务器。

```
$ export DISPLAY=dcv_server_ip:display_number
```

7. 从该远程服务器上，运行一个应用程序来测试 X 转发功能。例如：

```
xterm
```

测试应用程序（此处为 xterm）应显示在 NICE DCV 服务器的桌面环境中。

将 NICE DCV Web 浏览器客户端嵌入到 iFrame 中

默认情况下，为了防止点击劫持攻击，NICE DCV 不允许将 Web 浏览器客户端嵌入到 iFrame 中。不过，您可以覆盖该默认行为，以允许 Web 浏览器客户端在 iFrame 中运行。

有关防止点击劫持攻击的更多信息，请参阅 [Content Security Policy Cheat Sheet](#)。

要允许 Web 浏览器在 iFrame 中运行，您必须将 NICE DCV 服务器配置为向 Web 浏览器客户端发送以下额外的 HTTP 响应标头：

- web-x-frame-options
- web-extra-http-headers

我们建议您添加这两个标头，以确保 Web 浏览器之间的最佳兼容性。

Note

如果通过 NICE DCV 连接网关进行连接，则需要在网关配置中定义 x 帧选项。这是通过使用网关配置 [\[web-resources\]](#) 部分中的 local-resources-http-headers 参数来完成的。

Windows server

1. 打开 Windows 注册表编辑器，然后导航到 HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/connectivity/ 项。
2. 打开 web-x-frame-options 参数。对于值数据，输入 "ALLOW-FROM *https://server_hostname*"。

Note

如果该参数不存在，则创建一个新的字符串参数并将其命名为 web-x-frame-options。

3. 打开web-extra-http-headers参数。对于值数据，输入 [("Content-Security-Policy", "frame-ancestors https://*server_hostname*")]

Note

如果该参数不存在，则创建一个新的字符串参数并将其命名为 web-extra-http-headers。

4. 关闭 Windows 注册表编辑器。
5. [停止并重新启动](#) NICE DCV 服务器。

Linux server

1. 使用首选文本编辑器打开 /etc/dcv/dcv.conf。
2. 请在 [connectivity] 部分执行以下操作：
 - 对于 web-x-frame-options，输入 "ALLOW-FROM https://*server_hostname*"。
 - 对于 web-extra-http-headers，输入 [("Content-Security-Policy", "frame-ancestors https://*server_hostname*")]

例如：

```
[connectivity]
web-x-frame-options="ALLOW-FROM https://my-dcv-server.com"
web-extra-http-headers=[("Content-Security-Policy", "frame-ancestors https://my-dcv-server.com")]
```

3. 保存并关闭文件。
4. [停止并重新启动](#) NICE DCV 服务器。

默认情况下，大多数浏览器禁止访问某些功能，例如麦克风访问和全屏访问。要允许访问这些功能，请修改网页上的 iFrame 元素。例如，要允许访问麦克风和全屏模式，请按以下方式修改 iFrame 元素：

```
<iframe src="..." allow="microphone; fullscreen">/iframe>
```

NICE DCV 故障排除

本章介绍了如何找出和解决 NICE DCV 可能遇到的问题。

主题

- [使用日志文件](#)
- [排除在 Linux 上创建虚拟会话时发生的故障](#)
- [在更改 UID 后，Linux 会话无法启动](#)
- [修复 Windows 上的光标问题](#)
- [修复复制并粘贴到 IntelliJ IDEA 的问题](#)
- [自签名证书的重定向说明](#)
- [Windows 上的 NVIDIA GPU 发生多显示器/全屏故障](#)
- [监控 NICE DCV 性能和统计信息](#)

要获得其他支持，请使用任何以下资源。

- 如果您是 NICE DCV 本地客户并需要其他帮助，请与您的 NICE DCV 经销商联系。
- 如果在 Amazon EC2 上使用 NICE DCV，您可以在 [AWS Support](#) 中创建一个支持工单。
- 如果您没有 AWS 支持计划，您可以在 [AWS re:Post](#) 上发布您的问题以寻求 NICE DCV 社区的帮助。

使用日志文件

可以使用 NICE DCV 日志文件找出和解决 NICE DCV 服务器问题。可以在 NICE DCV 服务器上的以下位置中找到 NICE DCV 日志文件：

- Windows 服务器

```
C:\ProgramData\NICE\dcv\log\server.log
```


Note

默认情况下，ProgramData 文件夹可能会隐藏。如果未看到 ProgramData 文件夹，请将文件浏览器设置为显示隐藏项目。或者，在地址栏中输入 %programdata%，然后按 Enter 键。

Linux 服务器

```
/var/log/dcv/server.log
```

NICE DCV 服务器允许您配置日志文件的详细程度。提供了以下详细程度等级：

- error - 提供最少的详细信息。仅包括错误。
- warn - 包括错误和警告。
- info - 默认详细程度。包括错误、警告和信息消息。
- debug - 提供最多的详细信息。提供有助于调试问题的详细信息。

在 Windows 上更改日志文件详细程度

要配置日志文件详细程度，您必须使用 Windows 注册表编辑器配置 level 参数。

在 Windows 上更改日志文件详细程度

1. 打开 Windows 注册表编辑器。
2. 导航到 HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/log/ 注册表项。
3. 通过双击打开 level 参数。对于数值数据，键入 error、warn、info 或 debug，具体取决于所需的详细程度等级。
4. 选择确定，并关闭 Windows 注册表编辑器。

在 Linux 上更改日志文件详细程度

要配置日志文件详细程度，您必须配置 level 文件中的 dcv.conf 参数。

在 Linux 上更改日志文件详细程度

1. 导航到 `/etc/dcv/`，并使用您常用的文本编辑器打开 `dcv.conf`。
2. 找到 `level` 部分中的 `[log]` 参数，并将现有详细程度等级替换为 `error`、`warn`、`info` 或 `debug`。

```
[log]
level="verbosity_level"
```

3. 保存并关闭该文件。

排除在 Linux 上创建虚拟会话时发生的故障

主题

- [调查在 Linux 上创建虚拟会话失败的问题](#)
- [在 Linux 上创建故障防护虚拟会话](#)

如果连接到虚拟会话导致 `No session available` 或 `The sessionId session is not available` 错误，这可能是由于虚拟会话创建失败并被终止。

您可以使用 `dcv list-sessions` 命令检查会话是否存在。有关检查运行的会话的更多信息，请参阅[the section called “查看会话”](#)。如果该会话未出现在列表中，则它可能已失败。

调查在 Linux 上创建虚拟会话失败的问题

使用以下命令在 Linux 上[创建](#)一个虚拟会话：

```
$ dcv create-session session
```

只有在会话创建失败时，该命令才会返回错误。不过，可能会出现以下情况：最初成功创建会话，但会话在用户连接之前终止。您可能会注意到这一点，因为在您检查现有会话时（例如，使用 `dcv list-sessions` 命令或 `dcv describe-session session`），您可能找不到列出的会话。

大多数情况下，发生这种情况是因为已创建桌面会话，但会话立即失败，例如，由初始化脚本启动的应用程序之一崩溃或失败，或者缺少所需的工具之一。

如果会话创建失败，请检查以下内容：

- 检查 `/var/log/dcv/sessionlauncher.log` 文件，其中包含与创建新会话进程的 dcv 组件相关的日志。
- 检查 `/var/log/dcv/dcv-session.user.session.log` 文件，其中包含与 dcv 初始化脚本相关的日志。
- 检查与会话所有者对应的主目录中的 `$HOME/.xsession-errors` 文件。该文件包含系统 X 会话初始化脚本生成的日志，通常包含桌面 Session Manager 或脚本调用的其他应用程序生成的日志。
- 检查系统日志以获取有关故障系统和组件的更多信息。首先，检查 `dmesg` 输出（例如，如果进程失败）和 `journalctl -xe` 输出。
- [使用故障防护会话进行测试](#)，以确认该问题与使用的 Session Manager 无关。

如果仅特定用户发生故障，您也可以尝试执行以下操作：

- 检查用户配置，特别是删除或重命名用户配置时发生什么情况。

根据桌面环境和版本，配置目录可能是用户目录中的 `.gnome`、`.kde` 或 `.config`。

- 检查影响用户 PATH 或环境的特定用户配置。通常，特定用户的会话启动失败是由于框架（例如 `anaconda`）覆盖某些标准本机命令造成的，这可能会导致会话初始化中的 `dbus` 连接失败。
- 检查权限问题。为本地 `~/.dbus` 或 `~/.Xauthority` 设置的错误权限（例如，它们可能由 `root` 而不是用户拥有）可能导致桌面会话立即终止。

在 Linux 上创建故障防护虚拟会话

要验证会话创建失败是否与桌面环境启动相关，一种常见的策略是创建最小会话。我们将该会话称为“故障防护”会话。如果创建的故障防护会话正常工作，我们可以推断您的正常会话失败是由于默认系统桌面环境无法启动。相反，如果故障防护会话也失败，则问题更有可能与 NICE DCV 服务器设置有关。

故障防护会话通常由仅包含简单窗口管理器和终端的桌面会话组成。这样，用户就可以检查是否存在与使用的特定会话环境（通常是 `gnome` 或 `KDE`）相关的会话创建问题。

要创建故障防护会话，您需要为用户创建一个初始化脚本，其中包含以下内容：

```
#!/bin/sh
metacity &
xterm
```

这会启动 metacity 窗口管理器并启动 xterm 终端，在终止 xterm 进程时，会话也会立即终止。

您可以使用所选的其他 Session Manager 或终端，但前提是它们在系统上可用。

Note

您必须确保脚本不会立即终止。为此，您需要在脚本结束时启动一个不会立即终止的程序。在终止最后一个命令（该示例中的 xterm）时，也会终止初始化会话。同时，在窗口管理器之后启动另一个工具时，您需要确保它在后台运行（通过添加该示例中的 & ），以确保调用下一个命令。

然后，您需要确保可以执行初始化脚本：

```
$ chmod a+x init.sh
```

要从用户 Shell 中使用指定的初始化脚本创建会话，请运行以下命令，其中 `init.sh` 是以前创建的脚本：

```
$ dcv create-session dummy --init init.sh
```

要以超级用户身份为另一个用户创建会话，您可以改为运行以下命令：

```
$ sudo dcv create-session test --user user --owner user --init init.sh
```

最后，您可以启动一个测试应用程序，例如 `dcvgltest`（仅在安装了 `nice-dcv-glttest` 软件包时）或 `glxgears`，以验证 OpenGL 或任何其他应用程序是否正常工作。

在更改 UID 后，Linux 会话无法启动

在 Linux 主机上，如果更改用户的用户 ID (UID) 或使用修改用户 UID 的不同 Active Directory 配置，可能会导致在主机上启动 NICE DCV 会话失败。

该问题的原因是，未授权使用新 UID 运行的 DCV 会话进程访问仍保留以前 UID 的文件和文件夹。具体而言：

- NICE DCV 日志目录中的 [日志文件](#)

- 用户的主文件夹

该问题同时影响控制台会话和虚拟会话。

要解决该问题，请确保用户的主文件夹及其包含的文件具有正确的 UID，并删除具有以前 UID 的旧 [NICE DCV 日志文件](#)。

修复 Windows 上的光标问题

对于在 Windows Server 2012 或 Windows 10 和更高版本上运行的 NICE DCV 服务器，鼠标光标始终显示为箭头。甚至在文本输入字段或单击导航项上暂停时，也会出现此情况。如果没有连接到服务器的物理鼠标，或者设备管理器中未列出鼠标设备，则可能会发生这种情况。

解决问题

1. 打开“控制面板”，然后选择轻松访问中心。
2. 选择使鼠标更易于使用。
3. 选择打开鼠标键。
4. 选择应用、确定。

修复复制并粘贴到 IntelliJ IDEA 的问题

在尝试将文本从 macOS NICE DCV 客户端复制到 IntelliJ IDEA 时，无法粘贴文本。IntelliJ 无法接受 NICE DCV 默认使用的跨平台格式。要在 NICE DCV 上禁用跨平台文本以将文本粘贴到 IntelliJ 中，请在 NICE DCV 服务器上修改 `disabled-targets` 字段。

该更改将导致在 NICE DCV Web Client 中无法使用复制和粘贴。在进行该更改之前，请确保您希望仅在 NICE DCV 客户端上进行 IntelliJ IDEA 复制和粘贴。

配置服务器以将文本粘贴到 IntelliJ IDEA

1. 导航到 `/etc/dcv/`，并使用您常用的文本编辑器打开 `dcv.conf`。
2. 在 `[clipboard]` 部分中找到 `disabled-targets` 参数。如果没有 `disabled-targets` 或 `[clipboard]` 部分，请手动添加这些部分。
3. 添加以下内容以定义 `disabled-targets` 值。

```
[clipboard]
```

```
disabled-targets = ['dcv/text', 'JAVA_DATAFLAVOR:application/x-java-jvm-local-objectref; class=com.intellij.codeInsight.editorActions.FoldingData']
```

4. 保存并关闭该文件。
5. [停止并重新启动](#) NICE DCV 会话。

自签名证书的重定向说明

从基于 Web 的门户或应用程序重定向到 NICE DCV 会话时，如果自签名证书以前不受信任，该证书可能会破坏浏览器与会话的信任关系。以下是一个发生这种情况的示例：

1. 用户连接到从中加载应用程序的公司门户网站。
2. 该应用程序尝试使用自签名证书打开与 NICE DCV 服务器的直接安全连接。
3. 浏览器拒绝该安全连接，因为证书是自签名的。
4. 用户看不到远程服务器，因为未建立连接。

信任问题是步骤 3 特定的。在用户使用自签名证书连接到网站（例如，导航到 <https://example.com>）时，浏览器要求信任该证书。不过，通过 HTTP 或 HTTPS 提供服务的 Web 应用程序/页面尝试建立到 DCV 服务器的安全 WebSocket 连接。如果证书是自签名的，浏览器检查它以前是否受信任。如果证书以前不受信任，则会拒绝连接，而不会向用户发出请求，提示用户是否要信任该证书。

这种情况的可能解决方案：

- 具有 DCV 服务器计算机的有效证书 - 如果企业在其计算机中使用自定义域。对于证书，他们可以为 DCV 分发企业证书。

Example

用户 ---[有效证书]---> DCV 服务器实例

- 在代理/网关中保护一组 DCV 服务器。仅在这种情况下，代理/网关需要具有有效的证书，并且 DCV 服务器实例可以保留其自签名证书。对于该选项，他们可以使用 [DCV Connection Gateway](#)、ALB/NLB 或其他代理解决方案。

Example

用户/Cx ---[此处，我们需要有效的证书]---> 代理/网关---[自签名证书]---> DCV 服务器实例

- 在通过 [SDK](#) 启动连接之前，让用户信任自签名证书。只需在另一个选项卡/窗口/弹出窗口中打开该 URL，即可实现该目的：<https://example.com/version>。

Note

/version 终端节点在 HTTPS 连接中使用简单的网页进行响应，以表示 DCV 服务器版本。

以后，可以在实际 DCV 服务器连接中使用相同的自签名证书。

Windows 上的 NVIDIA GPU 发生多显示器/全屏故障

如果 Windows 服务器主机具有 NVIDIA GPU，DCV 全屏/多显示器功能可能会失败。在发生这种情况时，显示器将拒绝进入全屏模式，或者服务器无法配置具有多个远程显示器的显示布局。

该问题的原因是，与 NVIDIA 驱动程序集成失败。

可以查看服务器主机上的 C:\ProgramData\NICE\dcv\log\ 以找出该问题，该问题报告以下错误：

```
WARN display - Cannot change display layout
```

这会显示多次 (20 - 30)，然后显示以下内容：

```
EDID not set on output x gpu x after attempt x INFO DLMNVAPI:display -  
Unable to set EDID on output x, gpu x: NVAPI_ERROR (-1)
```

在重现该问题时，主机未正常运行：服务器无法一致地配置多显示器布局，并且没有可行的方法以永久修复该问题（只有少数临时性缓解措施）。

该问题的触发因素是，在使用多显示器时执行服务器操作系统重新引导，即，在关闭服务器主机时，主机上存在虚拟显示器。因此，为了避免该问题，需要在关闭服务器之前删除服务器端的所有显示器。可以使用以下命令（以管理员权限执行）以确保这一点：

```
C:\Program Files\NICE\DCV\Server\bin\dcvnvedid.exe --remove
```

可能的缓解措施是，重新安装或更新 Nvidia 驱动程序并重新引导主机。

监控 NICE DCV 性能和统计信息

从 NICE DCV 2023.1 服务器开始，您可以使用 Windows 性能计数器监控协议性能的各个方面，并收集有关 NICE DCV 会话和连接的统计信息。

用于收集性能计数器的工具：

- [性能监视器 \(PerfMon\)](#)：一种 Windows 本机工具，可用于实时或从日志文件中可视化性能数据。
- [LogMan](#)：一种命令行工具，可以根据指定的条件启动和停止日志记录。
- [TypePerf](#)：一种命令行工具，它将性能数据写入到命令窗口或日志文件中。
- [PowerShell](#)：Windows 脚本语言，可用于收集和處理性能数据。
- 第三方工具：可以使用多种第三方监控解决方案收集这些计数器并提供深入见解。

DCV 性能计数器分为 5 个计数器集。

计数器集

NICE DCV 服务器

该计数器集包含有关主机上的 DCV 服务器服务的全局统计信息。它还包含很多计数器的汇总变体（这些计数器还位于其他计数器集中），从而提供一种方法以访问整个服务器生命周期的汇总信息，并具有静态路径（您不必检索会话或连接标识符，即可读取该计数器集中的计数器）。

