

---

# NICE DCV

## 管理员指南

**亚马逊云科技**



## NICE DCV: 管理员指南

Copyright © 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 拥有的其它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。

Amazon Web Services 文档中描述的 Amazon Web Services 服务或功能可能因区域而异。要查看适用于中国区域的差异，请参阅[中国的 Amazon Web Services 服务入门](#)。

## Table of Contents

什么是好的 DCV ? .....	1
NICE DCV 的工作原理 .....	1
功能 .....	1
定价 .....	2
NICE DCV 服务器 .....	3
要求 .....	3
支持的功能 .....	4
设置 .....	6
安装 .....	6
Windows .....	6
Linux .....	10
许可 .....	33
NICE DCV 许可要求 .....	33
安装延期评估许可证 .....	35
安装生产许可证 .....	36
更新生产许可证 .....	43
升级 .....	43
兼容性注意事项 .....	43
在 Windows 上升级 NICE DCV 服务器 .....	43
在 Linux 上升级 NICE DCV 服务器 .....	44
卸载 .....	44
在 Windows 上卸载 NICE DCV 服务器 .....	44
在 Linux 上卸载 NICE DCV 服务器 .....	45
管理服务器 .....	46
启动服务器 .....	46
在 Windows 上启动服务器 .....	46
在 Linux 上启动服务器 .....	47
停止服务器 .....	47
在 Windows 上停止服务器 .....	47
在 Linux 上停止服务器 .....	48
启用 QUIC UDP .....	48
更改 TCP/UDP 端口和地址 .....	49
更改服务器的 TCP/UDP 端口 .....	50
监听特定的端点 .....	51
管理 TLS 证书 .....	52
断开空闲客户端连接 .....	53
在 Windows 上更改空闲超时期限 .....	54
在 Linux 上更改空闲超时期限 .....	54
在 Linux 上启用 GPU .....	55
启用触摸屏和触控笔支持 .....	56
配置手写笔 .....	57
启用游戏手柄支持 .....	58
支持 Xbox 360 控制器 .....	58
启用 USB 远程控制 .....	58
在 Windows 上将设备添加到允许列表中 .....	59
在 Linux 上将设备添加到允许列表 .....	59
配置智能卡缓存 .....	60
启用会话存储 .....	60
在 Windows 上启用会话存储 .....	61
在 Linux 上启用会话存储 .....	61
在 Linux 上配置打印机 .....	62
排查打印机问题 .....	63
在 Linux 上配置剪贴板 .....	63
将客户端剪贴板内容粘贴到主选区 .....	64

将主要选择内容复制到客户端剪贴板 .....	64
配置多通道音频 .....	64
在 Windows NICE DCV 服务器上配置音频通道 .....	65
在 Linux NICE DCV 服务器上配置音频通道 .....	66
配置 HTTP 标头 .....	67
在 Windows NICE DCV 服务器上配置 HTTP 标头 .....	67
在 Linux NICE DCV 服务器上配置 HTTP 标头 .....	67
配置身份验证 .....	68
在 Windows 上配置身份验证 .....	68
在 Lux 上配置身份验证 .....	69
配置授权 .....	70
默认权限文件 .....	70
自定义权限文件 .....	70
使用权限文件 .....	70
管理会话 .....	75
NICE DCV 会议简介 .....	75
控制台会话 .....	75
虚拟会话 .....	76
使用命令行工具管理会话 .....	76
在 Windows NICE DCV 服务器上使用命令行工具 .....	76
在 Linux NICE DCV 服务器上使用命令行工具 .....	77
启动会话 .....	77
手动启动控制台和虚拟会话 .....	77
启用自动控制台会话 .....	80
停止会话 .....	81
语法 .....	82
示例 .....	82
管理运行会话 .....	82
管理会话存储 .....	82
管理会话授权 .....	83
管理会话显示布局 .....	84
管理会话名称 .....	86
查看会话 .....	87
列出所有活动会话 .....	87
查看有关特定会话的信息 .....	87
获取屏幕截图 .....	89
语法 .....	89
选项 .....	89
示例 .....	90
如何 .....	91
使用外部验证 .....	91
配置 NICE DCV 服务器 .....	91
使用令牌 .....	92
服务要求 .....	92
查找并停止空闲会话 .....	93
启用到 X 服务器的远程 X 连接 .....	94
启用到 X 服务器的远程 X 连接 .....	94
在 iFrame 中嵌入 NICE DCV Web 浏览器客户端 .....	95
问题排查 .....	97
使用日志文件 .....	97
在 Windows 上更改日志文件详细程度 .....	97
在 Linux 上更改日志文件详细程度 .....	98
在 Linux 上创建虚拟会话故障排除 .....	98
调查 Linux 上的虚拟会话创建失败 .....	98
在 Linux 上创建故障安全虚拟会话 .....	99
UID 更改后 Linux 会话无法启动 .....	100
修复 Windows 上的光标问题 .....	100

修复复制并粘贴到 IntelliJ IDEA .....	100
参数参考 .....	101
audio 参数 .....	101
clipboard 参数 .....	102
connectivity 参数 .....	103
display 参数 .....	105
display/linux 参数 .....	108
input 参数 .....	109
license 参数 .....	109
log 参数 .....	110
security 参数 .....	111
session-management 参数 .....	115
session-management/automatic-console-session 参数 .....	117
session-management/defaults 参数 .....	118
smartcard 参数 .....	118
webcam 参数 .....	118
windows 参数 .....	119
修改配置参数 .....	119
Windows NICE DCV 服务器服务器服务器 .....	120
Linux NICE DCV 服务器 .....	120
NICE DCV 终止支持寿命 .....	122
EOSL 时间线 .....	122
面向客户的 EOSL 路径 .....	122
EOSL 常见问题 .....	123
发行说明和文档历史记录 .....	124
发行说明 .....	124
NICE DCV 2022.1-13300 .....	125
NICE DCV 2022.1-13216 .....	125
NICE DCV 2022.1-13067 .....	125
NICE DCV 2022.0-12760 .....	126
NICE DCV 2022.0-12627 .....	126
NICE DCV 2022.0-12123 .....	127
NICE DCV 2022.0-11954 .....	127
NICE DCV 2021.3-11591 .....	128
NICE DCV 2021.2-11445 .....	128
NICE DCV 2021.2-11190 .....	129
NICE DCV 2021.2-11135 .....	129
NICE DCV 2021.2-11048 .....	129
DCV 2021.1-10888858585 .....	130
DCV 2021.1-10555959999 .....	131
DCV 2021.1-10555555555 .....	131
DCV 2021.0-102424242424 .....	131
DCV 2020.2-966666666 .....	132
DCV 2020.2-950508-9505088 .....	132
DCV 2020.1-90121212-9012 .....	133
DCV 2020.1-90121212-9012 .....	133
DCV 2020.1-894444444 .....	133
DCV 2020.0-8428 .....	134
DCV 2019.1-7644 .....	135
DCV 2019.1-7423 .....	135
DCV 2019.0-7318 .....	135
DCV 2017.4-6898 .....	136
DCV 2017.3-6698 .....	137
DCV 2017.2-6182 .....	138
DCV 2017.1-5870 .....	139
DCV 2017.1-5777 .....	139
DCV 2017.0-5600 .....	139

DCV 2017.0-5121 .....	140
DCV 2017.0-4334 .....	140
DCV 2017.0-4100 .....	140
文档历史记录 .....	141
.....	cxliii

# 什么是好的 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 结合使用，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 的工作原理 \(p. 1\)](#)
- [NICE DCV 的功能 \(p. 1\)](#)
- [NICE DCV 定价 \(p. 2\)](#)

## NICE DCV 的工作原理

要使用 NICE DCV 服务器软件，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。您的用户通过使用 NICE DCV 客户端应用程序远程连接到会话来访问应用程序。建立连接后，NICE DCV 服务器软件会压缩应用程序的可视输出，并以加密的像素流形式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 的功能

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，而不传输几何体和场景信息。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，以匹配客户端窗口的大小。
- 支持多屏幕— 可让您在最多四个显示器上扩展会话桌面。
- 调整压缩级别— 根据网络的可用带宽和延迟自动适应视频压缩级别。
- 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。
- 每个服务器支持多个会话（仅限 Linux NICE DCV 服务器）— 支持每个 Linux NICE DCV 服务器拥有多个虚拟会话，以最大限度地节省成本。
- 支持 GPU 共享（仅限 Linux NICE DCV 服务器）— 可让您在 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。
- 支持触控输入、手写笔输入和游戏手柄— 允许您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 USB 和智能卡以及手写笔远程处理— 可让您在 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— 可让您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— 可让您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，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费用。您需要为您使用的实例和其他 Amazon EC2 功能支付标准费用。

需要许可证才能在本地服务器或其他基于云的服务器上安装 NICE DCV 服务器。有关更多信息，请参阅[授予 NICE DCV 服务器的许可](#) (p. 33)。

# NICE DCV 服务器

NICE DCV 服务器可用于 Windows 和 Linux。这两个服务器提供了类似的功能，但有一些差异。选择最适合您的需要的 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 \(p. 3\)](#)
- [支持的功能 \(p. 4\)](#)

## 要求

为获得良好的 NICE DCV ( NICE DCV )，请确保服务器满足以下最低要求。请记住，您的用户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。

如果您正在 Amazon EC2 实例上安装 NICE DCV 服务器上，我们建议您使用 Amazon EC2 G3、G4dn 或 G4ad 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](#) 和 [Amazon EC2 G4 实例](#)。您可以在任何其他实例类型上安装 NICE DCV 服务器可能存在屏幕分辨率限制。要绕过此限制，请下载并安装 [适用于 EC2 的 NICE DCV 虚拟显示驱动程序](#)。

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"> <li>• Windows 8.1</li> <li>• Windows Server 2012 R2</li> <li>• Windows 10</li> <li>• Windows 11</li> <li>• Windows Server 2016</li> <li>• Windows Server 2019</li> <li>• Windows Server 2022</li> </ul> <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"> <li>• Amazon Linux 2</li> <li>• CentOS 7.6 或更高版本</li> <li>• CentOS 8.x 或 CentOS Stream 8</li> <li>• RHEL 7.6 或更高版本</li> <li>• RHEL 8.x</li> <li>• 使用 SP3 或更高版本</li> <li>• SUSE Linux Enterp</li> <li>• Ubuntu 18.04</li> <li>• Ubuntu 20.04</li> <li>• Ubuntu 22.04</li> <li>• 洛基 Linux 8.5 及以上版本</li> </ul>
支持架构	64 位 x86	<ul style="list-style-type: none"> <li>• 64 位 x86</li> <li>• 64 位 ARM ( 仅支持运行亚马逊 Linux 2、CentOS 7.x/8.x、RHEL 7.x/8.x 和 Ubuntu 18.04 的 Amazon EC2 实例 )</li> </ul>
GPU	<p>( 可选 ) 对于基于硬件的视频编码，需要一个 NVidia 或 AMD GPU。如果您的服务器没有 GPU，则使用基于软件的视频编码。</p> <p>Note</p> <ul style="list-style-type: none"> <li>• NVIDIA GPU 需要使用 NVENC 进行基于硬件的视频编码。在 Ubuntu 20 上，一款计算能力 &gt;= 的 NVIDIA GPU 3.5 是必需的。</li> </ul>	

	Windows 服务器	Linux 服务器
	<ul style="list-style-type: none"> <li>AMD GPU 需要适用于 Linux 或 Windows 的高级媒体框架 (AMF)，或者仅适用于 Windows 的 Rapidfire 才能进行基于硬件的视频编码。对于 Linux，可以通过安装附加软件包在 Ubuntu 实例上使用 AMF 编码器 <code>amf-amdgpu-pro</code> 由 AMD 驱动程序提供。</li> </ul>	<p>对于跨虚拟会话的 GPU 共享，需要一个 NVIDIA GPU。</p> <p><b>Note</b></p> <p>对于使用 AMD GPU 的 Linux 服务器上仅支持控制台会话。</p>
Network	默认情况下，NICE DCV 服务器通过端口 843 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。	