Note

来自其他计数器集之一的汇总实例（例如“\DCV Server Connections(_Total)\Sent Bytes”）返回所有活动连接的总和，而全局计数器是在启动服务器后的累积信息，并包括已关闭的连接。

计数器名称	描述	单位	注释
Active Sessions	主机上的活动会话数	计数	
Total Sessions	在主机上创建的不断增加的会话数，包括已关闭的会话	计数	
Active Connections	到服务器的活动连接数	计数	
Total Connections	不断增加的服务器连接数，包括活动客户	计数	

计数器名称	描述	单位	注释
	端、重新连接的客户端和断开连接的客户端		
Idle Disconnections	由于不活动而断开连接的不断增加的连接数	计数	
Receive Rate bits/sec	服务器接收数据的速率 (以位/秒为单位)	位/秒	
Received Bytes	在启动服务后收到的总字节数	字节	
Send Rate bits/sec	服务器发送数据的速率 (以位/秒为单位)	位/秒	
Sent Bytes	在启动服务后发送的总字节数	字节	
HTTP Download Rate bits/sec	传出 HTTP 流量的带宽 (以位/秒为单位)	位/秒	文件存储的客户端到服务器流量计入接收速率
HTTP Downloaded Bytes	在启动服务后通过 HTTP 发送的总字节数	字节	文件存储的客户端到服务器流量计入接收字节数
Round-Trip Time ms	服务器和客户端之间的平均往返延迟 (以毫秒为单位)	毫秒	每 5 秒测量并更新一次
Minimum Round-Trip Time ms	在启动服务器后检测到的最小往返延迟 (以毫秒为单位)	毫秒	每 5 秒更新一次

DCV 服务器进程

该计数器集包含有关各个 NICE DCV 进程的信息。

agent_type can be one of: session_agent, system_agent, user_agent

每秒更新一次计数器。

计数器名称	描述	单位	注释
% Processor Time	进程使用的处理器时间百分比	百分比	百分比相对于一个 CPU 核心 (即 100% 表示进程占用一个线程)。 与 \Process(NAME)\% Processor Time 相同
Physical Memory Bytes	进程当前使用的物理内存量 (以字节为单位)	字节	与 \Process(NAME)\Working Set 相同
Virtual Memory Bytes	进程的虚拟地址空间的当前大小 (以字节为单位)	字节	
Process Identifier	数字进程标识符 (PID)	-	

NICE DCV 服务器会话

该集合中的计数器提供有关单个会话的信息。无论是否连接了用户，每个创建的会话都有一个该计数器集的实例。

如果管理员关闭会话，则会删除相应的实例；如果管理员重新创建具有相同名称的会话，所有计数器将从零重新开始。

计数器名称	描述	单位
Session Duration sec	会话已打开的总秒数	秒

计数器名称	描述	单位
Total Pixels	显示区域中的像素数，它是会话中的所有显示器的像素数之和	像素
Display Count	会话中的显示器数	计数

以下计数器与 NICE DCV 服务器计数器集中的计数器相同，仅描述略有差异：

计数器名称	描述
Active Connections	会话实例的活动连接数
Total Connections	到会话实例的不断增长的连接数，包括活动客户端、重新连接的客户端和断开连接的客户端
Idle Disconnections	由于不活动而断开的不断增长会话实例连接数
Ungraceful Disconnections	由于错误而断开的不断增长会话实例连接数
Receive Rate bits/sec	在会话中接收数据的速率（以位/秒为单位）
Received Bytes	在会话启动后收到的总字节数
Send Rate bits/sec	在会话中发送数据的速率（以位/秒为单位）
Sent Bytes	在会话启动后发送的总字节数
HTTP Download Rate bits/sec	会话中的传出 HTTP 数据的带宽（以位/秒为单位）
HTTP Downloaded Bytes	在会话中通过 HTTP 发送的总字节数
Round-Trip Time ms	会话中的服务器和客户端之间的平均往返延迟（以毫秒为单位）
Minimum Round-Trip Time ms	在建立会话后检测到的最小往返延迟（以毫秒为单位）

NICE DCV 服务器连接

该集合中的计数器提供有关单个客户端连接的信息。计数器集实例是在客户端连接到服务器时创建的，并在客户端断开连接时将其删除。connection_id 是一个数字，并且仅在一个服务器会话中是唯一的。

计数器名称	描述	单位
Connection Duration sec	连接已打开的总秒数	秒

以下计数器与“DCV 服务器”计数器集中的计数器相同，仅描述略有差异：

计数器名称	描述
Receive Rate bits/sec	在连接中接收数据的速率（以位/秒为单位）
Received Bytes	在建立连接后收到的总字节数
Send Rate bits/sec	在连接中发送数据的速率（以位/秒为单位）
Sent Bytes	在建立连接后发送的总字节数
HTTP Download Rate bits/sec	连接中的传出 HTTP 数据的带宽（以位/秒为单位）
HTTP Downloaded Bytes	在建立连接后通过 HTTP 发送的总字节数
Round-Trip Time ms	连接的平均往返延迟（以毫秒为单位）
Minimum Round-Trip Time ms	在建立连接后检测到的最小往返延迟（以毫秒为单位）

NICE DCV 服务器通道

该集合中的计数器提供有关客户端连接中的各个通道的信息。可能具有额外的扩展通道。

通道名称为：

- dcv::main
- dcv::display

- `dcv::input`
- `dcv::audio`
- `dcv::filestorage`
- `dcv::clipboard`

传入文件存储流量归类为 `dcv::filestorage` 通道。

传出文件存储流量包含在 DCV 服务器连接的 HTTP Download 计数器中。

Note

该集合中的计数器是 DCV 服务器连接中的计数器子集。

计数器名称	描述
Receive Rate bits/sec	通过通道接收数据的速率 (以位/秒为单位)
Received Bytes	通过通道收到的总字节数
Send Rate bits/sec	通过通道发送数据的速率 (以位/秒为单位)
Sent Bytes	通过通道发送的总字节数

NICE DCV 服务器图像

该集合中的计数器提供有关负责屏幕抓取、编码和传输的子系统的信息。

该集合中的计数器分为两组：

- 对于第一组中的计数器，NICE DCV 为每个会话收集一个值，并在 `$session_name` 实例中发布该值。
- 对于第二组中的计数器，NICE DCV 为每个会话中的每个编码器收集一个值。共有三个活动编码器：
 - 一个全帧编码器
 - 一个基于图块的编码器
 - 一个无损编码器

这些计数器是在 `$session_name:$encoder_name` 实例中发布的。

计数器名称	描述	单位	实例
Grabbed Frames/sec	捕获的帧速率 (以帧/秒为单位)	计数/秒	会话
Grabbed Frames	在启动会话后捕获的总帧数	计数	会话
Sent Frames/sec	每秒向连接的客户端发送屏幕帧的速率	计数/秒	会话
Dropped Frames/sec	每秒未发送到连接的客户端的屏幕帧的速率	计数/秒	会话
Display Latency ms	在捕获和呈现帧之间的平均时间 (以毫秒为单位)	毫秒	会话
Available Bandwidth bits/sec	连接中的估计可用带宽 (以位/秒为单位)	位/秒	会话
Encoded Frames/sec	每秒编码的屏幕帧速率	计数/秒	会话:编码器
Encoding Time ms	用于编码一个屏幕帧的平均时间 (以毫秒为单位)	毫秒	会话:编码器
Encoding Time per Megapixel ms	用于编码一百万像素的平均时间 (以毫秒为单位)	毫秒	会话:编码器
Frame Quality %	平均帧压缩质量, 以百分比表示	百分比	会话:编码器

计数器名称	描述	单位	实例
Frame Compression Ratio %	平均帧压缩率，定义为帧大小（以字节为单位）和压缩帧大小之间的比率	百分比	会话:编码器

NICE DCV 服务器参数参考

下表列出了可以配置以自定义 NICE DCV 服务器的参数。

Note

每个表中的重新加载上下文列指示重新加载参数的时间。可能的上下文包括：

- `server` - 在服务器启动时加载一次参数。如果更新参数值，则在重新启动服务器时将加载新值。
- `session` - 在创建会话时加载参数。如果更新参数值，则将为后续会话加载新值。
- `connection` - 在建立新的客户端连接时加载参数。如果更新参数值，则新值将用于后续客户端连接。
- `custom` - 加载参数的条件是该参数特有的。有关更多信息，请参阅参数说明。

主题

- [audio 参数](#)
- [clipboard 参数](#)
- [connectivity 参数](#)
- [display 参数](#)
- [display/linux 参数](#)
- [input 参数](#)
- [license 参数](#)
- [log 参数](#)
- [printer 参数](#)
- [redirection 参数](#)
- [security 参数](#)
- [session-management 参数](#)
- [session-management/automatic-console-session 参数](#)
- [session-management/defaults 参数](#)

- [smartcard 参数](#)
- [webcam 参数](#)
- [windows 参数](#)
- [修改配置参数](#)

audio 参数

下表介绍了 Linux NICE DCV 服务器的 [audio] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 audio 注册表项。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
avsync-support	字符串	会话	'auto'	确定客户端是否可以启用音频/视频同步 - 允许连接的客户端启用音频/视频同步。有效值为“enabled”、“disabled”或“auto”（默认值为“auto”）。如果指定了“auto”，只有在可以使用加速视频压缩时，才会在控制台会话上启用音频/视频同步。 - 自 2021.1-10557 版本起提供。
source-channels	整数 - DWORD (位)	会话	2	Linux 上的扬声器设备的声道数 - 设置 Linux 扬声器设备的声道数。该值必须小于或等于设备支持的声道数。允许的值为：2 (立体声)、4 (4.0 四声道)、6 (5.1 环绕声)、8 (7.1 环绕声)。默认值为 2 (立体声)。- 自 2020.0-8428 版本起提供。

clipboard 参数

下表介绍了 Linux NICE DCV 服务器的 [clipboard] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 clipboard 注册表项。

参数	类型 - Windows 注册表 类型	重新加 载上下 文	默认值	描述
enabled	true 或 false - DWORD (位)	会话	Linux : true - Windows : 1	是否应启用剪贴板功能 - 指定是否启用剪贴板功能。如果禁用剪贴板功能，用户将无法使用剪贴板远程控制。剪贴板监控也将被禁用。- 自 2020.0-8428 版本起提供。
max-image-area	整数 - DWORD (位)	会话	-1	剪贴板图像的最大区域 - 指定可以在服务器和客户端之间传输的剪贴板图像的最大区域 (像素数)。如果缺少该值或设置为 -1，则不会应用限制。- 自 2017.0-4334 版本起提供。
max-payload-size	整数 - DWORD (位)	会话	20971520	剪贴板数据的最大大小 - 指定可以在服务器和客户端之间传输的剪贴板数据的最大大小 (以字节为单位)。支持的最大值为 20 MB。如果缺少该值，则实施最大限制。- 自 2017.0-4334 版本起提供。
max-text-len	整数 - DWORD (位)	会话	-1	剪贴板文本的最大字符数 - 指定可以从服务器传输到客户端的剪贴板文本的最大字符数。将截断多余的字符。如果缺少该值或设置为 -1，则不会应

参数	类型 - Windows 注册表 类型	重新加 载上下 文	默认值	描述
				用限制。 - 自 2017.0-4334 版本起提供。
primary-selection-copy	true 或 false - DWORD (位)	会话	Linux : false - Windows : 0	在 Linux 中启用主选区复制 - Linux 桌面支持多个剪贴板：通用剪贴板和主选区。选择内容后，将更新或复制主选区域。然后，可以使用鼠标的中间按钮或 Shift+Insert 组合键粘贴它。启用此项后，将监控主选区域并将更新传播到客户端。 - 自 2019.0-7318 版本起提供。
primary-selection-paste	true 或 false - DWORD (位)	会话	Linux : false - Windows : 0	在 Linux 上启用主选区粘贴 - Linux 桌面支持多个剪贴板：通用剪贴板和主选区。选择内容后，将更新或复制主选区域。然后，可以使用鼠标的中间按钮或 Shift+Insert 组合键粘贴它。启用此项后，还会将客户端的剪贴板内容插入主选区中。 - 自 2019.0-7318 版本起提供。
update-timeout	整数 - DWORD (位)	会话	200	更新事件通知超时 - 指定从上次更新事件开始向客户端发送通知的等待时间（以毫秒为单位）。默认值为 200 毫秒。 - 自 2020.1-8942 版本起提供。

connectivity 参数

下表介绍了 Linux NICE DCV 服务器的 [connectivity] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 connectivity 注册表项。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
disconnect-on-lock	true 或 false - DWORD (位)	自定义	Linux : false - Windows : 0	是否在操作系统会话锁定时断开连接客户端 - 启用该参数以在远程操作系统会话锁定时强制断开连接客户端。否则，客户端将继续流式传输远程会话。目前，仅在控制台会话上支持。每次远程操作系统会话锁定时，都会读取该参数值。- 自 2023.1 版本起提供。
disconnect-on-logout	true 或 false - DWORD (位)	自定义	Linux : false - Windows : 0	是否在操作系统用户注销时断开连接客户端 - 启用该参数以在远程操作系统用户注销 (即关闭操作系统会话) 时强制断开连接客户端。否则，客户端将继续流式传输远程会话。目前，仅在控制台会话上支持。每次远程操作系统用户注销时，都会读取该参数值。- 自 2023.1 版本起提供。
enable-quick-front-end	true 或 false - DWORD (位)	server	Linux : false - Windows : 0	是否启用 QUIC 前端 - 指定是否应启用 QUIC 前端。- 自 2020.2-9508 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
idle-timeout	整数 - DWORD (位)	自定义	60	空闲超时 - 指定断开连接空闲客户端之前等待的分钟数。指定 0 表示从不断开空闲客户端。每 5 秒读取此参数值一次。- 自 2017.0-4100 版本起提供。
idle-timeout-warning	整数 - DWORD (位)	自定义	350	空闲超时警告 - 指定在向空闲客户端发出空闲超时断开连接警告之前等待的秒数 (相对于空闲超时)。指定 0 表示从不向空闲客户端发出警告。- 自 2017.4-6898 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
quic-listen-endpoints	字符串	server	['0.0.0.0', '::']	指定 DCV 侦听传入 QUIC 连接的终端节点 - 指定 DCV 将侦听传入 QUIC 连接的终端节点列表。终端节点可以是本地可绑定 IPv4 地址列表 (“0.0.0.0”用于通配所有可能的地址)，也可以是可绑定 IPv6 地址列表 (“::”用于通配所有可能的地址)，并带有由冒号 (“:”) 分隔的可选端口 (“:”) 分隔的可选端口。例如，“1.2.3.4:5678”在端口 5678 上侦听与“1.2.3.4”地址关联的接口上的传入连接。如果未指定端口，则将“quic-port”中的设置作为默认值。要指定具有 IPv6 地址的端口，请将地址用方括号括起来（例如“[::1]:8443”）。还支持包含显式接口的 IPv6 地址（例如“[%eth1]:8443”）。- 自 2022.0-11954 版本起提供。
quic-port	整数 - DWORD (位)	server	8443	QUIC 前端的 UDP 端口 - 指定 DCV 服务器侦听客户端连接的 UDP 端口。端口号必须介于 1024 和 65535 之间。有关如何应用该设置的更多详细信息，请参阅“quic-listen-endpoints”设置。- 自 2020.2-9508 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
web-listen-endpoints	字符串	server	['0.0.0.0', '::']	指定 DCV 侦听传入 Web 连接的终端节点 - 指定 DCV 将侦听传入 Web 连接的终端节点列表。终端节点可以是本地可绑定 IPv4 地址列表 (“0.0.0.0”用于通配所有可能的地址)，也可以是可绑定 IPv6 地址列表 (“::”用于通配所有可能的地址)，并带有由冒号 (“:”) 分隔的可选端口。例如，“1.2.3.4:5678”在端口 5678 上侦听与“1.2.3.4”地址关联的接口上的传入连接。如果未指定端口，则将“web-port”中的设置作为默认值。要指定具有 IPv6 地址的端口，请将地址用方括号括起来（例如“[::1]:8443”）。还支持包含显式接口的 IPv6 地址（例如“[%eth1]:8443”）。- 自 2022.0-11954 版本起提供。
web-port	整数 - DWORD (位)	server	8443	客户端的 TCP 端口 - 指定 DCV 服务器侦听客户端连接的 TCP 端口。端口号必须介于 1024 和 65535 之间。有关如何应用该设置的更多详细信息，请参阅“web-listen-endpoints”设置。- 自 2017.0-4100 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
web-root	字符串	server	"	嵌入式 Web 服务器的文档根目录 - 指定嵌入式 Web 服务器的文档根目录。 - 自 2017.0-4100 版本起提供。
web-url-path	字符串	server	'/'	嵌入式 Web 服务器的 URL 路径 - 指定嵌入式 Web 服务器的 URL 路径，必须以“/”开头。例如，将其设置为 /test/foo 意味着在 https://host:port/test/foo 可访问该 Web 服务器。 - 自 2017.0-4100 版本起提供。
web-use-hsts	true 或 false - DWORD (位)	server	Linux : true - Windows : 1	是否使用 HSTS - 启用该参数以强制浏览器禁止通过 HTTP 发送的任何通信。相反，将使用 HTTPS 执行所有到网页 (以及所有子域) 的传输。 - 自 2017.0-4100 版本起提供。
ws-keepalive-interval	整数 - DWORD (位)	server	10	Websocket keepalive 间隔 - 指定发送 keepalive 消息的间隔 (以秒为单位)。如果设置为 0，则禁用 keepalive 消息。 - 自 2017.0-4100 版本起提供。

display 参数

下表介绍了 Linux NICE DCV 服务器的 [display] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 display 注册表项。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
console-session-default-layout	字符串	会话	[]	控制台会话的默认屏幕分辨率和位置 - 指定控制台会话的默认屏幕分辨率和位置。如果设置了该项，则 DCV 在启动时设置请求的布局。可以为每个监视器配置分辨率 (w,h) 和位置 (x,y)。启用所有指定的监视器。默认布局示例值：[{'w':<800>, 'h':<600>, 'x':<0>, 'y': <0>}, {'w':<1024>, 'h':<768>, 'x':<800>, 'y':<0>}] - 自 2017.0-5600 版本起提供。
cuda-devices	字符串	连接	[]	用于流编码的 CUDA 设备 - 指定 DCV 用于分配编码和 CUDA 工作负载的本地 CUDA 设备列表。每个设备均由一个可从 nvidia-smi 命令中检索的数字标识。例如，cuda-devices=['0', '2'] 表示 DCV 使用两个 ID 分别为 0 和 2 的 GPU。此设置与 CUDA_VISIBLE_DEVICES 环境变量类似，但它仅适用于 DCV。如果未设置此选项，DCV 将使用从 0 开始的增量会话索引来选择要使用的下一个设备。 - 自 2017.2-6182 版本起提供。
enable-client-resize	true 或 false -	会话	Linux : true - Windows : 1	是否允许客户端设置显示布局 - 指定是否允许客户端设置显