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

## 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	NICE DCV (p. 6)	Linux NICE DCV 服务器 (p. 10)
控制台会话 (p. 75)	✓	✓
虚拟会话 (p. 75)	✗	✓
QUIC (UDP) 传输协议 (p. 48)	✓	✓
可配置的 TCP/UDP 端口和地址 (p. 49)	✓	✓
自定义 TLS 证书 (p. 52)	✓	✓
空闲客户端断开连接 (p. 53)	✓	✓
GPU 共享 (p. 55)	✗	✓
USB 远程控制 (p. 58)	✓	✓
智能卡支持 (p. 60)	✓	✓
网络摄像头支持	✓ ( Windows 10 和 Windows Server 2016 及更高版本 )	✗
会话存储和文件传输 (p. 60)	✓	✓
复制和粘贴 (p. 63)	✓	✓
自定义 HTTP 标头 (p. 67)	✓	✓
从会话打印 (p. 62)	✓	✓
立体声 2.0 音频播放 (p. 64)	✓	✓

NICE DCV 管理员指南  
支持的功能

---

功能	NICE DCV (p. 6)	Linux NICE DCV 服务器 (p. 10)
<a href="#">环绕立体声音频播放 (p. 64)</a>	✓ ( 高达 7.1 )	✓ ( 最高 5.1 )
<a href="#">立体声 2.0 音频录制 (p. 64)</a>	✓	✓
<a href="#">触摸屏支持 (p. 56)</a>	✓ ( Windows 8.1 和 Windows Server 2012 及更高版本 )	✓
<a href="#">触控笔支持 (p. 56)</a>	✓ ( Windows 10 和 Windows Server 2019 )	✓
<a href="#">游戏手柄支持 (p. 58)</a>	✓ ( Windows 10 和 Windows Server 2016 及更高版本 )	x

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

# 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。确保软件得到正确许可。

以下主题介绍如何安装和许可 NICE DCV 服务器。这些区域有：[许可 \(p. 33\)](#)主题仅适用于在本地和其他基于云的服务器上安装。这是因为在 Amazon EC2 实例上使用 NICE DCV 服务器不需要许可证。

## 主题

- [安装 NICE DCV 服务器 \(p. 6\)](#)
- [授予 NICE DCV 服务器的许可 \(p. 33\)](#)
- [升级 NICE DCV 服务器 \(p. 43\)](#)
- [卸载 NICE DCV 服务器 \(p. 44\)](#)

## 安装 NICE DCV 服务器

以下主题介绍如何在 Windows 和 Linux 上安装最新版本的 NICE DCV 服务器。如果您正在 Amazon EC2 实例或其他本地服务器或基于云的服务器上安装 NICE DCV，请执行这些步骤。

### 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 \(p. 43\)](#)。

## 主题

- [在 Windows 上安装 NICE DCV 服务器 \(p. 6\)](#)
- [在 Linux 上安装 NICE DCV 服务器 \(p. 10\)](#)

## 在 Windows 上安装 NICE DCV 服务器

本节介绍如何在 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 \(p. 6\)](#)
- [在 Windows 上安装 NICE DCV 服务器 \(p. 8\)](#)

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

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

## 主题

- [加速计算实例的先决条件 \(p. 6\)](#)
- [其他实例系列的先决条件 \(p. 7\)](#)

### 加速计算实例的先决条件

#### GPU 图形实例的先决条件

如果您使用的是 GPU 图形实例（例如 G2、G3、G4dn 或 G4ad 实例），我们建议您安装和配置相应的 NVIDIA 或 AMD GPU 驱动程序。GPU 驱动程序允许：

- 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
- 提高了服务器监视器的最大分辨率—高达 4096x2160 x 2160
- 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](#)。

- 有关使用 NVIDIA GPU ( 例如 G2、G3 或 G4dn 实例 ) 的实例，请参阅在 [Windows 上安装 NVIDIA 驱动程序](#)。
- 有关使用 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[深入了解新的 Amazon EC2 G4ad 实例](#)博客package。

### 其他加速计算实例的先决条件

如果您使用的加速计算实例不是 GPU 图形实例 ( 例如 P2、P3 或 P3dn 实例 ) ，我们建议您安装和配置相应的 NVIDIA GPU 驱动程序。NVIDIA GPU 驱动程序支持 H.264 视频流编码的硬件加速。

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[公有 NVIDIA 驱动程序](#)中的适用于 Windows 实例[Amazon EC2 用户指南](#)。

在加速计算实例上安装 NVIDIA GPU 驱动程序不会提高服务器监视器限制或分辨率。要添加其他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 vGPU 软件](#)在 NVIDIA 网站上。

### 其他实例系列的先决条件

对于加速计算实例以外的实例，我们建议您安装 NICE DCV 虚拟显示驱动程序。这包括通用实例、计算优化实例、内存优化实例和存储优化实例系列中的实例。

安装 NICE DCV 虚拟显示器驱动程序将启用以下功能：

- 最多Support 四台显示器
- Support 自定义分辨率
- Support 4K 超高清分辨率

您无法使用 Windows 控制面板管理 NICE DCV 服务器连接的服务器监视器。

#### Note

Windows Server 2016 及更高版本支持 NICE DCV 虚拟显示器驱动程序。

#### 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 ) 一起安装可能会导致发生冲突。为了避免发生冲突，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. 从以下位置下载 NICE DCV 虚拟显示驱动程序安装程序[NICE DCV 网站](#)。
2. 要通过运行向导来安装驱动程序，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-38.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

可以使用安装向导在 Windows 主机服务器上安装 NICE DCV 服务器。此向导将指导您完成一系列步骤，说明如何自定义 NICE DCV 服务器安装。此外，您也可以使用命令行来执行自动安装。这将使用默认设置来自动执行安装过程。

### 目录

- [使用向导 \(p. 8\)](#)
- [自动安装 \(p. 8\)](#)

### 使用向导

使用 NICE DCV 服务器安装向导执行指导式安装。

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. 从以下位置下载 NICE DCV 服务器安装程序[NICE DCV](#)网站。

#### Note

NICE DCV 服务器仅提供 64 位版本，并且仅在 64 位 Windows 操作系统上受支持。

#### Tip

这些区域有：[最新软件包](#)下载网站的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-2022.1-version_number.msi`。
4. 在欢迎屏幕上，选择 Next。
5. 在“End-User License Agreement (最终用户许可协议)”屏幕上，阅读许可协议。如果您接受条款，请选择我接受许可协议中的条款复选框，然后选择下一步。
6. (可选) 通过选择组件选择屏幕包。要标记要安装的组件，请选择该项并选择将安装在本地硬盘驱动器上。要从安装中省略某个组件，请选择该项目，然后选择整个功能将不可用。
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) 要在安装之后手动启动 NICE DCV 服务器，请选择不，我想手动启动 DCV 服务。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 Next (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。

#### 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。

### 自动安装

默认情况下，自动安装执行以下操作：

- 添加防火墙规则以允许通过端口 8443 进行通信。

- 允许 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`— 阻止安装程序添加防火墙规则。
- `DISABLE_SERVER_AUTOSTART=1`— 阻止 NICE DCV 服务器在安装之后自动启动 NICE DCV 服务器。
- `DISABLE_AUTOMATIC_SESSION_CREATION=1`— 阻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其他所有者。
- `ADDLOCAL=component_list`— 将元素添加到要安装的元素集中。
- `REMOVE=component_list`— 从要安装的元素集中移除元素。

#### Note

这些区域有：`REMOVE`选项在`ADDLOCAL`选项。两个列表中的元素未安装。

`component_list` 是一个逗号分隔的列表，可以包含以下值：

- `audioMicDriver`: 麦克风 Windows 驱动程序
- `audioSpkDriver`: 扬声器驱动程序
- `printerDriver`: 打印机驱动程序
- `usbDriver`: USB 设备远程化驱动程序（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
- `ALL`: 所有组件

### 使用自动安装功能在 Windows 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. 从以下位置下载 NICE DCV 服务器安装程序[NICE网站](#)。

#### Note

NICE DCV 服务器仅提供 64 位版本，并且仅在 64 位 Windows 操作系统上受支持。

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-2022.1-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-2022.1-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-2022.1-version_number.msi
```

```
ADDDLOCAL=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 \(p. 10\)](#)
- [在 Linux 上安装 NICE DCV 服务器 \(p. 19\)](#)
- [安装后检查 \(p. 31\)](#)

## Linux NICE DCV 服务器的先决条件

NICE DCV 使客户端能够访问 Linux 服务器上的远程图形 X 会话。这提供了对相应的 Linux 桌面的访问权限。NICE DCV 支持两种类型的 Linux 桌面流管理：控制台会话和虚拟会话。有关控制台和虚拟会话的更多信息，请参阅[管理 NICE DCV 会话 \(p. 75\)](#)。

本主题介绍如何准备好必需的先决条件以在 Linux 服务器上使用 NICE DCV。

### 目录

- [安装桌面环境和桌面管理器 \(p. 10\)](#)
- [禁用 Wayland 协议 \(仅限 GDM3\) \(p. 14\)](#)
- [配置 X 服务器 \(p. 15\)](#)
- [安装 glxinfo 实用程序 \(p. 15\)](#)
- [验证 OpenGL 软件渲染 \(p. 16\)](#)
- [为图形实例安装 GPU 驱动程序 \(p. 16\)](#)
- [为非 GPU 实例安装 xDummy 驱动程序 \(p. 18\)](#)

## 安装桌面环境和桌面管理器

安装一个桌面环境和桌面管理器，以提高您在 Linux 服务器上使用 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。桌面环境有很多，NICE DCV 可与其中的大部分协作。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，以及如何在支持的操作系统上配置和启动 X 服务器。

### RHEL 7.x/8.x, CentOS 7.x/8.x, and Rocky Linux

RHEL 7.x/8.x、CentOS 7.x/8.x 和 Rocky Linux 的默认桌面环境是 Gnome3，默认桌面管理器是 GDM。

在 RHEL 7.x/8.x、CentOS 7.x/8.x 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 7.x/8.x、CentOS 8.x 和 Rocky Linux 8.5 及更高版本

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

- CentOS 7.x

```
$ 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

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 18.x

对于 Ubuntu 18.x，默认桌面环境是 Gnome3，默认桌面管理器是 GDM3。在 Ubuntu 18.x 中，NICE DCV 控制台会话目前不支持 GDM3。因此，如果您打算使用 NICE DCV 控制台会话，则建议您采用 LightDM 桌面管理器。

在 Ubuntu 18.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 LightDM。

```
$ sudo apt install lightdm
```

2. 更新软件包来确保 Linux 服务器是最新的。

```
$ sudo apt 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。根据运行的会话类型，可能需要对系统进行不同的配置。

- 控制台会话

Ubuntu 20.x 上的 NICE DCV 主机会话目前不支持 LightDM。如果您打算使用 NICE DCV 控制台会话，则建议您采用 GDM3 桌面管理器。

- 虚拟会话

由于一个已知的 [GDM 问题](#)，虚拟会话无法与 Ubuntu 20.x 上的 GDM3 配合使用。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，则可以禁用桌面管理器，因为它不是运行虚拟会话所必需的。通过在创建虚拟会话之前运行以下命令，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外，还需要在系统上启动 X 服务器，然后才能创建虚拟会话。为此，请运行以下命令：

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

```
sudo dcvstartx &
```

## 在 Ubuntu 20.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 (仅适用于控制台会话)

```
$ sudo apt install gdm3
```

2. 如果使用 GDM3，请确认 GDM3 已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
```

如果未将 GDM3 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
```

### 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
```

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

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

2. 更新软件包来确保 Linux 服务器是最新的。

```
$ sudo zypper update
```

3. 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。根据您运行的会话类型，您可能需要对系统进行不同的配置。

- 控制台会话

SUSE Linux Enterprise 15.x 上的 NICE DCV 控制台会话目前不支持 LightDM。如果您打算使用 NICE DCV 控制台会话，则建议您采用 GDM3 桌面管理器。

- 虚拟会话

由于一个已知的 [GDM 问题](#)，虚拟会话无法在 SUSE Linux Enterprise 15.x 上运行。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，则可以禁用桌面管理器，因为它不是运行虚拟会话所必需的。通过在创建虚拟会话之前运行以下命令，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外，还需要在系统上启动 X 服务器，然后才能创建虚拟会话。为此，请运行以下命令：

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

```
sudo dcvstartx &
```

## 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
```