参数	类型 - Windows 注册表 类型	重新加 载上下 文	默认值	描述
	DWORD (位)			示布局。 - 自 2017.0-4100 版本起提供。
enable-qu	true 或 false - DWORD (位)	会话	Linux : true - Windows : 1	是否发送质量更新 - 指定是否发送质量更新。 - 自 2017.0-4100 版本起提供。
enable-yuv444-encoding	字符串	会话	“default-off”	是否启用 YUV444 编码 - 启用或禁用 YUV444 编码。如果为“always-on”，服务器优先使用针对高色彩精度优化的 YUV444 格式。如果为“always-off”，服务器优先使用针对流式传输性能优化的格式。“default-on”和“default-off”值具有相同的语义，即，由客户端决定。允许的值：always-on、always-off、default-on、default-off。 - 自 2022.0-11954 版本起提供。
grabber-target-fps	整数 - DWORD (位)	会话	0	帧抓取器的目标每秒帧数 - 设置每秒抓取帧数的上限。值 0 默认为每个特定帧缓冲区读取器的标准行为，例如改用 target-fps 或不限限制抓取。并非所有帧捕获后端都支持该设置。 - 自 2017.1-5870 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
max-compressor-threads	整数 - DWORD (位)	会话	4	最大压缩器线程数 - 指定最大压缩器线程数。- 自 2017.0-4100 版本起提供。
max-head-resolution	字符串	自定义	(4096, 2160)	最大头分辨率 - 设置客户端请求的显示头的最大分辨率。显示头等效于主显示器。在发出每个客户端布局请求时将重新加载该设置。当客户端请求较大的头分辨率时，服务器会调整分辨率，以确保它与此选项设置的最大宽度值和最大高度值匹配。- 自 2017.0-4100 版本起提供。
max-layout-area	整数 - DWORD (位)	自定义	0	最大布局区域 (以像素为单位) - 设置客户端可请求的显示布局的最大区域 (以像素为单位)。超出此限制的布局将被忽略。此最大值旨在提供必须发送的显示数据量的上限，而不提供显示布局几何体的约束。如果设置为 0，则不对布局区域施加任何限制。在发出每个客户端布局请求时将重新加载该设置。- 自 2019.1-7423 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
max-num-heads	整数 - DWORD (位)	自定义	4	最大头数 - 指定客户端可请求的最大显示头数。显示头等效于主显示器。在发出每个客户端布局请求时将重新加载该设置。当客户端请求更多数量的头时，服务器会调整头数，以使该值不超过此选项设置的值。- 自 2017.0-4100 版本起提供。
min-head-resolution	字符串	自定义	(640, 480)	最小头分辨率 - 设置客户端可请求的显示头的最小分辨率。显示头等效于主显示器。在发出每个客户端布局请求时将重新加载该设置。当客户端请求较小的头分辨率时，服务器会调整分辨率，以确保它与此选项设置的最小宽度值和最小高度值匹配。- 自 2017.0-4100 版本起提供。
target-fps	整数 - DWORD (位)	会话	-1	目标每秒帧数 - 指定每秒允许的最大帧数。值 0 表示无限制。值为 -1 表示，将根据服务器特性和会话类型确定 target-fps 值。对于低于 2020.2 的版本，无法识别 -1 值，默认值为 25。- 自 2017.0-4100 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
use-grabber-dirty-region	true 或 false - DWORD (位)	会话	Linux : true - Windows : 1	是否使用脏区域 - 指定是否使用脏屏幕区域。如果启用，则抓取器试图从屏幕的脏区域中抓出新帧。- 自 2017.0-4100 版本起提供。
web-client-max-head-resolution	字符串	自定义	(1920, 1080)	Web 客户端的最大头分辨率 - 设置 Web 客户端可请求的显示头的最大分辨率。显示头等效于主显示器。在发出每个客户端布局请求时将重新加载该设置。如果 Web 客户端明确设置了最大分辨率，则忽略此设置。基于此选项设置的最大宽度值和最大高度值应用 max-head-resolution 限制选项。如果此值设置为 (0, 0)，则忽略该选项。- 自 2020.0-8428 版本起提供。

display/linux 参数

下表介绍了 Linux NICE DCV 服务器的 [display/linux] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 display/linux 注册表项。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
gl-displays	字符串	会话	[:0.0]	3D 加速 X 显示器 - 指定 DCV 用于虚拟会话中的 OpenGL 渲染的本地 3D 加速 X 显示器和屏幕列表。如果缺少此值，则无法在虚拟会话中运行 OpenGL 应用程序。对于控制台会话，忽略此设置。- 自 2017.0-4100 版本起提供。

input 参数

下表介绍了 Linux NICE DCV 服务器的 [input] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 input 注册表项。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
enable-autorepeat	true 或 false - DWORD (位)	会话	Linux : true - Windows : 1	是否允许在 Linux 上自动重复 - 指定是否允许自动重复单个键。- 自 2017.2-6182 版本起提供。
enable-gamepad	true 或 false - DWORD (位)	会话	Linux : true - Windows : 1	是否允许游戏手柄输入 - 指定是否启用游戏手柄。- 自 2022.0-11954 版本起提供。

参数	类型 - Windows 注册表 类型	重新加 载上下 文	默认值	描述
enable-relative-move	true 或 false - DWORD (位)	会话	Linux : true - Windows : 1	是否允许相对鼠标移动 - 指定是否允许相对鼠标移动。 - 自 2017.0-5121 版本起提供。
enable-stylus	true 或 false - DWORD (位)	会话	Linux : true - Windows : 1	是否允许触控笔输入 - 指定是否启用触控笔。 - 自 2019.0-7318 版本起提供。
enable-touch	true 或 false - DWORD (位)	会话	Linux : true - Windows : 1	是否允许触摸输入 - 指定是否启用触摸。 - 自 2017.3-6698 版本起提供。

license 参数

下表介绍了 Linux NICE DCV 服务器的 [license] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 license 注册表项。

参数	类型 - Windows 注册表 类型	重新加 载上下 文	默认值	描述
license-file	字符串	会话	"	许可证 - 指定 DCV 服务器在非 EC2 实例上运行时使用的许可证。许可是通过 RLM 许可证授予的。它可以包含一组许可证规范，以“;” (在 Windows 上) 或“:” (在 Linux 上) 分

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
				隔。每个许可证规范可以是延期评估许可证的本地许可证文件，也可以是以“端口@主机名”格式为浮动许可证指定的 RLM 服务器端口和主机名。如果指定了多个许可证，服务器将依次尝试每个许可证，直到验证了第一个许可证（例如，正确识别了许可证文件或可以连接到远程 RLM 服务器）。如果未指定任何值，服务器将查找默认许可证文件（Linux 上的“/usr/share/license/license.lic”或 Windows 上的“C:\Program Files\NICE\DCV\Server\license\license.lic”）；如果找不到默认许可证文件，则使用演示许可证。在 EC2 实例上，将忽略该参数。- 自 2017.0-4100 版本起提供。

log 参数

下表介绍了 Linux NICE DCV 服务器的 [log] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 log 注册表项。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
directory	字符串	server	"	日志输出目录 - 指定将日志保存到的目标。如果未指定此项，则默认为“C:\ProgramData\NICE\DCV\log”（在 Windows 上）和“/var/log/dcv/”（在 Linux 上）。- 自 2017.0-4100 版本起提供。
enable-image-audit	true 或 false - DWORD (位)	server	Linux : false - Windows : 0	启用传输的图像的内容审核 - 指定是否必须将任何传输的图像内容保存在单独的文件中。这些图像将存储在日志子目录中，并在审核 CSV 文件中报告文件名。如果禁用了 transfer-audit，将忽略该值。- 自 2023.0-14852 版本起提供。
level	字符串	自定义	“info”	日志级别 - 指定日志文件详细程度。详细程度（按提供的详细信息量排列）包括：“error”、“warn”、“info”和“debug”。新值在配置上进行更改并传播到 DCV 代理进程后立即生效。对于 2019.1 版本以及之前的版本，仅在 DCV 代理进程启动时设置其日志级别。- 自 2017.0-4100 版本起提供。
max-file-size	整数 - DWORD (位)	server	0	轮换前的最大日志文件大小（以兆字节为单位）- 指定触发轮换之前的最大日志文件

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
				大小。如果值为“0”，则禁用按大小轮换，而是在重新启动生成日志文件的进程时进行轮换。- 自 2022.1-13067 版本起提供。
rotate	整数 - DWORD (位)	server	10	日志文件轮换次数 - 指定在删除之前轮换日志文件的次数。如果值为 0，则删除而不分割旧版本。- 自 2017.0-4100 版本起提供。
rotation-interval	字符串	server	“none”	两次连续日志文件轮换的最大时间间隔 - 指定两次连续日志文件轮换的最大时间间隔。如果值为“none”，则不会根据时间轮换文件。其他可能的值为“every-minute”、“every-twenty-minutes”、“every-hour”和“every-day”。- 自 2022.1-13067 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
rotation-suffix	字符串	server	“counter”	要附加到轮换的日志文件的后缀 - 指定要附加到轮换的日志文件的后缀。如果指定了“counter”，则将一个简单的递增计数器后缀附加到每个轮换的日志文件中。如果指定了“timestamp”，则将“YYYY-MM-DD-HH-MM”格式的时间戳应用于日志文件。如果具有该时间戳的轮换文件在日志文件夹中已存在，则会在该时间戳后面附加一个额外数字计数器。 - 自 2022.1-13067 版本起提供。
transfer-audit	字符串	server	“none”	要审核的传输方向 - 指定要审核的传输方向。如果启用此参数，则在服务器和客户端之间传输新的 CSV 文件日志。允许的值：none、server-to-client、client-to-server 和 all。如果此值缺失或等于“none”，则禁用传输审核且不创建文件。 - 自 2017.0-4100 版本起提供。

printer 参数

下表介绍了 Linux NICE DCV 服务器的 [printer] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 printer 注册表项。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
file-printer-name	字符串	自定义	“DCV Printer”	用于文件下载的虚拟 DCV 打印机的名称 - 表示 DCV 服务器上的虚拟 DCV 打印机名称的字符串。在 Linux 中，每次创建新的 Linux DCV 会话时，都会从配置中读取该值。如果该设置不是空的，并且将 PREFIX 字符串作为值，则会在 CUPS 中注册名为“PREFIX - SESSION-NUMBER”的新虚拟打印机。如果该设置是空的，则不会注册任何 DCV 虚拟打印机。在 Windows 中，该设置用于更改系统上的默认打印机。如果设置为空字符串，则 DCV 不会更改当前的默认打印机。- 自 2022.0-11954 版本起提供。
use-default-printer	字符串	自定义	“client-decides”	决定如何设置默认打印机 - 服务器决定将哪个打印机设置为默认打印机。接受的值为“client-decides”、“always-on”、“always-off”。如果该设置的值为“always-off”，则服务器不会将任何打印机设置为默认打印机。如果值为“always-on”，则设置“printer”部分下面的“file-printer-name”设置中指定的打印机。如果值为“client-decides”，则设置从客户端发送的默认打印机。

参数	类型 - Windows 注册表 类型	重新加 载上下 文	默认值	描述
				如果客户端没有发送默认打印机，则设置“printer”部分下面的“file-printer-name”中指定的打印机。默认值为“client-decides”。目前仅在 Windows 上支持。- 自 2022.2-13907 版本起提供。

redirection 参数

下表介绍了 Linux NICE DCV 服务器的 [redirection] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 redirection 注册表项。

参数	类型 - Windows 注册表 类型	重新加 载上下 文	默认值	描述
enable-timezone-redirection	字符串	会话	“client-decides”	允许或拒绝从客户端到服务器的时区重定向 - 允许或拒绝将时区从客户端重定向到服务器。接受的值为：“always-on”、“always-off”和“client-decides”。如果设置为“always-on”，则优先用户客户端将其时区发送到服务器，并且该时区变为服务器的时区。如果设置为“always-off”，则服务器向客户端显示自己的时区。将丢弃任何客户端时区消息。如果设置为“client-decides”，则优

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
				先用户客户端可以将其时区发送到服务器，并且该时区变为服务器的时区。客户端可以选择不将其时区发送到服务器。默认值为“client-decides”。- 自 2022.2-13907 版本起提供。

security 参数

下表介绍了 Linux NICE DCV 服务器的 [security] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 security 注册表项。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
allowed-http-host-regex	字符串	server	'^.+\$\$'	允许的主机正则表达式 - 指定一个正则表达式模式，它表示该 DCV 服务器可以提供服务的主机名。如果传入 HTTP 请求的主机标头与此模式不匹配，则请求本身会失败，并伴随出现“403 (禁止访问)”的状态码。这是防止 HTTP 主机标头攻击的一种安全手段。该模式必须是有效的类似于 Javascript 的正则表达式。模式中的字母大小写同时匹配。示例：“^(www\.)?example\.com\$”。- 自 2017.0-4100 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
allowed-ws-origin-regex	字符串	server	“^https://.+”	<p>允许的来源 - 指定一个正则表达式模式，它表示该 DCV 服务器接受的来源。建立 WebSocket 连接时，客户端的握手中的源标头字段指示建立连接的脚本的源。如果传入 HTTP 请求的源标头与此模式不匹配，则请求本身会失败，并伴随出现“403 (禁止访问)”的状态码。这是防止跨站点 WebSocket 劫持 (CSWSH) 攻击的一种安全手段。该模式必须是有效的类似于 Javascript 的正则表达式。模式中的字母大小写同时匹配。源标头的形式如下： <scheme> "://" <host> [":" <port>]。示例：“^https://(www\.)?example\.com(:443)?\$”。</p> <p>- 自 2017.0-4100 版本起提供。</p>
auth-connection-setup-timeout	整数 - DWORD (位)	server	120	<p>身份验证通道连接设置超时 - 指定为身份验证通道连接设置过程留出的时间 (以秒为单位)，以在超时之前完成该过程。如果设置过程超时，则通道被关闭。如果设置为 0，则禁用身份验证通道连接设置超时。 - 自 2017.0-4100 版本起提供。</p>

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
auth-token-verifier	字符串	server	"	身份验证令牌验证程序的终端节点 - 指定 DCV 服务器使用的身份验证令牌验证程序的终端节点 (URL)。如果为空，则使用内部身份验证令牌验证程序。- 自 2017.0-4100 版本起提供。
auth-token-verifier-timeout	整数 - DWORD (位)	server	100	身份验证令牌验证程序超时 (以秒为单位) - 指定等待 DCV 服务器使用的身份验证令牌验证程序的时间 (以秒为单位)。- 自 2023.0-14852 版本起提供。
authentication	字符串	server	"system"	身份验证方法 - 指定 DCV 服务器使用的客户端身份验证方法。使用 "system" 将客户端身份验证工作委派给底层操作系统。使用 "none" 以禁用客户端身份验证并授权访问所有客户端。- 自 2017.0-4100 版本起提供。
authentication-threshold	整数 - DWORD (位)	server	3	身份验证阈值 - 指定在服务器关闭连接之前每个客户端可以失败的身份验证次数。要允许无限制的身份验证尝试，请使用 0。- 自 2017.0-4100 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
ca-file	字符串	server	"	CA 文件 - 指定包含 DCV 服务器信任的证书颁发机构 (CA) 的文件。如果为空，请使用系统提供的默认信任存储。- 自 2017.0-4100 版本起提供。
certificate-to-user-file	字符串	自定义	"	证书到用户映射文件 - 指定包含证书到用户映射列表的文件。- 自 2022.0-11954 版本起提供。
ciphers	字符串	server	"ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA384"	TLS 连接上使用的密码列表 - 指定 TLS 连接上使用的密码列表。该密码列表必须使用字符“:”分隔，且必须得到 openssl 和客户端支持。- 自 2017.0-4100 版本起提供。