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

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

### 2. 更新软件包来确保 Linux 服务器是最新的。

```
$ sudo zypper update
```

### 3. 重新启动 Linux 服务器。

```
$ sudo reboot
```

## 禁用 Wayland 协议 ( 仅限 GDM3 )

NICE DCV 不支持 Wayland 协议。如果您使用的是 GDM3 桌面管理器，则必须禁用 Wayland 协议。如果您不使用 GDM3，请跳过此步骤。

### 禁用 Wayland 协议

#### 1. 使用首选文本编辑器打开以下文件。

- RHEL 8.x、CentOS 8.x、CentOS 8.x、CentOS 8.x、SUSE 8.x、

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

- Ubuntu 2.x

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

#### 2. 在 [daemon]setseWaylandEnable 到 false.

```
[daemon]
```

```
WaylandEnable=false
```

3. 重新启动 GDM 服务。

- RHEL 8.x 和 CentOS 8.x

```
$ sudo systemctl restart gdm
```

- Ubuntu 2.x

```
$ sudo systemctl restart gdm3
```

- SUSE Linux Enterpris

```
$ 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

如果您打算使用不带 GPU 共享的虚拟会话，则不需要 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 7.x/8.x、CentOS 7.x/8.x、Rocky Linux、Amazon Linux 2、Ubuntu 18.x、20.x、22.x 和 SUSE Linux Enterprise 12.x、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。

如果命令返回 `multi-user.target`，则表明 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。

安装 glxinfo 实用程序

运行以下命令：

- RHEL 7.x/8.x, CentOS 7.x/8.x 和亚马逊 Linux 2

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

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

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

- SUSE Linux Enterprise 12.x

```
$ 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 的 Linux 服务器并打算使用 OpenGL，请确保 Mesa 驱动程序已安装在 Linux 服务器上并正确配置。

### 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 \(p. 16\)](#)
- [安装和配置 AMD 驱动程序 \(p. 18\)](#)

### 安装和配置 NVIDIA 驱动程序

对于拥有专用 NVIDIA GPU 的 Linux 服务器，请确保相应的 NVIDIA 驱动程序已安装且正确配置。有关如何在 Amazon EC2 Linux 实例上安装 NVIDIA 驱动程序的说明，请参阅在 [Linux 服务器上安装 NVIDIA 驱动程序](#) 中的适用于 Linux 实例 Amazon EC2 用户指南。

## Note

- 这仅适用于拥有 NVIDIA GPU 的 Linux 服务器。
- GRID 驱动程序支持每个 GPU 一个 GPU 最多四个 4K 显示器。对于安装的每个 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，请更新 `xorg.conf`。

## 生成更新的 `xorg.conf`

1. 运行以下命令。

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

如果您使用的是 G3 或 G4 Amazon EC2 实例，并且要使用多显示器控制台会话，请将 `--connected-monitor=DFP-0,DFP-1,DFP-2,DFP-3` 参数。这如下所示。

```
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的 `/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。

- RHEL 7.x CentOs 7.x、Amazon Linux、Amazon Linux Enterprise 12.x、Ubuntu 18.x、

```
$ 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，并且其计算能力大于或等于 3.0，或者大于或等于 3.5（对于 Ubuntu 20）。

要验证 NVENC 支持，请参阅 [NVIDIA 视频编码和解码 GPU Support 矩阵](#)。要检查计算能力，请参阅 [NVIDIA 计算容量表](#)。

如果您的 NVIDIA GPU 不支持 NVENC 编码，或者它不具备所需的计算能力，则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[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[深入了解新的 Amazon EC2 G4ad 实例](#)博客 package。

## 为非 GPU 实例安装 xDummy 驱动程序

主题

- [安装和配置 xDummy 驱动程序 \(p. 18\)](#)

## 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 Xdummy 驱动程序已安装且正确配置。xDummy 驱动程序允许 X 服务器在没有真正的 GPU 时使用虚拟帧缓冲区运行。

Note

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

## 安装 xDummy 驱动程序

运行以下命令:

- RHEL 7.x/8.x, CentOS 7.x/8.x 和亚马逊 Linux 2

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

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

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

- SUSE Linux Enterprise 12.x

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

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

## 在 `xorg.conf` 中配置 xDummy

1. 打开 `/etc/X11/xorg.conf` 使用首选文本编辑器存储。
2. 将以下部分添加到配置中。

```
Section "Device"
    Identifier "DummyDevice"
```

```
Driver "dummy"
Option "ConstantDPI" "true"
Option "IgnoreEDID" "true"
Option "NoDDC" "true"
VideoRam 2048000
EndSection

Section "Monitor"
Identifier "DummyMonitor"
HorizSync 5.0 - 1000.0
VertRefresh 5.0 - 200.0
Modeline "1920x1080" 23.53 1920 1952 2040 2072 1080 1106 1108 1135
Modeline "1600x900" 33.92 1600 1632 1760 1792 900 921 924 946
Modeline "1440x900" 30.66 1440 1472 1584 1616 900 921 924 946
Modeline "1366x768" 72.00 1366 1414 1446 1494 768 771 777 803
Modeline "1280x800" 24.15 1280 1312 1400 1432 800 819 822 841
Modeline "1024x768" 18.71 1024 1056 1120 1152 768 786 789 807
EndSection

Section "Screen"
Identifier "DummyScreen"
Device "DummyDevice"
Monitor "DummyMonitor"
DefaultDepth 24
SubSection "Display"
Viewport 0 0
Depth 24
Modes "1920x1080" "1600x900" "1440x900" "1366x768" "1280x800" "1024x768"
virtual 1920 1080
EndSubSection
EndSection
```

#### Note

提供的配置就是一个示例。你可以添加更多模式，然后设置不同的模式virtual解析。您还可以配置多个虚拟监视器。

### 3. 重新启动 X 服务器以使更改生效。

- RHEL 7.x CentOS 7.x、Amazon Linux、Amazon Linux Enterprise 12.x、Ubuntu 18.x、

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

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

## 在 Linux 上安装 NICE DCV 服务器

NICE DCV 服务器的安装使用了一系列 RPM 或 .deb 程序包，具体取决于您的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 root 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 7.x

NICE DCV 服务器适用于基于 64 位 x86 和 64 位 ARM 架构的亚马逊 Linux 2、RHEL 和 CentOS 7.x 服务器。

## Important

这些区域有：`nice-dcv-gl`和`nice-dcv-gltest`程序包不适用于基于 64 位 ARM 架构的服务器。

### 在 Amazon Linux 2、RHEL 7.x 和 CentOS 7.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 服务器程序包使用安全 GPG 签名进行数字签名的。要允许程序包管理器验证程序包签名，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. 从下载程序包[NICE DCV 下载网站](#). 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了正确的存档。

- 64 位：

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2022.1/Servers/nice-dcv-2022.1-13300-el7-x86_64.tgz
```

- 64 位 Windows

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2022.1/Servers/nice-dcv-2022.1-13300-el7-aarch64.tgz
```

## Tip

这些区域有：[最新软件包](#)下载网站的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：

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

- 64 位 Windows

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

4. 提取的内容.tgz归档并导航到提取的目录。

- 64 位：

```
$ tar -xvzf nice-dcv-2022.1-13300-el7-x86_64.tgz && cd nice-dcv-2022.1-13300-el7-x86_64
```

- 64 位 Windows

```
$ tar -xvzf nice-dcv-2022.1-13300-el7-aarch64.tgz && cd nice-dcv-2022.1-13300-el7-aarch64
```

5. 安装 NICE DCV 服务器。

- 64 位：

```
$ sudo yum install nice-dcv-server-2022.1.13300-1.el7.x86_64.rpm
```

- 64 位 Windows

```
$ sudo yum install nice-dcv-server-2022.1.13300-1.el7.aarch64.rpm
```

6. (可选) 要在 NICE DCV 版本 2021.2 及更高版本中使用 Web 客户端, 请安装 nice-dcv-web-viewer 程序包。

- 64 位 :

```
$ sudo yum install nice-dcv-web-viewer-2022.1.13300-1.el7.x86_64.rpm
```

- 64 位 Windows

```
$ sudo yum install nice-dcv-web-viewer-2022.1.13300-1.el7.aarch64.rpm
```

7. (可选) 要使用虚拟会话, 请安装 nice-xdcv 程序包。

- 64 位 :

```
$ sudo yum install nice-xdcv-2022.1.433-1.el7.x86_64.rpm
```

- 64 位 Windows

```
$ sudo yum install nice-xdcv-2022.1.433-1.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 :

```
$ sudo yum install nice-dcv-gl-2022.1.973-1.el7.x86_64.rpm
```

### Note

您可以有选择性地安装 nice-dcv-glttest 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 一起使用 EnginFrame, 为安装 nice-dcv-simple-external-authenticator 程序包。

- 64 位 :

```
$ sudo yum install nice-dcv-simple-external-  
authenticator-2022.1.193-1.el7.x86_64.rpm
```

- 64 位 Windows

```
$ sudo yum install nice-dcv-simple-external-  
authenticator-2022.1.193-1.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
```

启用 EPEL 存储库后，运行以下命令安装 DKMS：

```
$ sudo yum install dkms
```

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pulseaudio-utils程序包已在您的系统上安装。使用以下命令安装它。

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

## RHEL/CentOS 8.x and Rocky Linux 8.5

NICE DCV 服务器适用于 RHEL、基于 64 位 x86 和 64 位 ARM 架构的 CentOS 8.x 服务器以及 Rocky Linux 8.5 或更高版本。

### Important

这些区域有：nice-dcv-gl和nice-dcv-gltest程序包不适用于基于 64 位 ARM 架构的服务器。

## 在 RHEL 8.x、CentOS 8.x 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 服务器程序包使用安全 GPG 签名进行数字签名的。要允许程序包管理器验证程序包签名，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. 从下载程序包[NICE DCV 下载网站](#)。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了正确的存档。

- 64 位：

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2022.1/Servers/nice-dcv-2022.1-13300-el8-x86_64.tgz
```

- 64 位 Windows

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2022.1/Servers/nice-dcv-2022.1-13300-el8-aarch64.tgz
```

### Tip

这些区域有：[最新软件包](#)下载网站的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：

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

- 6位：

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

4. 提取的内容.tgz归档并导航到提取的目录。

- 6位：

```
$ tar -xvzf nice-dcv-2022.1-13300-el8-x86_64.tgz && cd nice-dcv-2022.1-13300-el8-x86_64
```

- 6位：

```
$ tar -xvzf nice-dcv-2022.1-13300-el8-aarch64.tgz && cd nice-dcv-2022.1-13300-el8-aarch64
```

5. 安装 NICE DCV 服务器。

- 6位：

```
$ sudo yum install nice-dcv-server-2022.1.13300-1.el8.x86_64.rpm
```

- 6位：

```
$ sudo yum install nice-dcv-server-2022.1.13300-1.el8.aarch64.rpm
```

6. (可选) 如果您打算将 Web 客户端与 NICE DCV 版本 2021.2 及更高版本一起使用, 请安装nice-dcv-web-viewer程序包。

- 64 位：

```
$ sudo yum install nice-dcv-web-viewer-2022.1.13300-1.el8.x86_64.rpm
```

- 6位：

```
$ sudo yum install nice-dcv-web-server-2022.1.13300-1.el8.aarch64.rpm
```

7. (可选) 要使用虚拟会话, 请安装nice-xdcv程序包。

- 6位：

```
$ sudo yum install nice-xdcv-2022.1.433-1.el8.x86_64.rpm
```

- 6位：

```
$ sudo yum install nice-xdcv-2022.1.433-1.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。

- 6位：

```
$ sudo yum install nice-dcv-gl-2022.1.973-1.el8.x86_64.rpm
```

## Note

您可以有选择性地安装 `nice-dcv-glttest` 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 一起使用 EnginFrame，为安装 `nice-dcv-simple-external-authenticator` 程序包。

- 6位：

```
$ sudo yum install nice-dcv-simple-external-  
authenticator-2022.1.193-1.el8.x86_64.rpm
```

- 6位：

```
$ sudo yum install nice-dcv-simple-external-  
authenticator-2022.1.193-1.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
```

启用 EPEL 存储库后，运行以下命令安装 DKMS：

```
$ sudo yum install dkms
```

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 `pulseaudio-utils` 程序包已在您的系统上安装。使用以下命令安装它。

```
$ 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 企业服务器 (SLES) 12.x/15.x 服务器。

### 在 SLES 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 服务器程序包使用安全 GPG 签名进行数字签名的。要允许程序包管理器验证程序包签名，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. 从下载程序包[NICE DCV 下载网站](#). 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了正确的存档。

- SLES 12.x