参数	类型 - Windows 注册表 类型	重新加 载上下 文	默认值	描述
connection-estab-t imeout	整数 - DWORD (位)	server	5	连接建立超时 - 指定为连接过程留出的时间 (以秒为单位), 以在超时之前完成该过程。如果连接过程超时, 则连接被关闭。如果设置为 0, 则建立连接无超时限制。- 自 2017.0-4100 版本起提供。
connection-setup-t imeout	整数 - DWORD (位)	server	5	通道连接设置超时 - 指定为通道连接设置过程留出的时间 (以秒为单位), 以在超时之前完成该过程。如果设置过程超时, 则通道被关闭。如果设置为 0, 则通道连接设置无超时限制。- 自 2017.0-4100 版本起提供。
crl-file	字符串	自定义	"	CRL 文件 - 指定包含证书吊销列表 (CRL) 的文件。- 自 2022.0-11954 版本起提供。
enable-gssapi	true 或 false - DWORD (位)	server	Linux : false - Windows : 0	启用 GSSAPI SASL 机制 - 启用或禁用 GSSAPI SASL 机制, 该机制允许使用 Kerberos 进行 DCV 身份验证。- 自 2017.3-6698 版本起提供。
max-connections- per-user	整数 - DWORD (位)	server	10	用户的最大连接数 - 指定每个用户允许的最大并发连接数。超出的连接将被拒绝。- 自 2017.0-4100 版本起提供。

参数	类型 - Windows 注册表 类型	重新加 载上下 文	默认值	描述
no-tls-strict	true 或 false - DWORD (位)	server	Linux : false - Windows : 0	启用或禁用严格证书验证 - 在连接到外部身份验证令牌验证程序时启用或禁用严格证书验证。如果身份验证令牌验证程序使用自签名证书，则必须禁用严格的证书验证。- 自 2017.0-4100 版本起提供。
os-auto-lock	true 或 false - DWORD (位)	会话	Linux : true - Windows : 1	是否在最后一个客户端连接结束时锁定操作系统会话 - 如果启用，则在关闭最后一个客户端连接时锁定操作系统会话。- 自 2017.1-5777 版本起提供。
pam-service-name	字符串	server	“dcv”	PAM 服务名称 - 指定 DCV 使用的 PAM 配置文件的名称。默认 PAM 服务名称为“dcv”，对应 /etc/pam.d/dcv 配置文件。仅当使用“system”身份验证方法时才使用此参数。- 自 2017.0-4100 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
passwd-file	字符串	server	"	密码文件 - 指定用于检查用户凭证的密码文件 (仅适用于 DCV 身份验证模式)。如果为空，请对 Linux 使用 <code>\${XDG_CONFIG_HOME}/NICE/dcv/passwd</code> 中的默认文件，或对 Windows 使用 <code>%CSIDL_LOCAL_APPDATA%\NICE\dcv\passwd</code> 中的默认文件。- 自 2017.0-4100 版本起提供。
server-fqdn	字符串	server	"	服务器 FQDN - 指定服务器完全限定域名。空白表示 <code>gethostname()</code> 。- 自 2017.3-6698 版本起提供。
service-name	字符串	server	"dcv"	服务名称 - 服务的注册名称 (通常是协议名称)。- 自 2020.0-8428 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
supervision-control	字符串	自定义	“disabled”	会话的监督控制类型 - 指定会话的监督控制类型。可能的值为“disabled”和“enforced”。如果该值设置为“enforced”，则可以将无监督访问权限设置为允许或拒绝用户在协作会话中进行无所有者访问。如果允许用户进行无监督访问，则用户可以在没有所有者的情况下访问会话。默认情况下，为所有者以外的所有用户拒绝该权限。如果该值设置为“disabled”（默认值），则服务器不会强制实施该监督控制和权限。在配置中更改后，新值立即生效。- 自 2021.3-11591 版本起提供。
user-realm	字符串	server	“	服务器用户领域 - 指定服务器的用户领域。- 自 2017.3-6698 版本起提供。

session-management 参数

下表介绍了 Linux NICE DCV 服务器的 [session-management] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 session-management 注册表项。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
create-session	true 或 false - DWORD (位)	server	Linux : false - Windows : 0	在服务器启动时创建控制台会话 - 指定是否在服务器启动时自动创建控制台会话 (ID 为“console”)。 - 自 2017.0-4100 版本起提供。
enable-gl-in-virtual-sessions	字符串	会话	“default-on”	是否使用 dcv-gl 功能 - 指定是否使用 dcv-gl 功能 (需要许可证)。允许的值 : always-on、always-off、default-on、default-off。 - 自 2017.0-4100 版本起提供。
max-concurrent-clients	整数 - DWORD (位)	会话	-1	每个会话的最大并发客户端数 - 指定每个会话的最大并发客户端数。如果设置为 -1，则表示无限制。若要只设置自动会话的限制，则使用“session-management/automatic-console-session”部分的“max-concurrent-clients”。 - 自 2017.0-4100 版本起提供。
max-concurrent-sessions	整数 - DWORD (位)	server	0	最大并发会话数 - 指定允许的最大并发会话数。此限制当前仅适用于虚拟会话，因为控制台会话本质上限于一个。指定 0 表示不强制施加任何限制。 - 自 2019.0-7318 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
max-sessions-per-user	整数 - DWORD (位)	server	0	每个用户的最大会话数 - 指定每个用户可以拥有的最大允许并发会话数。该限制目前仅适用于虚拟会话。指定 0 表示不强制施加任何限制。 - 自 2021.0-10242 版本起提供。
virtual-session-default-layout	字符串	会话	[]	虚拟会话的默认布局 - 如果设置了该参数，Xdcv 将配置为在启动时创建指定的布局。可以为每个监视器配置分辨率 (w,h) 和位置 (x,y)。启用所有指定的监视器。默认布局示例值：[{'w':<800>, 'h':<600>, 'x':<0>, 'y': <0>}, {'w':<1024>, 'h':<768>, 'x':<800>, 'y':<0>}] 对于此设置，最大监视器数（在 virtual-session-monitors 设置中指定）的优先级高于数组中的元素数。例如，如果设置了五台监视器，而最大监视器数为四，则只创建前四台监视器。如果设置此键，则忽略已启用监视器数（在 virtual-session-monitors 设置中指定）。 - 自 2017.0-5600 版本起提供。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
virtual-session-font-path	字符串	会话	"	是否添加特殊字体路径 - 指定特殊字体的路径。有些应用程序要求将特殊字体传递到 X 服务器。 - 自 2017.0-4100 版本起提供。
virtual-session-source-profile	true 或 false - DWORD (位)	会话	Linux : false - Windows : 0	是否在会话启动器中获取用户配置文件 - 指定运行会话启动器脚本的 Shell 是否应获取用户配置文件。默认情况下，该值为 false，DCV 将使用“bash --noprofile --norc”运行会话启动器脚本 - 自 2021.3-11591 版本起提供。
virtual-session-xdcv-args	字符串	会话	"	传递给 Xdcv 的其他参数 - 指定要传递给 Xdcv 的任何其他参数。 - 自 2017.0-4334 版本起提供。

session-management/automatic-console-session 参数

下表介绍了 Linux NICE DCV 服务器的 [session-management/automatic-console-session] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 session-management/automatic-console-session 注册表项。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
client-eviction-policy	字符串	server	“reject-new-connection”	指定达到限制时如何处理客户端连接 - 指定在达到每个会话的最大并发客户端数时是拒绝新连接，还是自动关闭现有的连接。允许的值为“reject-new-connection”（关闭传入连接）和“same-user-oldest-connection”（服务器关闭同一用户未与会话交互时间最长的连接；如果没有该信息，则关闭连接时间最长的连接）。- 自 2022.1-13067 版本起提供。
max-concurrent-clients	整数 - DWORD (位)	server	-1	每个会话的最大并发客户端数 - 指定每个会话允许的最大并发客户端数。如果设置为 -1，则表示无限制。- 自 2017.0-5600 版本起提供。
owner	字符串	server	”	自动创建的“console”会话的所有者 - 指定“console”会话所有者的用户名。如果为空，则所有者为启动了 DCV 服务器的用户。此设置仅在 create-session 设置为“true”的情况下，适用于在服务器启动时自动创建的“控制台”会话。- 自 2017.0-5600 版本起提供。
permissions-file	字符串	server	”	自动“console”会话的权限文件 - 指定用于检查用户的 DCV 功能访问权限的权限文件的路

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
				径。如果为空，则只有所有者才对会话拥有完全访问权限。 - 自 2017.0-5600 版本起提供。
storage-root	字符串	server	"	文件存储根文件夹的路径 - 指定用于控制台会话存储的文件夹的完整路径。如果 storage-root 为空或该文件夹不存在，则禁用文件存储。 - 自 2017.0-5600 版本起提供。

session-management/defaults 参数

下表介绍了 Linux NICE DCV 服务器的 [session-management/defaults] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 session-management/defaults 注册表项。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
permissions-file	字符串	会话	"	所有会话中包含的默认权限 - 指定要与用户为每个会话选择的权限自动合并的权限文件的路径。如果为空，请使用“default.perm”文件。对于 Linux，该文件位于 /etc/dcv/ 下，对于 Windows，该文件位于 DCV 安装文件夹中（如 C:\Program Files\NICE\DCV\Ser

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
				ver\conf)。 - 自 2017.0-5600 版本起提供。

smartcard 参数

下表介绍了 Linux NICE DCV 服务器的 [smartcard] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 smartcard 注册表项。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
enable-cache	字符串	自定义	“default-on”	是否启用智能卡缓存消息 - 启用或禁用智能卡缓存。启用此项后，NICE DCV 服务器将缓存从客户端的智能卡收到的最后一个值。未来的调用直接从服务器的缓存中检索，而不是从客户端检索。这有助于减少在客户端和服务器之间传输的流量，并提高性能。允许的值包括“always-on”、“always-off”、“default-on”和“default-off”。每当启动客户端智能卡应用程序时，都会从配置中读取此值。 - 自 2017.2-6182 版本起提供。

webcam 参数

下表介绍了 Linux NICE DCV 服务器的 [webcam] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 webcam 注册表项。

参数	类型 - Windows 注册表 类型	重新加 载上下 文	默认值	描述
max-resolution	字符串	连接	(1280, 720)	最大网络摄像头分辨率 - 指定向应用程序公开的最大网络摄像头分辨率。 - 自 2021.0-10 242 版本起提供。
preferred-resolution	字符串	连接	(480, 360)	首选的网络摄像头分辨率 - 在客户端提供的分辨率中指定首选的网络摄像头分辨率。如果不支持指定的分辨率，则会选择最接近的匹配分辨率，并向应用程序公开该分辨率。如果指定的值之一为 0，则禁用网络摄像头共享。 - 自 2021.0-10 242 版本起提供。

windows 参数

下表介绍了 Linux NICE DCV 服务器的 [windows] 文件的 /etc/dcv/dcv.conf 部分中的配置参数以及 Windows NICE DCV 服务器的 windows 注册表项。

参数	类型 - Windows 注册表类型	重新加载上下文	默认值	描述
disable-display-sleep	true 或 false - DWORD (位)	会话	Linux : true - Windows : 1	禁止显示器进入节能模式 - 指定是否禁止显示器进入节能模式。 - 自 2017.0-4100 版本起提供。
printer	字符串	会话	"	要设置为默认值的打印机 - 指定虚拟 DCV 打印机的名称。该名称用于更改系统上的默认打印机。如果设置为空字符串，则 DCV 不会更改当前的默认打印机。已弃用：请使用“printer”部分的“file-printer-name”。 - 自 2017.0-4100 版本起提供。

修改配置参数

本节介绍了如何修改 NICE DCV 服务器的配置参数。有关 Windows 服务器的注册表项、Linux 服务器的部分、参数名称、类型和有效值的详细信息，请参阅 [NICE DCV 服务器参数参考](#)。

主题

- [Windows NICE DCV 服务器](#)
- [Linux NICE DCV 服务器](#)

Windows NICE DCV 服务器

对于 Windows NICE DCV 服务器，请使用 Windows 注册表编辑器、PowerShell 或命令行修改配置参数。

使用 Windows 注册表编辑器修改配置参数

1. 打开 Windows 注册表编辑器。

2. 导航到以下注册表路径：

```
HKEY_USERS\S-1-5-18\Software\GSettings\com\nicesoftware\dcv/
```

3. 选择参数所在的注册表项。如果该注册表项不存在，则使用 [NICE DCV 服务器参数参考](#) 中所述的确切的注册表项名称创建该注册表项。
4. 打开（双击）该参数。如果该参数不存在，则使用 [NICE DCV 服务器参数参考](#) 中所述的类型和名称添加该参数。

使用 PowerShell 修改配置参数

1. 以管理员身份运行 PowerShell。
2. 使用 [NICE DCV 服务器参数参考](#) 中所述的注册表项名称添加该注册表项。

```
PS C:\> New-Item -Path "Microsoft.PowerShell.Core\Registry::\HKEY_USERS  
\S-1-5-18\Software\GSettings\com\nicesoftware\dcv\" -Name registry_key -Force
```

3. 使用 [NICE DCV 服务器参数参考](#) 中所述的类型和名称在注册表项中创建参数。

```
PS C:\> New-ItemProperty -Path "Microsoft.PowerShell.Core\Registry::  
\HKEY_USERS\S-1-5-18\Software\GSettings\com\nicesoftware\dcv\registry_key" -  
Name parameter_name -PropertyType parameter_type -Value parameter_value -Force
```

使用命令行修改配置

1. 以管理员身份运行命令行。
2. 使用 [NICE DCV 服务器参数参考](#) 中所述的注册表项名称、参数类型和名称创建注册表项并添加参数。

```
C:\> reg.exe ADD "HKEY_USERS\S-1-5-18\Software\GSettings\com\nicesoftware\dcv  
\registry_key" /v parameter_name /t parameter_type /d parameter_value /f
```

Linux NICE DCV 服务器

对于 Linux NICE DCV 服务器，可以使用文本编辑器或命令行工具（例如 `crudini`）修改配置参数。

使用文本编辑器修改配置参数

1. 使用所需的文本编辑器打开 `/etc/dcv/dcv.conf`。
2. 找到文件中的相应部分。如果该部分不存在，则使用 [NICE DCV 服务器参数参考](#) 中所述的部分名称添加该部分。

```
[section]
```

3. 找到部分中的相应参数并修改其值。如果该参数不存在，则使用 [NICE DCV 服务器参数参考](#) 中所述的参数名称添加该参数。

```
parameter_name="parameter_value"
```

4. 保存并关闭该文件。

使用 crudini 修改配置参数

使用 [NICE DCV 服务器参数参考](#) 中所述的部分和参数名称创建该部分并添加参数。

```
$ sudo crudini --set /etc/dcv/dcv.conf section_name parameter_name 'parameter_value'
```

NICE DCV 支持生命周期结束

NICE DCV 支持生命周期结束 (EOSL) 定义了一个时间点，在此之后，NICE DCV 的特定主要版本（及其所有次要版本）将不再获得支持，并且不再测试该版本与新版本的兼容性。

在 EOSL 日期之前，NICE DCV 支持团队继续为配置问题提供全面支持。仅为最新版本的 NICE DCV 服务器和 NICE DCV 客户端实施缺陷解决方案和功能请求。不会为旧版本实施缺陷解决方案和功能请求。

在 EOSL 日期之后，不再提供进一步的支持或维护。我们还会停止测试兼容性问题。要继续获得支持，您必须升级到最新的 NICE DCV 版本。

主题

- [EOSL 时间表](#)
- [客户的 EOSL 途径](#)
- [EOSL 常见问题解答](#)

EOSL 时间表

下表显示了 NICE DCV 主要版本的 EOSL 时间表。

NICE DCV 主要版本	最初发行日期	EOSL 日期
NICE DCV 2016.x	2015 年 12 月 31 日	2021 年 3 月 31 日
NICE DCV 2017.x	2017 年 12 月 18 日	2021 年 12 月 31 日
NICE DCV 2019.x	2019 年 8 月 5 日	2022 年 12 月 31 日
NICE DCV 2020.x	2020 年 4 月 16 日	2023 年 12 月 31 日

NICE DCV 主要版本	最初发行日期	EOSL 日期
NICE DCV 2021.x	2021 年 4 月 12 日	2024 年 12 月 31 日
NICE DCV 2022.x	2022 年 2 月 23 日	2025 年 12 月 31 日
NICE DCV 2023.x	2023 年 5 月 3 日	2026 年 12 月 31 日

客户的 EOSL 途径

如果您在 AWS 上运行 NICE DCV，则不需要 NICE DCV 许可证。您只需为用于您的工作负载的底层 AWS 资源付费。如果您当前使用的 NICE DCV 版本已超过 EOSL 日期，请使用 [NICE 下载页面](#) 升级到最新的 NICE DCV 版本，或者使用 AWS Marketplace 中的 [NICE DCV AMI](#) 继续获得支持。

如果您在本地运行 NICE DCV 或使用第三方云服务提供商，并且您当前使用的 NICE DCV 版本已超过 EOSL 日期，请与您的经销商或分销商联系以评估可用的升级途径。如果您具有有效的支持合同，您可以免费升级到最新版本的 NICE DCV。有关 NICE DCV 分销商和经销商的信息，请参阅 [NICE 网站](#)。