```
$ curl -O https://d1uj6qtbmh3dt5.cloudfront.net/2022.1/Servers/nice-dcv-2022.1-13300-sles12-x86_64.tgz
```

- SLES 15.x

```
$ curl -O https://d1uj6qtbmh3dt5.cloudfront.net/2022.1/Servers/nice-dcv-2022.1-13300-sles15-x86_64.tgz
```

#### Tip

这些区域有：[最新软件包](#)下载网站的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. 提取的内容 .tgz 归档并导航到提取的目录。

- SLES 12.x

```
$ tar -xvzf nice-dcv-2022.1-13300-sles12-x86_64.tgz && cd nice-dcv-2022.1-13300-sles12-x86_64
```

- SLES 15.x

```
$ tar -xvzf nice-dcv-2022.1-13300-sles15-x86_64.tgz && cd nice-dcv-2022.1-13300-sles15-x86_64
```

5. 安装 NICE DCV 服务器。

- SLES 12.x

```
$ sudo zypper install nice-dcv-server-2022.1.13300-1.sles12.x86_64.rpm
```

- SLES 15.x

```
$ sudo zypper install nice-dcv-server-2022.1.13300-1.sles15.x86_64.rpm
```

6. (可选) 如果您打算将 Web 客户端与 NICE DCV 版本 2021.2 及更高版本一起使用，请安装 nice-dcv-web-viewer 程序包。

- SLES 12.x

```
$ sudo zypper install nice-dcv-web-viewer-2022.1.13300-1.sles12.x86_64.rpm
```

- SLES 15.x

```
$ sudo zypper install nice-dcv-web-viewer-2022.1.13300-1.sles15.x86_64.rpm
```

7. (可选) 要使用虚拟会话, 请安装 nice-xdcv 程序包。

- SLES 12.x

```
$ sudo zypper install nice-xdcv-2022.1.433-1.sles12.x86_64.rpm
```

- SLES 15.x

```
$ sudo zypper install nice-xdcv-2022.1.433-1.sles15.x86_64.rpm
```

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

- SLES 12.x

```
$ sudo zypper install nice-dcv-gl-2022.1.973-1.sles12.x86_64.rpm
```

- SLES 15.x

```
$ sudo zypper install nice-dcv-gl-2022.1.973-1.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 一起使用 EnginFrame, 为安装 nice-dcv-simple-external-authenticator 程序包。

- SLES 12.x

```
$ sudo zypper install nice-dcv-simple-external-  
authenticator-2022.1.193-1.sles12.x86_64.rpm
```

- SLES 15.x

```
$ sudo zypper install nice-dcv-simple-external-  
authenticator-2022.1.193-1.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

如果您使用的是 SLES 15 SP1 或 SP2，请将 **15** 在上面的命令中 **15.1** 要么 **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 `pulseaudio-utils` 程序包已在您的系统上安装。使用以下命令安装它。

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

## Ubuntu 18.04/20.04/22.04

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

这些区域有：`nice-dcv-gl`和`nice-dcv-gltest`程序包不适用于基于 64 位 ARM 架构的服务器。

要在 Ubuntu 上安装 NICE DCV 服务器 18.04/20.04/22.04

1. 启动并连接到您打算安装 NICE DCV 服务器的服务器。
2. NICE DCV 服务器程序包使用安全 GPG 签名进行数字签名的。要允许程序包管理器验证程序包签名，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. 从下载程序包 [NICE DCV 下载网站](#). 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了正确的存档。

- Ubuntu 18.04 ( 64 位 x86 )

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2022.1/Servers/nice-dcv-2022.1-13300-ubuntu1804-x86_64.tgz
```

- Ubuntu 18.04 (64 位 ARM)

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2022.1/Servers/nice-dcv-2022.1-13300-ubuntu1804-aarch64.tgz
```

- Ubuntu 20.04 ( 64 位 x86 )

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2022.1/Servers/nice-dcv-2022.1-13300-ubuntu2004-x86_64.tgz
```

- Ubuntu 2buntu 2buntu 2buntu 2

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2022.1/Servers/nice-dcv-2022.1-13300-ubuntu2204-x86_64.tgz
```

#### Tip

这些区域有：[最新软件包](#)下载网站的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 18.04 ( 64 位 x86 )

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-ubuntu1804-x86_64.tgz
```

- Ubuntu 18.04 (64 位 ARM)

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-ubuntu1804-aarch64.tgz
```

- Ubuntu 20.04 ( 64 位 x86 )

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

- Ubuntu 2buntu 2buntu 2buntu 2

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

4. 提取的内容.tgz归档并导航到提取的目录。

- Ubuntu 18.04 ( 64 位 x86 )

```
$ tar -xvzf nice-dcv-2022.1-13300-ubuntu1804-x86_64.tgz && cd nice-dcv-2022.1-13300-ubuntu1804-x86_64
```

- Ubuntu 18.04 (64 位 ARM)

```
$ tar -xvzf nice-dcv-2022.1-13300-ubuntu1804-arm64.tgz && cd nice-  
dcv-2022.1-13300-ubuntu1804-arm64
```

- Ubuntu 20.04 ( 64 位 x86 )

```
$ tar -xvzf nice-dcv-2022.1-13300-ubuntu2004-x86_64.tgz && cd nice-  
dcv-2022.1-13300-ubuntu2004-x86_64
```

- Ubuntu 2buntu 2buntu 2buntu 2

```
$ tar -xvzf nice-dcv-2022.1-13300-ubuntu2204-x86_64.tgz && cd nice-  
dcv-2022.1-13300-ubuntu2204-x86_64
```

5. 安装 NICE DCV 服务器。

- Ubuntu 18.04 ( 64 位 x86 )

```
$ sudo apt install ./nice-dcv-server_2022.1.13300-1_amd64.ubuntu1804.deb
```

- Ubuntu 18.04 (64 位 ARM)

```
$ sudo apt install ./nice-dcv-server_2022.1.13300-1_arm64.ubuntu1804.deb
```

- Ubuntu 20.04 ( 64 位 x86 )

```
$ sudo apt install ./nice-dcv-server_2022.1.13300-1_amd64.ubuntu2004.deb
```

- Ubuntu 2buntu 2buntu 2buntu 2