EOSL 常见问题解答

1. 我使用的 NICE DCV 版本已在本地或第三方云服务提供商处到达 EOSL 日期，但我当前具有支持合同。我会受到 EOSL 影响吗？

如果您具有有效的支持合同，则 NICE DCV 支持合同的条款允许您将 NICE DCV 许可证升级到最新版本，而无需支付额外费用。在这种情况下，影响微乎其微。如果您的支持合同已过期，您可以使用以下方法之一继续获得全面支持：

1. 使用新的付费许可证升级到最新版本的 NICE DCV 版本。
2. 在 EOSL 时间表之前续订您的支持合同，这会为您提供一种升级到最新 NICE DCV 版本的途径。
3. 支付恢复费用以恢复旧支持合同，该费用相当于在您的支持合同过期后的这段时间的当前支持服务费用的 70%。

2. 我在 Amazon EC2 上使用的 NICE DCV 版本已到达 EOSL 日期，我应该如何升级到支持的版本？

客户可以随时升级到完全支持的 NICE DCV 版本以在 Amazon EC2 上使用，而无需支付额外费用。

3. 我是否可以将到达 EOSL 日期的 NICE DCV 客户端版本与支持的 NICE DCV 服务器一起使用，或者将到达 EOSL 日期的 NICE DCV 服务器版本与支持的 NICE DCV 客户端一起使用？

可以，但我们强烈建议您将客户端和服务端软件升级到最新版本，因为不再将错误修复应用于到达 EOSL 日期的版本。

安全性

云安全 AWS 是重中之重。作为 AWS 客户，您可以受益于专为满足大多数安全敏感型组织的要求而构建的数据中心和网络架构。

安全是双方 AWS 的共同责任。[责任共担模式](#)将其描述为云的安全性和云中的安全性：

- 云安全 — AWS 负责保护在 AWS 云中运行 AWS 服务的基础架构。AWS 还为您提供可以安全使用的服务。作为[AWS 合规计划](#)的一部分，第三方审计师定期测试和验证我们安全的有效性。要了解适用于 NICE DCV 的合规计划，请参阅按合规计划提供的[范围内的 AWS 服务按合规计划](#)。
- 云端安全-您的责任由您使用的 AWS 服务决定。您还需要对其他因素负责，包括您的数据的敏感性、您的公司的要求以及适用的法律法规。

本文档帮助您了解如何在使用 NICE DCV 时应用责任共担模式。以下主题说明了如何配置 NICE DCV 以满足您的安全性和合规性目标。您还将学习如何使用其他 AWS 服务来帮助您监控和保护您的 NICE DCV 资源。

内容

- [NICE DCV 中的数据保护](#)
- [NICE DCV 合规性验证](#)

NICE DCV 中的数据保护

分 AWS [担责任模型](#)适用于 NICE DCV 中的数据保护。如本模型所述 AWS，负责保护运行所有内容的全球基础架构 AWS Cloud。您负责维护对托管在此基础架构上的内容的控制。您还负责您所使用的 AWS 服务的安全配置和管理任务。有关数据隐私的更多信息，请参阅[数据隐私常见问题](#)。有关欧洲数据保护的信息，请参阅 AWS Security Blog 上的 [AWS Shared Responsibility Model and GDPR](#) 博客文章。

出于数据保护目的，我们建议您保护 AWS 账户凭证并使用 AWS IAM Identity Center 或 AWS Identity and Access Management (IAM) 设置个人用户。这样，每个用户只获得履行其工作职责所需的权限。我们还建议您通过以下方式保护数据：

- 对每个账户使用多重身份验证 (MFA)。
- 使用 SSL/TLS 与资源通信。AWS 我们要求使用 TLS 1.2，建议使用 TLS 1.3。

- 使用设置 API 和用户活动日志 AWS CloudTrail。
- 使用 AWS 加密解决方案以及其中的所有默认安全控件 AWS 服务。
- 使用高级托管安全服务（例如 Amazon Macie），它有助于发现和保護存储在 Amazon S3 中的敏感数据。
- 如果您在 AWS 通过命令行界面或 API 进行访问时需要经过 FIPS 140-2 验证的加密模块，请使用 FIPS 端点。有关可用的 FIPS 端点的更多信息，请参阅 [《美国联邦信息处理标准 \(FIPS\) 第 140-2 版》](#)。

我们强烈建议您切勿将机密信息或敏感信息（如您客户的电子邮件地址）放入标签或自由格式文本字段（如名称字段）。这包括你使用控制台、API 或 SDK AWS 服务使用 NICE DCV 或其他软件开发工具包 AWS CLI 的情况。AWS 在用于名称的标签或自由格式文本字段中输入的任何数据都可能会用于计费或诊断日志。如果您向外部服务器提供网址，强烈建议您不要在网址中包含凭证信息来验证对该服务器的请求。

数据加密

所有安全服务均具有一项重要功能，即信息在未处于活动使用状态时都会加密。

静态加密

NICE DCV 本身不存储任何客户数据。可以静态加密 NICE DCV 服务器主机上的数据。在上使用 NICE DCV 时 AWS，请参阅《亚马逊 EC2 用户指南》中的“[静态加密](#)”部分和《亚马逊 EC2 用户指南》中的“[静态加密](#)”部分。

传输中加密

对于从 NICE DCV 客户端和 NICE DCV 服务器传输的所有数据，将通过 HTTPS/TLS 连接发送所有内容以进行加密。

要配置证书，请参阅[管理 TLS 证书](#)。

NICE DCV 合规性验证

作为多个合规计划的一部分，第三方审计师评估 AWS 服务的安全 AWS 性和合规性。使用 NICE DCV 访问服务不会改变该服务的合规性。

有关特定合规计划范围内的 AWS 服务列表，请参阅[AWS 按合规计划划分的范围内 AWS 服务（按合分）](#)。有关一般信息，请参阅[AWS 合规计划 AWS](#)。

您可以使用下载第三方审计报告 AWS Artifact。有关更多信息，请参阅[在 AWS Artifact 中下载报告](#)。

您在使用 NICE DCV 时的合规责任取决于您的数据的敏感度、贵公司的合规目标以及适用的法律和法规。AWS 提供了以下资源来帮助实现合规性：

- [安全与合规性快速入门指南](#) — 这些部署指南讨论了架构注意事项，并提供了在上面部署以安全性和合规性为重点的基准环境的步骤。AWS
- [AWS 合 AWS 规资源](#) — 此工作簿和指南集合可能适用于您的行业和所在地区。
- [使用 AWS Config 开发人员指南中的规则评估资源](#) — 该 AWS Config 服务评估您的资源配置在多大程度上符合内部实践、行业准则和法规。
- [AWS Security Hub](#) — 此 AWS 服务可全面了解您的安全状态 AWS，帮助您检查是否符合安全行业标准 and 最佳实践。

NICE DCV 发行说明和文档历史记录

该页面提供 NICE DCV 发行说明和文档历史记录。

主题

- [NICE DCV 发行说明](#)
- [文档历史记录](#)

NICE DCV 发行说明

本节简要说明了 NICE DCV 的主要更新、功能版本和错误修复。所有更新是按发行日期排列的。我们经常更新文档以处理您发给我们的反馈。

主题

- [DCV 2023.1-16388 — 2024 年 3 月 5 日](#)
- [DCV 2023.1-16388 – 2023 年 12 月 19 日](#)
- [DCV 2023.1-16220 - 2023 年 11 月 9 日](#)
- [DCV 2023.0-15487 - 2023 年 6 月 29 日](#)
- [DCV 2023.0-15065 - 2023 年 5 月 3 日](#)
- [DCV 2023.0-15022 - 2023 年 4 月 21 日](#)
- [DCV 2023.0-14852 - 2023 年 3 月 28 日](#)
- [DCV 2022.2-14521 - 2023 年 2 月 17 日](#)
- [DCV 2022.2-14357 - 2023 年 1 月 18 日](#)
- [DCV 2022.2-14175 - 2022 年 12 月 21 日](#)
- [DCV 2022.2-14126 - 2022 年 12 月 9 日](#)
- [DCV 2022.2-13907 - 2022 年 11 月 11 日](#)
- [DCV 2022.1-13300 - 2022 年 8 月 4 日](#)
- [DCV 2022.1-13216 - 2022 年 7 月 21 日](#)
- [DCV 2022.1-13067 - 2022 年 6 月 29 日](#)
- [DCV 2022.0-12760 - 2022 年 5 月 23 日](#)

- [DCV 2022.0-12627 - 2022 年 5 月 19 日](#)
- [DCV 2022.0-12123 - 2022 年 3 月 23 日](#)
- [DCV 2022.0-11954 - 2022 年 2 月 23 日](#)
- [DCV 2021.3-11591 - 2021 年 12 月 20 日](#)
- [DCV 2021.2-11445 - 2021 年 11 月 18 日](#)
- [DCV 2021.2-11190 - 2021 年 10 月 11 日](#)
- [DCV 2021.2-11135 - 2021 年 9 月 24 日](#)
- [DCV 2021.2-11048 - 2021 年 9 月 1 日](#)
- [DCV 2021.1-10851 - 2021 年 7 月 30 日](#)
- [DCV 2021.1-10598 - 2021 年 6 月 10 日](#)
- [DCV 2021.1-10557 - 2021 年 5 月 31 日](#)
- [DCV 2021.0-10242 - 2021 年 4 月 12 日](#)
- [DCV 2020.2-9662 - 2020 年 12 月 4 日](#)
- [DCV 2020.2-9508 - 2020 年 11 月 11 日](#)
- [DCV 2020.1-9012 - 2020 年 9 月 30 日](#)
- [DCV 2020.1-9012 - 2020 年 8 月 24 日](#)
- [DCV 2020.1-8942 - 2020 年 8 月 3 日](#)
- [DCV 2020.0-8428 - 2020 年 4 月 16 日](#)
- [DCV 2019.1-7644 - 2019 年 10 月 24 日](#)
- [DCV 2019.1-7423 - 2019 年 9 月 10 日](#)
- [DCV 2019.0-7318 - 2019 年 8 月 5 日](#)
- [DCV 2017.4-6898 - 2019 年 4 月 16 日](#)
- [DCV 2017.3-6698 - 2019 年 2 月 24 日](#)
- [DCV 2017.2-6182 - 2018 年 10 月 8 日](#)
- [DCV 2017.1-5870 - 2018 年 8 月 6 日](#)
- [DCV 2017.1-5777 - 2018 年 6 月 29 日](#)
- [DCV 2017.0-5600 - 2018 年 6 月 4 日](#)
- [DCV 2017.0-5121 - 2018 年 3 月 18 日](#)

- [DCV 2017.0-4334 - 2018 年 1 月 24 日](#)
- [DCV 2017.0-4100 - 2017 年 12 月 18 日](#)

DCV 2023.1-16388 — 2024 年 3 月 5 日

内部版本号	更改和错误修复	
<ul style="list-style-type: none"> • nice-dcv-server: 16388 	<ul style="list-style-type: none"> • 修复了当显示缩放比例设置为不同于的值时，Windows Client 中的扩展出现的问题 100%。 • 修复了 Windows 客户端中相对鼠标模式和高 DPI 鼠标的问题。 	
<ul style="list-style-type: none"> • nice-dcv-client(Windows) : 8993 	<ul style="list-style-type: none"> • 修复了在 Windows 客户端中使用按键发布键盘组合时出现的问题。Shift 	
<ul style="list-style-type: none"> • nice-dcv-viewer (macOS) 6203 		
<ul style="list-style-type: none"> • nice-dcv-viewer (Linux) : 6203 		
<ul style="list-style-type: none"> • nice-xdcv : 565 		
<ul style="list-style-type: none"> • nice-dcv-gl: 1047 		
<ul style="list-style-type: none"> • nice-dcv- 		

内部版本号	更改和错误修复	
gltest: 325		
• nice-dcv-simple-external-身份验证器 : 228		

DCV 2023.1-16388 – 2023 年 12 月 19 日

内部版本号	更改和错误修复	
• nice-dcv-server: 16388	<ul style="list-style-type: none"> 修复了 Windows 代理启动时可能导致流式传输失败和日志记录过多的争用情况。 修复了在运行期间更改空闲超时设置时，dcv list-connections 所报告的上次交互时间。 	
• nice-dcv-client(Windows) : 8934	<ul style="list-style-type: none"> 修复了 Windows 服务器上的 NVIDIA GRID 驱动程序 528.89 的兼容性问题。 修复了 Web 客户端中可能导致流式传输失败的视频解码问题。 	
• nice-dcv-viewer (macOS) 6203	<ul style="list-style-type: none"> 修复了在服务器上禁用更改显示分辨率时，Windows 客户端上的多台显示器上的全屏显示问题。 修复了 Linux 和 macOS 客户端上摄像头分辨率的问题。 	
• nice-dcv-		

内部版本号	更改和错误修复	
viewer (Linux) : 6203	<ul style="list-style-type: none"> 修复了 Linux 和 macOS 客户端上双击和三击鼠标的问题。 修复了 Linux 和 macOS 客户端上的重定向问题 WebAuth。 	
nice-xdcv : 565		
nice-dcv-gl: 1047		
nice-dcv-gltest: 325		
nice-dcv-simple-ex ternal-身份验证器 : 228		

DCV 2023.1-16220 - 2023 年 11 月 9 日

内部版本号	新功能	更改和错误修复
nice-dcv-server: 16220	<p>NICE DCV 添加了以下功能：</p> <ul style="list-style-type: none"> 支持重定向来自在远程 Google Chrome 或 Microsoft Edge 浏览器中运行的网络应用程序的会话中 WebAuth N 个请求。重定向的请 	<ul style="list-style-type: none"> 在 Windows 上为剪贴板添加了透明图像支持。 修复了 Windows 上并发访问剪贴板的问题，该问题导致某些应

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> • nice-dcv-client(Windows) : 8908 • nice-dcv-viewer (macOS) 6125 • nice-dcv-viewer (Linux) : 6125 • nice-xdcv : 565 • nice-dcv-gl: 1047 • nice-dcv-gltest: 325 • nice-dcv-simple-external-身份验证器 : 228 	<p>求将传送到客户端，从而允许符合 FIDO2 标准的身份验证器（例如 YubiKey 或 Windows Hello）来验证用户身份。</p> <ul style="list-style-type: none"> • 新的适用于 Windows 主机的间接显示驱动程序（IDD）可以优化图形管道，并通过协议显著降低总体 CPU 使用率。 • 现在可以使用 Windows 性能计数器跟踪各种 DCV 协议指标，例如帧速率、网络带宽、CPU 使用率等，这可以帮助用户了解其网络和 DCV 协议的性能。 	<p>用程序无法成功执行剪切和粘贴操作。</p> <ul style="list-style-type: none"> • 修复了可能导致 Windows 上的 NICE DCV 服务器将显示器缩放系数重置为 100% 的问题 • 在 Windows 和 Linux 上添加了设置，以在控制台会话的用户注销和屏幕锁定时自动断开连接客户端 • 修复了音频堆栈中可能导致噪音和声音失真的问题。 • 可以在重新连接时恢复网络摄像头流式传输，而无需关闭服务器上的应用程序 • 改进了 Windows 本机客户端上高 DPI 鼠标的相对鼠标行为 • 修复了 macOS 原生客户端中的 SmartCard 支持问题 • 修复了 Linux 本机客户端上的高像素密度支持 • 改进了 Web 客户端和 Windows 本机客户端上的用户界面可访问性 • 修复了在 macOS 上使用 Web 客户端时的某些键盘布局限制 • 将第三方依赖项更新为最新版本 • Xdcv 已更新为 XServer 21.1.9 版本 • 删除了对 Windows Server 2012R2、Ubuntu 18.04 和 Suse Enterprise Linux 15SP4 的支持

内部版本号	新功能	更改和错误修复
-------	-----	---------

- 错误修复和性能改进

DCV 2023.0-15487 - 2023 年 6 月 29 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> • nice-dcv-server: 15487 • nice-dcv-client(Windows) : 8771 • nice-dcv-viewer (macOS) 5629 • nice-dcv-viewer (Linux) : 5629 • nice-xdcv : 551 • nice-dcv-gl: 1039 	<ul style="list-style-type: none"> • 修复了 Web 客户端中的一个问题，该问题可能导致在使用 Chrome 114 或更高版本时出现颜色错误。 • 修复了 NICE DCV 服务器和 Xdcv 的 el7 rpm 软件包，以避免卸载时出错。 • 修复了 Windows 服务器上的 NVIDIA GRID 驱动程序 528.89 的兼容性问题。 • 修复了一个问题，该问题可能导致剪贴板在某些 Windows 应用程序上无法正常工作。 • dcv-gl 软件包现在需要使用最新版本的 NICE DCV 服务器软件包，以确保安装或更新该软件包时配置正确无误。 • 修复了 Windows 客户端上的一个问题，该问题可能导致调整大小后使用错误的分辨率。 • 修复了 macOS 和 Linux 客户端中的 IPv6 地址支持。 • macOS 客户端现在允许将 Control + 单击配置为右键单击。 • Web 客户端现在允许在支持的浏览器上全屏显示时使用特殊键和组合键。 • 更新了 OpenSSL 第三方库。

内部版本号	更改和错误修复	
<ul style="list-style-type: none"> nice-dcv-glttest: 318 nice-dcv-simple-external-身份验证器 : 208 		

DCV 2023.0-15065 - 2023 年 5 月 3 日

内部版本号	更改和错误修复	
<ul style="list-style-type: none"> nice-dcv-server: 15065 nice-dcv-client(Windows) : 8671 nice-dcv-viewer (macOS) 5483 	<ul style="list-style-type: none"> 修复了可能导致无法释放许可证令牌的 <code>close-session</code> 问题。 修复了 macOS 原生客户端上的崩溃问题。BigSur 	

内部版本号	更改和错误修复	
<ul style="list-style-type: none"> nice-dcv-viewer (Linux) : 5483 nice-xdcv : 547 nice-dcv-gl: 1027 nice-dcv-gltest: 318 nice-dcv-simple-ex ternal-身份验证器 : 208 		

DCV 2023.0-15022 - 2023 年 4 月 21 日

内部版本号	更改和错误修复	
<ul style="list-style-type: none"> nice-dcv- 	<ul style="list-style-type: none"> 修复了一个并发性问题，该问题可能导致在调整屏幕大小后流式传输无法正常工作。 	

内部版本号	更改和错误修复	
server: 15022	<ul style="list-style-type: none"> 修复了 NICE DCV 服务器上的一种争用情况，该争用情况可能导致 QUIC 连接失败。 	
<ul style="list-style-type: none"> nice-dcv-client(Windows) : 8671 	<ul style="list-style-type: none"> 修复了 NICE DCV 服务器中与具有隐藏光标的应用程序相关的崩溃。 修复了 Windows 服务器上的日语键盘输入问题。 改进了网络摄像头流的音频/视频同步。 	
<ul style="list-style-type: none"> nice-dcv-viewer (macOS) 5456 	<ul style="list-style-type: none"> 更新了 ICU 和 libxml2 第三方库。 将 Xdcv 更新为 XServer 版本 21.1.8，并修复了可能导致虚拟会话无法启动的 XKB 问题。 修复了一个问题，该问题可能导致 	
<ul style="list-style-type: none"> nice-dcv-viewer (Linux) : 5456 	<ul style="list-style-type: none"> Windows、macOS 和 Linux 本机客户端上的视频解码失败。 修复了 macOS 和 Linux 本机客户端上的设置问题。 	
<ul style="list-style-type: none"> nice-xdcv : 547 		
<ul style="list-style-type: none"> nice-dcv-gl: 1027 		
<ul style="list-style-type: none"> nice-dcv-gltest: 318 		
<ul style="list-style-type: none"> nice-dcv-simple-external- 		

内部版本号	更改和错误修复
身份验证器： 206	

DCV 2023.0-14852 - 2023 年 3 月 28 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 14852 nice-dcv-client(Windows) : 8655 nice-dcv-viewer (macOS) 5388 nice-dcv-viewer (Linux) : 5388 nice-xdcv : 527 	<p>NICE DCV 添加了以下功能：</p> <ul style="list-style-type: none"> 在选定的显示器上为 macOS 和 Linux 上的 NICE DCV 客户端添加了全屏支持。 为所有客户端添加了以拖放方式启动文件上传的支持。 添加了 Red Hat Enterprise Linux 9、Rocky Linux 9 和 CentOS Stream 9。 为 Linux 上的 NICE DCV 服务器添加了时区重定向支持。 	<ul style="list-style-type: none"> 修复了 QUIC 传输中的一些问题，这些问题可能导致不正确的带宽估计和视觉失真。 更新了 macOS 和 Linux 客户端的用户界面。 现在，Windows 安装程序在用户可见的应用程序名称中一致地使用 NICE DCV。 重新设计了 Windows 上的剪贴板支持实施以提高可靠性。 修复了在 Windows 上使用德语键盘布局时的 Caps Lock 键问题。

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-gl: 1022 nice-dcv-gltest: 318 nice-dcv-simple-external-身份验证器 : 206 		

DCV 2022.2-14521 - 2023 年 2 月 17 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 14521 nice-dcv-client(Windows) : 8570 nice-dcv-viewer (macOS) : 5125 nice-dcv-viewer (Linux) : 4804 nice-xdcv : 519 nice-dcv-gl: 1012 nice-dcv-gltest: 307 nice-dcv-simple-external-身份验证器 : 198 	<ul style="list-style-type: none"> 修复了 macOS 客户端上的日语和西班牙语键盘问题。 修复了 Windows NICE DCV 服务器上的数字键盘键问题。 修复了 QUIC 连接的内存泄漏。 提高了使用旧视频驱动程序时的 Windows NICE DCV 客户端稳定性。 更新了 OpenSSL 和 libsoup 第三方库。 将 Xdcv 更新为 XServer 版本 21.1.7。

DCV 2022.2-14357 - 2023 年 1 月 18 日

内部版本号	更改和错误修复
<ul style="list-style-type: none">nice-dcv-server: 14357nice-dcv-client(Windows) : 8522nice-dcv-viewer (macOS) : 4804nice-dcv-viewer (Linux) : 4804nice-xdcv : 487nice-dcv-gl: 1012nice-dcv-gltest: 307nice-dcv-simple-external-身份验证器 : 198	<ul style="list-style-type: none">修复了 Suse Linux 12 上的虚拟会话崩溃问题，该崩溃是从 Suse 软件包的最新更新开始发生的。修复了 DCV-GL 中与 X Pixmap 处理相关的内存泄漏。将 DCV-GL 与 xrestop 工具集成在一起，以便将 X Pixmap 与相应的进程相关联。改进 Windows 服务器上的网络摄像头和音频重定向，以使其与 Windows 的本机行为更一致：在发生操作系统事件时，流不会中断。改进 Windows NICE DCV 客户端处理输入法的方式。修复了 Windows NICE DCV 客户端中的剪贴板问题，该问题与仅将回车符作为行分隔符的文本相关。

DCV 2022.2-14175 - 2022 年 12 月 21 日

内部版本号	更改和错误修复
<ul style="list-style-type: none">nice-dcv-server: 14175nice-dcv-client(Windows) : 8472nice-dcv-viewer (macOS) : 4804nice-dcv-viewer (Linux) : 4804nice-xdcv : 487nice-dcv-gl: 983nice-dcv-gltest: 307nice-dcv-simple-external-身份验证器 : 198	<ul style="list-style-type: none">修复了使用 WebSocket 连接时服务器中文件描述符泄漏的问题。Xdcv 已更新为 XServer 21.1.6 版本。

DCV 2022.2-14126 - 2022 年 12 月 9 日

内部版本号	更改和错误修复
<ul style="list-style-type: none">• nice-dcv-server: 14126• nice-dcv-client(Windows) : 8472• nice-dcv-viewer (macOS) : 4804• nice-dcv-viewer (Linux) : 4804• nice-xdcv : 481• nice-dcv-gli: 983• nice-dcv-glttest: 301• nice-dcv-simple-external-身份验证器 : 198	<ul style="list-style-type: none">• 修复了 Windows 服务器中使用韩语键盘时的问题。• 修复了 Windows 服务器上可能导致 Windows 11 挂起的 USB 重定向问题。• 修复了在“rotate”参数设置为 0 时服务器上的日志轮换问题。• 修复了 macOS 和 Linux 客户端中的一个问题，该问题可能导致流在特定网络条件下冻结。• 修复了 Windows 本机客户端在全屏显示时无法正确调整大小的问题。• 修复了 macOS 和 Linux 客户端中的一个问题，该问题可能导致文件上传期间崩溃。• 修复了 macOS 客户端中的一个问题，该问题可能导致音频停止工作。• 修复了 Linux 客户端中的一个问题，该问题可能导致使用 NVIDIA GPU 时崩溃。• 修复了 Web 客户端中的一个问题，该问题可能导致时区重定向 UI 与服务器不同步。• 修复了 Web 客户端中的一个问题，该问题可能导致无法加载会话后页面。• 更新了 libTIFF 和 MIT-Kerberos 开源依赖项。

DCV 2022.2-13907 - 2022 年 11 月 11 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> • nice-dcv-server: 13907 • nice-dcv-client(Windows) : 8427 • nice-dcv-viewer (macOS) 4653 • nice-dcv-viewer (Linux) : 4653 • nice-xdcv : 481 • nice-dcv-gl: 983 • nice-dcv-gltest: 301 	<p>NICE DCV 添加了以下功能：</p> <ul style="list-style-type: none"> • 在选定的显示器上为 Windows 上的 NICE DCV 客户端添加了全屏支持。 • 为 macOS 上的本机客户端添加了高像素密度显示支持。 • 为 macOS 和 Linux 上的 NICE DCV 客户端添加了打印机重定向。 • 为 Windows 上的 NICE DCV 服务器添加了时区重定向支持。 • 为 Ubuntu 22.04 添加了 GNOME-Shell 扩展，以支持控制台会话的单点登录。 • 在使用开源驱动程序时，在 AMD GPU 上添加了基于 VAAPI 的编码器。 	<ul style="list-style-type: none"> • 将 Web 客户端用户界面更新为 Cloudscape 设计风格。 • 修复了 Agent 中由客户端重新连接触发的内存泄漏。 • 在 Ubuntu 20.04 上使用虚拟会话时，为使用 GDM3 的系统添加了支持。 • 修复了 Ubuntu 20.04 上的虚拟会话中间歇性导致黑屏的问题。 • 修复了 Web 客户端中的一个问题，该问题导致更改选项卡时剪贴板更新丢失。 • 修复了数字键盘的 Enter 键问题。

内部版本号	新功能	更改和错误修复
• nice-dcv-simple-external-身份验证器 : 198		

DCV 2022.1-13300 - 2022 年 8 月 4 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> • nice-dcv-server: 13300 • nice-dcv-client(Windows) : 8261 • nice-dcv-viewer (macOS) : 4279 • nice-dcv-viewer (Linux) : 4251 • nice-xdcv : 433 • nice-dcv-gli: 973 • nice-dcv-glitest: 295 • nice-dcv-simple-external-身份验证器 : 193 	<ul style="list-style-type: none"> • 在多个协作者连接到会话时，不会自动解锁 Windows。 • 修复了服务器无法加载指定证书文件的问题。 • 修复了 macOS 客户端上导致音频失真的问题。

DCV 2022.1-13216 - 2022 年 7 月 21 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> • nice-dcv-server: 13216 • nice-dcv-client(Windows) : 8261 • nice-dcv-viewer (macOS) : 4251 • nice-dcv-viewer (Linux) : 4251 	<ul style="list-style-type: none"> • 修复了所有客户端中的一个问题，该问题导致无法连接到 NICE DCV Server 2019.1 和更早版本。

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-xdcv : 433 nice-dcv-gl: 966 nice-dcv-gltest: 295 nice-dcv-simple-external-身份验证器 : 193 	<ul style="list-style-type: none"> 修复了 Windows 服务器上的 SmartCard 重定向问题。 修复了一个问题，该问题可能导致在连接到具有 GPU 的主机上的 NICE DCV 服务器时流式传输失败。

DCV 2022.1-13067 - 2022 年 6 月 29 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 13067 nice-dcv-client(Windows) : 8248 nice-dcv-viewer (macOS) 4241 nice-dcv-viewer (Linux) : 4241 nice-xdcv : 433 	<p>NICE DCV 添加了以下功能：</p> <ul style="list-style-type: none"> 为服务器添加了对 Ubuntu 22.04 以及 Rocky Linux 8.5 和更高版本的支持。 为本机客户端添加了 Ubuntu 22.04 支持。 改进了 Windows、macOS 和 Linux 本机客户端的协作体验。 	<ul style="list-style-type: none"> 提高了非 GPU 服务器的性能，总体 CPU 使用率降低高达 30%。 现在，可以在指定时间间隔或大小限制的设置中配置日志轮换。 修复了可能导致初始握手失败的 QUIC 传输问题。 修复了一个问题，该问题可能导致 Linux 服务器上的相对鼠标移动无法按某些应用程序的预期方式工作。

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-gl: 966 nice-dcv-gltest: 295 nice-dcv-simple-external-身份验证器 : 193 		

DCV 2022.0-12760 - 2022 年 5 月 23 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 12760 nice-dcv-client(Windows) : 8145 nice-dcv-viewer (macOS) : 4131 nice-dcv-viewer (Linux) : 4131 nice-xdcv : 424 nice-dcv-gl: 961 nice-dcv-gltest: 291 nice-dcv-simple-external-身份验证器 : 188 	<p>更改 :</p> <p>修复了在指定 web-url-path 选项时无法成功连接 Web 客户端的问题。</p>

DCV 2022.0-12627 - 2022 年 5 月 19 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 12627 nice-dcv-client(Windows) : 8145 nice-dcv-viewer (macOS) : 4131 nice-dcv-viewer (Linux) : 4131 nice-xdcv : 424 nice-dcv-gi: 961 nice-dcv-gltest: 291 nice-dcv-simple-external-身份验证器 : 188 	<p>更改：</p> <ul style="list-style-type: none"> 修复了 QUIC 传输中的一些问题，这些问题可能导致不正确的带宽估计和视觉失真。 修复了 Windows 服务器安装程序中的音频服务问题，该问题可能导致更新过程失败。 修复了 Windows 客户端安装程序中的 USB 处理问题，该问题可能导致卸载过程失败。 修复了在 macOS 和 Linux 客户端中保存屏幕截图时出现的问题。 更新了 OpenSSL、zlib 和 gdk-pixbuf 第三方库。

DCV 2022.0-12123 - 2022 年 3 月 23 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 12123 nice-dcv-client(Windows) : 7920 nice-dcv-viewer (macOS) : 3973 nice-dcv-viewer (Linux) : 3973 nice-xdcv : 424 nice-dcv-gi: 961 nice-dcv-gltest: 291 nice-dcv-simple-external-身份验证器 : 188 	<p>NICE DCV 添加了以下功能：</p> <ul style="list-style-type: none"> 为 macOS 和 Linux 客户端添加了启用高色彩精度的选项。 	<p>更改：</p> <ul style="list-style-type: none"> 改进了使用 QUIC 传输时的带宽估计和图像质量。 <p>修复：</p> <ul style="list-style-type: none"> 修复了使用 NVIDIA 驱动程序 510.xx 时 Linux 上的控制台会话中的视觉失真。 修复了 Windows 本机客户端中通过蓝牙连接的 DualShock 4 个控制器的问题。

内部版本号	新功能	更改和错误修复
		<ul style="list-style-type: none"> 修复了启用网络摄像头时 macOS 客户端中可能发生的崩溃。

DCV 2022.0-11954 - 2022 年 2 月 23 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 11954 nice-dcv-client(Windows) : 7866 nice-dcv-viewer (macOS) : 3929 nice-dcv-viewer (Linux) : 3929 nice-xdcv : 424 nice-dcv-gl: 961 nice-dcv-gltest: 291 nice-dcv-simple-external-身份验证器 : 188 	<p>NICE DCV 添加了以下功能：</p> <ul style="list-style-type: none"> 适用于 Windows 服务器和 Windows 本机客户端的游戏控制器支持。 NICE DCV Web Client 现在可以利用支持 WebCodecs 它的浏览器。 为 Windows 和 Web 客户端添加了启用高色彩精度的选项。 改进了协作体验：在有人加入会话时通知用户 将 CentOS 8 Stream 添加到支持的 Linux 发行版列表中。 	<p>更改：</p> <ul style="list-style-type: none"> 现在无需重新启动 NICE DCV 服务器即可更新 TLS 证书。 现在可以将 NICE DCV 服务器配置为侦听特定网络接口或者特定 IPv4 或 IPv6 地址。 现在还在 Linux 系统上自动配置“DCV 打印机”。 现在以更高的优先级执行 Windows 上的 NICE DCV 进程。 <p>修复：</p> <ul style="list-style-type: none"> 修复了在使用具有 GPU 的实例的情况下重新启动 Windows 2016 上的 Agent 时发生的崩溃。 修复了在从 NICE DCV 客户端重定向某些 USB 设备的情况下注销会话时发生的 Windows 崩溃。

内部版本号	新功能	更改和错误修复
		<ul style="list-style-type: none"> 在执行授权检查时，规范化包含 Windows 域的用户名。 改进了 Windows 客户端中的相对鼠标模式。 修复了 CapsLock 密钥同步的问题。

DCV 2021.3-11591 - 2021 年 12 月 20 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 11591 nice-dcv-client(Windows) : 7801 nice-dcv-viewer (macOS) : 3829 nice-dcv-viewer (Linux) : 3829 nice-xdcv : 415 nice-dcv-gl: 952 nice-dcv-gltest: 284 nice-dcv-simple-external-身份验证器 : 176 	<p>NICE DCV 添加了以下功能：</p> <ul style="list-style-type: none"> 已更新 Web Client 的用户界面。 现在支持 EC2 G5 和 G5g 实例。 Windows Server 2022 和 Windows 11 现在是支持的操作系统。 	<ul style="list-style-type: none"> Linux 虚拟会话的初始化脚本不再加载用户的 bash 配置文件，从而避免环境变量覆盖系统默认值的重复性问题。 nice-dcv-ext-authenticator 现在需要 Python 3。

DCV 2021.2-11445 - 2021 年 11 月 18 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 11445 nice-dcv-client(Windows) : 7792 nice-dcv-viewer (macOS) : 3797 	<p>修复：</p> <ul style="list-style-type: none"> 修复了导致客户端无法在 macOS Monterey 上正常工作的问题。

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-viewer (Linux) : 3797 nice-xdcv : 411 nice-dcv-gl: 946 nice-dcv-gltest: 279 nice-dcv-simple-external-身份验证器 : 160 	<ul style="list-style-type: none"> 提高了 Windows 上的服务器的安全性。 修复了一个错误，该错误可能导致无法正确应用多显示器布局，特别是在使用 Web Client 时。 修复了一个问题，该问题可能导致 Delete 键无法在某些 Windows 应用程序中正常工作。 将 Linux 上的 Web 客户端软件包标记为与旧版本的服务器软件包（包括 Web 客户端本身）互斥。

DCV 2021.2-11190 - 2021 年 10 月 11 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 11190 nice-dcv-client(Windows) : 7788 nice-dcv-viewer (macOS) : 3776 nice-dcv-viewer (Linux) : 3776 nice-xdcv : 411 nice-dcv-gl: 946 nice-dcv-gltest: 279 nice-dcv-simple-external-身份验证器 : 160 	<p>修复：</p> <ul style="list-style-type: none"> 修复了 Windows 客户端中的一个问题，该问题导致用户在连接到具有过期证书的服务器时无法关闭证书验证对话框。 修复了触控笔上的中键单击按钮在本机客户端上无法正常工作的问题。 修复了 Xdcv 中的性能下降问题，该问题导致无法加载旧 X11 字体。 修复了 macOS 和 Linux 客户端中的一个问题，即，使用具有死键的键盘布局时，键盘组合键无法正常工作。

DCV 2021.2-11135 - 2021 年 9 月 24 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 11135 	<p>修复：</p>

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-client(Windows) : 7781 nice-dcv-viewer (macOS) : 3740 nice-dcv-viewer (Linux) : 3740 nice-xdcv : 408 nice-dcv-gl: 944 nice-dcv-gltest: 279 nice-dcv-simple-external-身份验证器 : 160 	<ul style="list-style-type: none"> 修复了 QUIC 数据包大小协商问题，该问题可能导致在使用 2021.2 客户端连接旧服务器时出现连接和性能问题。 修复了 NVIDIA 设备选择错误，该错误可能导致 NVENC 编码器失败。 修复了具有 Windows 和 NVIDIA GPU 的计算机上的问题，这些问题可能导致压缩失真和颜色准确性失真。 修复了 Linux 服务器上的修饰键错误，该错误可能导致某些键盘组合键无法正常工作。 修复了具有 M1 CPU 的计算机上的 macOS 客户端的性能下降问题。 修复了 macOS 客户端中的一个错误，该错误导致某些键盘组合键无法正常工作。 修复了 Linux 虚拟会话中处理触摸事件的方式问题，该问题可能导致会话终止。

DCV 2021.2-11048 - 2021 年 9 月 1 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 11048 nice-dcv-client(Windows) : 7774 nice-dcv-viewer (macOS) : 3690 nice-dcv-viewer (Linux) : 3690 nice-xdcv : 406 nice-dcv-gl: 944 nice-dcv-gltest: 279 	<p>NICE DCV 添加了以下功能：</p> <ul style="list-style-type: none"> 改进了 Web 客户端剪贴板。通过这些改进，您现在可以使用 NICE DCV Web Client 在 Google Chrome 和 Microsoft Edge 上复制和粘贴 PNG 格式图像。 适用于 Windows 和 macOS 客户端的屏幕截图阻止功能。该功能禁止用户获取 NICE DCV 会话内容屏幕截 	<p>更改：</p> <ul style="list-style-type: none"> NICE DCV Web Client 现在是 Linux 上的单独软件包，也是 Windows 安装程序中的可选组件。通过该更改，客户可以决定是否部署该 Web 客户端。 在使用 NVENC 编码器时，现在支持 H.264 高级配置文件 通过将 NVENC 编码器与 NVIDIA GPU 一起使用，您

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-simple-external-身份验证器：160 	<p>图，从而添加一层额外的安全保护。如果启用，用户捕获的任何屏幕截图都会导致空白屏幕。</p> <ul style="list-style-type: none"> 提高了流媒体质量。使用 QUIC 协议时，流媒体质量特别通过更好 build-to-losless 的性能而得到改善。 添加了指定客户端行为的 certificate-validation-policy 选项。在服务器提供不受信任的 X.509 证书（例如自签名证书）时，您可以使用该选项。 可以更改运行时在音频驱动程序中配置的声道数。 在 dcvinput Xorg 模块中添加了 Pressure2K 选项。您可以使用该选项将触控笔的压力灵敏度范围从 0-65335 更改为 0-2048，以与 Mari 和 Nuke 等应用程序兼容。 增加了对谷歌浏览器和微软 Edge 上实验性 WebCodecs API 的支持。在浏览器中启用该 API 时，NICE DCV Web Client 可以使用该 API 加速视频解码并提供更高的帧速率。 	<p>可以减少使用的带宽，同时保持相同的图像质量。</p> <ul style="list-style-type: none"> 在具有多个 GPU 的计算机上，NICE DCV 服务器现在使用所有可用的 GPU 进行压缩。 NICE DCV 附带的所有 Windows 驱动程序现已通过 WHQL 认证。 OpenSSL 已更新为 1.1.1 版本。 Xdcv 已更新为 XServer 1.20.13 版本。 <p>修复：</p> <ul style="list-style-type: none"> 修复了 macOS 客户端上的数字键盘按键问题。 修复了一个问题，该问题导致某些 USB 设备（例如游戏手柄）无法正确重定向到 Windows 服务器。 修复了一个错误，即，在断开连接时，无法正确释放修饰键。 修复了使用 Ubuntu 20.04 和 Intel GPU 时在 Linux 本机客户端中发生的崩溃。

DCV 2021.1-10851 - 2021 年 7 月 30 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> • nice-dcv-server: 10851 • nice-dcv-client(Windows) : 7744 • nice-dcv-viewer(macOS) : 3590 • nice-dcv-viewer(Linux) : 3560 • nice-xdcv : 392 • nice-dcv-gli: 937 • nice-dcv-glttest: 275 • nice-dcv-simple-external-身份验证器 : 154 	<p>更改 :</p> <ul style="list-style-type: none"> • 我们提高了 Windows、Linux 和 macOS 客户端的稳定性。 <p>修复 :</p> <ul style="list-style-type: none"> • 修复了一个错误，该错误导致 Windows 服务器上的 AMD 和 NVIDIA 图形适配器出现屏幕闪烁。 • 修复了连接到运行多个会话的 Linux 服务器时出现的偶发性问题。 • 修复了与处理 Linux 服务器上的非西方键盘布局相关的错误。 • 修复了 Windows 客户端中的连接窗口上的视觉失真。 • 修复了 Windows 上的 USB 重定向驱动程序中的多个错误并改进了设备兼容性。

DCV 2021.1-10598 - 2021 年 6 月 10 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> • nice-dcv-server: 10598 • nice-dcv-client(Windows) : 7713 • nice-dcv-viewer(macOS) : 3473 • nice-dcv-viewer(Linux) : 3473 • nice-xdcv : 392 • nice-dcv-gli: 937 • nice-dcv-glttest: 275 	<ul style="list-style-type: none"> • 修复了服务器的 Windows 安装程序在 session owner 字段中预填充当前用户的问题。 • 提高了 macOS 和 Linux 客户端的总体稳定性。

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-simple-external-身份验证器 : 154 	

DCV 2021.1-10557 - 2021 年 5 月 31 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 10557 nice-dcv-client(Windows) : 7713 nice-dcv-viewer (macOS) : 3450 nice-dcv-viewer (Linux) : 3454 nice-xdcv : 392 nice-dcv-gl: 937 nice-dcv-gltest: 275 nice-dcv-simple-external-身份验证器 : 154 	<ul style="list-style-type: none"> NICE DCV 添加了客户端选项，以便在连接到具有 GPU 的服务器时实现准确的音频/视频同步。 NICE DCV 在 Linux 控制台会话上添加了麦克风支持。 	<ul style="list-style-type: none"> 降低了没有 GPU 的 Windows 服务器主机上的 CPU 使用率。 修复了在 macOS 和 Linux 客户端中读取 .dcv 连接文件的问题。 为不支持硬件加速解码的 macOS 计算机添加了改用软件解码的功能。 为 macOS 客户端读取系统密钥链存储中的 CA 证书添加了支持。

DCV 2021.0-10242 - 2021 年 4 月 12 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 10242 nice-dcv-client(Windows) : 7643 nice-dcv-viewer (macOS) : 3186 nice-dcv-viewer (Linux) : 3294 nice-xdcv : 380 	<ul style="list-style-type: none"> 为 Windows NICE DCV 服务器添加了网络摄像头重定向支持。 为 Linux NICE DCV 服务器添加了打印机重定向支持。 为 macOS 客户端上的 M1 处理器添加了支持。 	<ul style="list-style-type: none"> 优化了具有 NVIDIA GPU 的 Linux 服务器和 Amazon EC2 实例上的 GPU 和 CPU 资源使用情况。 为 Linux NICE DCV 服务器添加了在 Amazon EC2 G4ad 实例上使用 AMD GPU

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-gl: 912 nice-dcv-gltest: 266 nice-dcv-simple-external-身份验证器 : 134 	<ul style="list-style-type: none"> 为 macOS 客户端添加了多显示器显示支持。 	<ul style="list-style-type: none"> 进行 GPU 加速视频编码的支持。 优化了音频处理以减少音频延迟 如果在服务器上启用了 QUIC 协议，则将客户端的默认设置更改为 QUIC 协议。 在 DCV 命令行工具中添加了新的 get-screenshot 命令。 添加了使用 close-session 命令的 --logout-user 选项的强制注销选项。您可以在关闭控制台会话时使用该选项。

DCV 2020.2-9662 - 2020 年 12 月 4 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 9662 nice-dcv-client(Windows) : 7490 nice-dcv-viewer (macOS) : 2117 nice-dcv-viewer (Linux) : 3007 nice-xdcv : 359 nice-dcv-gl: 881 nice-dcv-gltest: 259 nice-dcv-simple-external-身份验证器 : 125 	<ul style="list-style-type: none"> 增强了 Web 浏览器客户端中使用的安全协议。 提高了与 Windows 客户端一起使用的 Amazon EC2 G4ad 实例的性能和可靠性。 修复了 Windows 客户端连接设置对话框中的端口选择问题。

DCV 2020.2-9508 - 2020 年 11 月 11 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> • nice-dcv-server: 9508 • nice-dcv-client(Windows) : 7459 • nice-dcv-viewer (macOS) : 2078 • nice-dcv-viewer (Linux) : 1737 • nice-xdcv : 359 • nice-dcv-gl: 881 • nice-dcv-gltest: 259 • nice-dcv-simple-external-身份验证器 : 125 	<ul style="list-style-type: none"> • 添加了 QUIC (基于 UDP) 传输协议支持。 • 添加了 SLES 15 和 Ubuntu 20.4 支持。 • 为 Windows NICE DCV 服务器添加了智能卡支持。 	<ul style="list-style-type: none"> • 对于在具有 NVIDIA GPU 的服务器和 EC2 实例上托管的控制台会话，将 NICE DCV 帧速率限制器的默认值更改为 60 FPS。 • 优化了在具有 NVIDIA GPU 的 EC2 实例上托管的 Windows NICE DCV 服务器上使用的 GPU 和 CPU 资源。 • 添加了 list-endpoints NICE DCV CLI 命令。这会列出当前活动终端节点。 • version NICE DCV CLI 命令支持 --json 选项。 • 在 Linux 服务器上，create-session NICE DCV CLI 命令现在支持 --disable-login-monitor 选项。 • 改进了与 Linux NICE DCV 服务器上的不同显示管理器的兼容性。 • 修复了几个键盘输入处理问题。 • 现在动态重新加载 USB 设备允许列表文件。

DCV 2020.1-9012 - 2020 年 9 月 30 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 9012 nice-dcv-client (Windows) : 7342 nice-dcv-viewer (macOS) : 1986 nice-dcv-viewer (Linux) : 1545 nice-xdcv : 338 nice-dcv-gl: 840 nice-dcv-gltest: 246 nice-dcv-simple-external-身份验证器 : 111 	<ul style="list-style-type: none"> 添加了缺少的 macOS 客户端图标。

DCV 2020.1-9012 - 2020 年 8 月 24 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 9012 nice-dcv-client (Windows) : 7342 nice-dcv-viewer (macOS) : 1910 nice-dcv-viewer (Linux) : 1545 nice-xdcv : 338 nice-dcv-gl: 840 nice-dcv-gltest: 246 nice-dcv-simple-external-身份验证器 : 111 	<ul style="list-style-type: none"> 修复了 Amazon S3 在 AWS GovCloud 区域中的访问问题 改进了基于 Web 的客户端

DCV 2020.1-8942 - 2020 年 8 月 3 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 8942 	<ul style="list-style-type: none"> Linux NICE DCV 服务器现在支持 AWS 基于 	<ul style="list-style-type: none"> 在没有 GPU 的 Amazon EC2 实例上为新的 NICE

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> • nice-dcv-client (Windows) : 7342 • nice-dcv-viewer (macOS) : 1910 • nice-dcv-viewer (Linux) : 1545 • nice-xdcv : 338 • nice-dcv-gl: 840 • nice-dcv-gltest: 246 • nice-dcv-simple-external-身份验证器 : 111 	<p>Graviton2 的 Arm 实例，例如 m6g、c6g 和 r6g。有关更多信息，请参阅 AWS Graviton 处理器。</p> <ul style="list-style-type: none"> • 在 Linux NICE DCV 服务器上添加了 RHEL 8.x 和 CentOS 8.x 支持。 • 在使用 Windows NICE DCV 服务器和 Windows NICE DCV 客户端时，添加了打印机重定向支持。 • 在 macOS 和 Linux 本机 NICE DCV 客户端上添加了具有压力敏感度的触控笔支持。 • 为 Linux NICE DCV 服务器和 Linux NICE DCV 客户端添加了环绕声 5.1 支持。 • 为 Linux NICE DCV 本机客户端添加了触摸屏支持。 • 您现在可以将自定义名称与 NICE DCV 会话相关联。 • 在 macOS 本机 NICE DCV 客户端上支持硬件加速解码和渲染。 	<p>DCV 虚拟显示驱动程序添加了支持。</p> <ul style="list-style-type: none"> • 解决了一个问题，该问题导致在使用 NVENC 编码器时由于色彩空间转换而出现视觉失真。 • dcv list-sessions 命令现在始终包含控制台会话（如果存在） • 在较新的 Linux 发行版上，控制台会话 Agent 现在作为桌面会话的一部分启动，以更好地支持较新的显示管理器，例如 GDM3。 • 现在，在激活具有 dcv:// 模式的 URL 时，将自动打开本机客户端。 • 改进了 macOS 本机客户端和 Web 客户端处理键盘修饰键的方式。 • 改进了 DCV-GL 中的视觉和 fbconfig 选择，以改进对某些应用程序的支持。 • 降低了文件传输期间的 CPU 使用率 • 改进了 Web 浏览器客户端中的 WebGL 渲染，以减少使用的资源。

DCV 2020.0-8428 - 2020 年 4 月 16 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 8428 nice-dcv-client (Windows) : 7238 nice-dcv-viewer (macOS) : 1716 nice-dcv-viewer (Linux) : 1358 nice-xdcv : 296 nice-dcv-gl: 759 nice-dcv-gltest: 229 nice-dcv-simple-external-身份验证器 : 87 	<ul style="list-style-type: none"> 在 Linux 服务器上添加了屏幕触控笔和触摸支持。 在 Windows 服务器上为 Windows 本机客户端添加了 7.1 环绕声播放支持。 在 Linux 本机客户端上添加了硬件加速和触控笔支持。 添加了新的 API 命令以设置服务器端的显示布局。 在 Microsoft Edge 浏览器 (79.0.309 或更高版本) 上添加了多显示器 Web 客户端显示支持。 	<ul style="list-style-type: none"> 在全屏模式下，现在可以隐藏 Windows 客户端上的工具栏控制。 在 Windows 本机客户端上添加了 NTLM 代理支持。 改进了对使用 NVIDIA 适配器的 Windows 无头物理主机的支持。 删除了对传统 NVIDIA NvIFR 库的支持。 增加了对最新的 Windows 10 上的 Windows 图形捕获 API 的支持。 在 EC2 实例上为 Amazon EC2 实例元数据服务 (I MDS) v2 添加了支持。 DCV CLI 提供了新的 on-client-connected /disconnected 命令来检测客户端与会话连接或断开连接的时间。 增加了对指定主机名以绑定外部身份验证器证书的支持。 DCV-GL 现在在支持 GL 厂商中立的分派库 (GLvnd) 的系统上使用它。

DCV 2019.1-7644 - 2019 年 10 月 24 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 7644 nice-dcv-client (Windows) : 7114 nice-dcv-viewer (macOS) : 1535 nice-dcv-viewer (Linux) : 1124 nice-xdcv : 226 nice-dcv-gl: 544 nice-dcv-gltest: 220 nice-dcv-simple-external-身份验证器 : 77 	<ul style="list-style-type: none"> 修复了 NICE EnginFrame 和其他会话管理器使用的集成 API 中的一个问题。 修复了 32 位版本的 Windows 本机客户端的问题。

DCV 2019.1-7423 - 2019 年 9 月 10 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 7423 nice-dcv-client (Windows) : 7087 nice-dcv-viewer (macOS) : 1535 nice-dcv-viewer (Linux) : 1124 nice-xdcv : 226 nice-dcv-gl: 544 nice-dcv-gltest: 220 nice-dcv-simple-external-身份验证器 : 77 	<ul style="list-style-type: none"> 改进了 Windows 上的 DCV 服务器的安全性。 修复了 Linux 上的 Autodesk Maya 的渲染问题。 增加了与键盘操作相关的改进和错误修复。

DCV 2019.0-7318 - 2019 年 8 月 5 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 7318 	<ul style="list-style-type: none"> Web 客户端支持多显示器。 	<ul style="list-style-type: none"> 为 Windows 触摸输入添加了改进的压力灵敏度兼容性。

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> • nice-dcv-client (Windows) : 7059 • nice-dcv-viewer (macOS) : 1530 • nice-dcv-viewer (Linux) : 968 • nice-xdcv : 224 • nice-dcv-gl: 529 • nice-dcv-gltest: 218 • nice-dcv-simple-external-身份验证器 : 72 	<ul style="list-style-type: none"> • Windows Server 2019 上支持触控笔输入。 • macOS 和 Linux 本机客户端上音频输入/输出。 • Linux 服务器上的增强剪贴板功能 (中键单击粘贴) 。 	<ul style="list-style-type: none"> • 改进了 Windows 上具有异构图形适配器的系统的行为。 • 减少了检测非活动连接所需的时间 (例如 , 响应客户端上从有线网络到 Wi-Fi 网络的更改) 。 • 在 Linux 上无法捕获光标图标时 , 减少了日志记录。 • 支持在虚拟会话 Xdcv 组件中禁用复合扩展。 • 添加了限制并发虚拟会话数量的选项。 • 改进了安装 Bash 5 的系统的脚本兼容性。 • 更改了 OpenGL 和 GLES 的默认设置 , 以便在 Linux 客户端上自动检测并使用这些设置进行渲染。 • 在 GL 窗口的可见性发生变化时 , 更新 DCV-GL 屏幕缓冲区。 • 修复了 Windows 7 上的 Windows 客户端中的鼠标滚轮检测。 • 修复了导致 Windows 客户端在某些 Windows 7 系统上加载库时出现故障的问题。 • 改进了横向打印文档时 Windows 客户端上的打印。

DCV 2017.4-6898 - 2019 年 4 月 16 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> • nice-dcv-server: 6898 • nice-dcv-client (Windows) : 6969 • nice-dcv-viewer (macOS) : 1376 • nice-dcv-viewer (Linux) : 804 • nice-xdcv : 210 • nice-dcv-gi: 490 • nice-dcv-glttest: 216 • nice-dcv-simple-external-身份验证器 : 70 	<ul style="list-style-type: none"> • 适用于 macOS 的新本机客户端。 	<ul style="list-style-type: none"> • Windows 本机客户端现在使用硬件加速进行解码和渲染 (如果在系统中提供该功能)。 • 现在 , dcv 命令行工具在 Windows 和 Linux 上使用相同的选项和输出格式。 • dcv 命令行工具现在报告有关许可证的信息。 • 客户端现在会在因不活动导致断开连接之前向用户显示警告。 • 改进了对使用多个修饰符的键盘组合的支持。 • 针对通信失败改进了与 Reprise License Manager 的交互的稳健性。 • 在 Linux 上 , dcvusers 命令行工具现在默认为将数据保存到 dcv 用户主目录。 • 在 Linux 上使用带多个 GPU 的 NVENC 硬件编码器时 , 遵循 nvidia-smi 工具使用的相同顺序。 • Linux 客户端现在接收和处理来自 Windows DCV 打印机的打印文件。

DCV 2017.3-6698 - 2019 年 2 月 24 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> • nice-dcv-server: 6698 • nice-dcv-client: 5946 • nice-dcv-viewer (Linux) : 683 • nice-xdcv : 207 • nice-dcv-gl: 471 • nice-dcv-gltest: 210 • nice-dcv-simple-external-身份验证器 : 66 	<ul style="list-style-type: none"> • 增加了对 Kerberos (GSSAPI) 身份验证的支持。 • 增加了对支持触摸的 Windows 版本上的触摸事件的支持。 • 在使用系统身份验证 (Windows 凭证提供程序) 时自动解锁 Windows 会话。 	<ul style="list-style-type: none"> • 增加了选择使用 Y'UV444 编码的选项。 • EL6 RPM 现在包括 NVENC 编码器模块。 • Windows 系统身份验证现在接受 name@domain 格式。 • Yubikey USB 设备现已添加到允许列表中。 • 改进了日语键盘支持。 • 输入授权权限更为精细。增加了操作虚拟光标的 pointer 权限。相对鼠标模式取决于鼠标 (针对运动注入) 和指针 (针对运动反馈)。增加了 keyboard-sas 权限以在 Windows 上操作 SAS (Ctrl+Alt+Del)。keyboardsas 依赖于 keyboard 权限。 • 修复了支持异步剪贴板 API 的浏览器上 Web 客户端中的空剪贴板事件的问题。 • 修复了导致客户端无法接收到第一帧的捕获模块上的竞用问题。 • 改进了对并发文件存储传输的处理。 • 使用较新的 NVIDIA 驱动程序在 Windows 上修复了

内部版本号	新功能	更改和错误修复
		<p>NvIFR。新的驱动程序改变了行为。现在将自动检测驱动程序版本，并相应地执行内存处理。</p> <ul style="list-style-type: none"> 从不停止尝试重新获取 RLM 许可证令牌。这使您能够从 <code>licensing error</code> 状态恢复，甚至在延长时段后也是如此。 添加了在 Windows 客户端中设置全屏键盘快捷键的选项。 在 Windows 客户端中改进了在多个显示器之间拖动窗口时的自动调整逻辑。 在 Windows 客户端中修复了 UI 不触发断开连接时的 <code>prompt-reconnect</code> 选项。 修复了 DCV-GL 与 NVIDIA 驱动程序 410.xx 不兼容的问题。 使用 Matlab 和 Blender 应用程序修复了 DCV-GL 中的性能下降问题。

DCV 2017.2-6182 - 2018 年 10 月 8 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 6182 nice-dcv-client: 5890 	<ul style="list-style-type: none"> 增加了对 Linux 虚拟会话的音频播放支持。 改进了智能卡性能。 	<ul style="list-style-type: none"> 与键盘操作相关的改进和错误修复。

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> • nice-dcv-viewer (Linux) : 503 • nice-xdcv : 180 • nice-dcv-gl: 427 • nice-dcv-gltest : 201 • nice-dcv-simple-external-身份验证器 : 58 	<ul style="list-style-type: none"> • 增加了 Linux 客户端的文件传输支持。 	<ul style="list-style-type: none"> • 更改配置中的日志级别不再需要重新启动服务器。 • Windows Server 安装程序现在跳过安装 Microsoft C 运行时可再发行组件 (如果已安装)。 • 在 EC2 上运行时，如果访问许可证的 S3 失败，则用户界面中将显示一条通知。 • Linux dcv 命令行工具现在支持 list-connections 和 describe-session 子命令，并包含一个用于发出 JSON 输出的选项。 • 在 display 部分中添加了 cuda-devices 设置。该设置将服务器配置为在不同的 CUDA 设备上分发 NVENC 编码。 • 改进了处理多个并发命令时的会话创建代码的可靠性。 • 将默认剪贴板限制增加到了 20 MB。 • Windows 客户端现在检测旧 .dcv 文件并启动 DCV 2016 Endstation (如果已安装)。 • DCV 简单外部身份验证器现在始终使用系统 Python 解释器，而不是环境中设置的解释器。 • 改进了 DCV-GL 的回读策略，以提高性能和可靠性。

内部版本号	新功能	更改和错误修复
		<ul style="list-style-type: none"> DCV-GL 现在检查窗口大小是否在前缓冲区回读后发生了更改。这修复了 Coot 应用程序的渲染问题。

DCV 2017.1-5870 - 2018 年 8 月 6 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 5870 nice-dcv-client: 5813 nice-dcv-viewer (Linux) : 450 nice-xdcv : 170 nice-dcv-gl: 366 nice-dcv-gltest: 198 nice-dcv-simple-external-身份验证器 : 53 	<p>发布了 Ubuntu 18.04 程序包。在控制台模式中工作时，必须将系统配置为使用 LightDM 或您选择的其他显示管理器，因为 GDM 不会公开所需的 X11 显示信息。虚拟会话不受此限制的影响。</p>	<ul style="list-style-type: none"> 现在，在创建会话时，将会读取许可证设置。这样，管理员无需重新启动服务器即可更改该设置。 解决了 Windows 客户端存在的导致程序在某些系统上意外退出的稳定性问题。 减少了可能出现的错误情况下的日志记录。

DCV 2017.1-5777 - 2018 年 6 月 29 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 5777 nice-dcv-client: 5777 nice-dcv-viewer (Linux) : 438 nice-xdcv : 166 nice-dcv-gl: 366 nice-dcv-gltest: 189 	<ul style="list-style-type: none"> 增加了 Linux 本机客户端。 增加了对 3DConnexion 鼠标和 USB 存储设备的支持。 Windows 会话在最后一个客户端断开连接时自动锁定。 	<ul style="list-style-type: none"> Linux 版本中的性能改进。 已将 NVIDIA 设备上的默认硬件编码器更改为 NVENC，以避免新 NVIDIA 驱动程序中的 NvIFR 出现问题。 改进了 Linux 上的智能卡支持。

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-simple-external-身份验证器 : 51 		<ul style="list-style-type: none"> 修复了使用 Linux 控制台会话时上传文件的文件权限。

DCV 2017.0-5600 - 2018 年 6 月 4 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 5600 nice-dcv-client: 5600 nice-xdcv : 160 nice-dcv-gl: 279 nice-dcv-gltest: 184 nice-dcv-simple-external-身份验证器 : 48 	<ul style="list-style-type: none"> 增加了对 Linux 上的多个显示器的支持。 Windows 客户端性能改进。 在 Chrome 66+ 上使用了新的剪贴板 API。 增加了适用于 Windows 的 NVENC 编码器。 	<ul style="list-style-type: none"> EC2 上的使用现在需要能够从运行 DCV 服务器的实例访问 S3。 针对服务器帧处理和 Windows 客户端解码的性能改进。 修复了与修饰符相关的键盘问题 NumPad 和被屏蔽的修饰符。 防止在 Linux 上使用外部身份验证器时发生文件描述符泄露。 修复了可能出现的智能卡连接错误。

DCV 2017.0-5121 - 2018 年 3 月 18 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 5121 nice-dcv-client: 5121 nice-xdcv : 146 nice-dcv-gl: 270 nice-dcv-gltest: 184 	<ul style="list-style-type: none"> Windows 本机客户端现在可以识别 DPI。 增加了对相对鼠标移动模式的支持。 	<ul style="list-style-type: none"> 防止 Linux 上的 Ansys cfx5solve 挂起。 修复了 Windows 10 上可能出现的代理挂起情况。 改进了 Web Client 用户界面。

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-simple-external-身份验证器 : 46 		<ul style="list-style-type: none"> 指定域时规范化的 Windows 用户名。 修复了 RHEL6 上的外部验证器。

DCV 2017.0-4334 - 2018 年 1 月 24 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> nice-dcv-server: 4334 nice-dcv-client: 4334 nice-xdcv : 137 nice-dcv-gl: 254 nice-dcv-gltest: 184 nice-dcv-simple-external-身份验证器 : 45 	<ul style="list-style-type: none"> 改进了键盘操作。 修复了 RHEL6 上的 DBus 问题，即，关闭会话不允许创建新的会话。 改进了对本机客户端上的 SOCKS5 代理的支持。 解决了一个错误，该错误导致在虚拟会话上运行时 Headwave 崩溃，以及导致在虚拟会话上运行时 Chimera 崩溃。 改进了对虚拟会话的字体支持。

DCV 2017.0-4100 - 2017 年 12 月 18 日

内部版本号
<ul style="list-style-type: none"> nice-dcv-server: 4100 nice-dcv-client: 4100 nice-xdcv : 118 nice-dcv-gl: 229 nice-dcv-gltest: 158 nice-dcv-simple-external-身份验证器 : 35

文档历史记录

下表介绍了该版本的 NICE DCV 的文档。

更改	描述	日期
NICE DCV 版本 2023.1	NICE DCV 修复了 Windows 客户端 2023.1。有关更多信息，请参阅 DCV 2023.1-16388 — 2024 年 3 月 5 日 。	2024年3月5日
NICE DCV 版本 2023.1	NICE DCV 修复至 2023.1。有关更多信息，请参阅 DCV 2023.1-16388 – 2023 年 12 月 19 日 。	2023 年 12 月 19 日
NICE DCV 版本 2023.1	NICE DCV 2023.1 现已发布。有关更多信息，请参阅 DCV 2023.1-16220 - 2023 年 11 月 9 日 。	2023 年 11 月 9 日
NICE DCV 版本 2023.0	NICE DCV 不再支持生命周期结束的操作系统。	2023 年 6 月 30 日
NICE DCV 版本 2023.0	NICE DCV 2023.0 修复。有关更多信息，请参阅 DCV 2023.0-15487 - 2023 年 6 月 29 日 。	2023 年 6 月 29 日
NICE DCV 版本 2023.0	NICE DCV 2023.0 修复。有关更多信息，请参阅 DCV 2023.0-15065 - 2023 年 5 月 3 日 。	2023 年 5 月 3 日
NICE DCV 版本 2023.0	NICE DCV 2023.0 更新和修复。有关更多信息，请参阅 DCV 2023.0-15022 - 2023 年 4 月 21 日 。	2023 年 4 月 21 日

更改	描述	日期
NICE DCV 版本 2023.0	NICE DCV 2023.0 现已发布。 有关更多信息，请参阅 DCV 2023.0-14852 - 2023 年 3 月 28 日 。	2023 年 3 月 28 日
NICE DCV 版本 2022.2	NICE DCV 2022.2 现已发布。 有关更多信息，请参阅 DCV 2022.2-13907 - 2022 年 11 月 11 日 。	2022 年 11 月 11 日
NICE DCV 版本 2022.1	NICE DCV 2022.1 现已发布。 有关更多信息，请参阅 DCV 2022.1-13067 - 2022 年 6 月 29 日 。	2022 年 6 月 29 日
NICE DCV 版本 2022.0	NICE DCV 2022.0 现已发布。 有关更多信息，请参阅 DCV 2022.0-11954 - 2022 年 2 月 23 日 。	2022 年 2 月 23 日
NICE DCV 版本 2021.3	NICE DCV 2021.3 现已发布。 有关更多信息，请参阅 DCV 2021.3-11591 - 2021 年 12 月 20 日 。	2021 年 12 月 20 日
NICE DCV 版本 2021.2	NICE DCV 2021.2 现已发布。 有关更多信息，请参阅 DCV 2021.2-11048 - 2021 年 9 月 1 日 。	2021 年 9 月 1 日
NICE DCV 版本 2021.1	NICE DCV 2021.1 现已发布。 有关更多信息，请参阅 DCV 2021.1-10557 - 2021 年 5 月 31 日 。	2021 年 5 月 31 日

更改	描述	日期
NICE DCV 版本 2021.0	NICE DCV 2021.0 现已发布。有关更多信息，请参阅 DCV 2021.0-10242 - 2021 年 4 月 12 日 。	2021 年 4 月 12 日
NICE DCV Web Client SDK	NICE DCV Web Client SDK 现已发布。NICE DCV Web Client SDK 是一个 JavaScript 库，你可以用它来开发自己的 NICE DCV 网络浏览器客户端应用程序，最终用户可以使用这些应用程序来连接正在运行的 NICE DCV 会话并与其交互。有关更多信息，请参阅 《NICE DCV Web Client SDK 开发人员指南》 。	2021 年 3 月 24 日
NICE DCV 版本 2020.2	NICE DCV 2020.2 现已发布。有关更多信息，请参阅 DCV 2020.2-9508 - 2020 年 11 月 11 日 。	2020 年 11 月 11 日
NICE DCV 版本 2020.1	NICE DCV 2020.1 现已发布。有关更多信息，请参阅 DCV 2020.1-8942 - 2020 年 8 月 3 日 。	2020 年 8 月 3 日
NICE DCV 版本 2020.0	NICE DCV 2020.0 包括对环绕声 7.1、触摸和触控笔以及使用新 Microsoft Edge 浏览器的多显示器的支持。有关更多信息，请参阅《NICE DCV 管理员指南》中的 安装 NICE DCV 服务器 。	2020 年 4 月 16 日

更改	描述	日期
HTTP 响应标头	可以配置 NICE DCV 服务器以发送额外的 HTTP 响应标头。	2019 年 8 月 26 日
macOS 客户端	NICE DCV 现在提供 macOS 客户端。有关更多信息，请参阅《NICE DCV 用户指南》中的 macOS 客户端 。	2019 年 4 月 18 日
智能卡缓存	NICE DCV 服务器现在可以缓存从客户端接收的智能卡数据，以帮助提高性能。有关更多信息，请参阅《NICE DCV 管理员指南》中的 配置智能卡缓存 。	2018 年 10 月 8 日
Linux 客户端	NICE DCV 提供适用于 RHEL 7、CentOS 7、SLES 12 和 Ubuntu 16.04/18.04 的 Linux 客户端。有关更多信息，请参阅《NICE DCV 用户指南》中的 Linux 客户端 。	2018 年 8 月 29 日
更新了参数参考	更新了参数参考。有关更多信息，请参阅《NICE DCV 管理员指南》中的 NICE DCV 服务器参数参考 。	2018 年 8 月 7 日
USB 远程控制	NICE DCV 允许客户使用专用 USB 设备，例如 3D 指点设备或绘图板。有关更多信息，请参阅《NICE DCV 管理员指南》中的 启用 USB 远程控制 。	2018 年 8 月 7 日
NICE DCV 的初始版本。	该内容的第一版。	2018 年 6 月 5 日

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。