```
$ sudo apt install ./nice-dcv-server_2022.1.13300-1_amd64.ubuntu2204.deb
```

6. ( 可选 ) 如果您打算将 Web 客户端与 NICE DCV 版本 2021.2 及更高版本一起使用, 请安装 nice-dcv-web-viewer 程序包。

- Ubuntu 18.04 ( 64 位 x86 )

```
$ sudo apt install ./nice-dcv-web-viewer_2022.1.13300-1_amd64.ubuntu1804.deb
```

- Ubuntu 18.04 (64 位 ARM)

```
$ sudo apt install ./nice-dcv-web-viewer_2022.1.13300-1_arm64.ubuntu1804.deb
```

- Ubuntu 20.04 ( 64 位 x86 )

```
$ sudo apt install ./nice-dcv-web-viewer_2022.1.13300-1_amd64.ubuntu2004.deb
```

- Ubuntu 2buntu 2buntu 2buntu 2

```
$ sudo apt install ./nice-dcv-web-viewer_2022.1.13300-1_amd64.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 18.04 ( 64 位 x86 )

```
$ sudo apt install ./nice-xdcv_2022.1.433-1_amd64.ubuntu1804.deb
```

- Ubuntu 18.04 (64 位 ARM)

```
$ sudo apt install ./nice-xdcv_2022.1.433-1_arm64.ubuntu1804.deb
```

- Ubuntu 20.04 ( 64 位 x86 )

```
$ sudo apt install ./nice-xdcv_2022.1.433-1_amd64.ubuntu2004.deb
```

- Ubuntu 2buntu 2buntu 2buntu 2

```
$ sudo apt install ./nice-xdcv_2022.1.433-1_amd64.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 18.04 ( 64 位 x86 )

```
$ sudo apt install ./nice-dcv-gl_2022.1.973-1_amd64.ubuntu1804.deb
```

- Ubuntu 20.04 ( 64 位 x86 )

```
$ sudo apt install ./nice-dcv-gl_2022.1.973-1_amd64.ubuntu2004.deb
```

- Ubuntu 2buntu 2buntu 2buntu 2

```
$ sudo apt install ./nice-dcv-gl_2022.1.973-1_amd64.ubuntu2204.deb
```

### 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。

10. ( 可选 ) 如果您打算将 NICE DCV 与 NICE 一起使用 EnginFrame , 为安装 nice-dcv-simple-external-authenticator 程序包。

- Ubuntu 18.04 ( 64 位 x86 )

```
$ sudo apt install ./nice-dcv-simple-external-  
authenticator_2022.1.193-1_amd64.ubuntu1804.deb
```

- Ubuntu 18.04 (64 位 ARM)

```
$ sudo apt install ./nice-dcv-simple-external-  
authenticator_2022.1.193-1_arm64.ubuntu1804.deb
```

- Ubuntu 20.04 ( 64 位 x86 )

```
$ sudo apt install ./nice-dcv-simple-external-  
authenticator_2022.1.193-1_amd64.ubuntu2004.deb
```

- Ubuntu 2buntu 2buntu 2buntu 2

```
$ sudo apt install ./nice-dcv-simple-external-  
authenticator_2022.1.193-1_amd64.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
```

安装 DKMS 后, 运行以下命令安装 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pulseaudio-utils程序包已在您的系统上安装。使用以下命令安装它。

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

## 安装后检查

本主题提供了在安装 NICE DCV 后应执行的一些安装后检查任务, 目的是确保 NICE DCV 服务器已正确配置。

### 目录

- [确保 NICE DCV 服务器可访问 \(p. 31\)](#)
- [确保 X 服务器可访问 \(p. 31\)](#)
- [验证 DCV GL 是否正确安装 \(p. 32\)](#)
- [验证 NICE DCV DEB 软件包签名 \(p. 33\)](#)

## 确保 NICE DCV 服务器可访问

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。如果您的防火墙阻止通过端口 8443 访问, 则必须更改 NICE DCV 服务器的通信端口。有关更多信息, 请参阅[更改 NICE DCV 服务器 TCP/UDP 端口和监听地址 \(p. 49\)](#)。

此外, 如果您正在 EC2 实例上设置 NICE DCV, 请创建一个安全组。这是为了启用对 NICE DCV 服务器进行通信的端口的访问。有关更多信息, 请参阅[如何在 EC2 上配置安全组](#)。

## 确保 X 服务器可访问

必须确保 NICE DCV 控制台和虚拟会话可以访问 X 服务器。

### 控制台会话

安装 NICE DCV 服务器后, dcv 将创建用户。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。

如果命令没有返回 `SI:localuser:dcv`，则表明该 `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 7.x/8.x, CentOS 7.x/8.x、Amazon Linux、Ubuntu 18.x 和 SUSE Linux Enterprise 12.x 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\n\([^ ]+\).*\/\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。

如果命令没有返回 `LOCAL:`，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 7.x/8.x, CentOS 7.x/8x、Amazon Linux、Ubuntu 18.x 和 SUSE Linux Enterprise 12.x 12.x

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

```
$ sudo dcvgladmin disable
```

```
$ sudo dcvgladmin enable
```

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

## 验证 DCV GL 是否正确安装

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 dcvgladmin
```

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

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

验证 DEB 程序包签名

使用以下命令：

```
gpg1 --import NICE-GPG-KEY-SECRET  
dpkg-sig --verify nice-dcv-server_2022.1.13300-1_amd64.deb
```

这将返回一条包含该术语的消息GOODSIG以确认签名已验证。以下示例显示了签名确认消息。代替###，则会显示该密钥。

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

## 授予 NICE DCV 服务器的许可

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 版本 2017.0 及更高版本。

## NICE DCV 许可要求

主题

- [Amazon EC2 上的 NICE DCV \(p. 33\)](#)
- [本地服务器和其他基于云的服务器上的 NICE DCV \(p. 34\)](#)
- [微软远程访问 Windows 服务器的许可要求 \(p. 35\)](#)

## Amazon EC2 上的 NICE DCV

您不需要许可证服务器，即可在 EC2 实例上安装和使用 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，以确定是否提供了有效的许可证。

确保您的实例具有以下属性：

- 它可以访问 Amazon S3 终端节点。如果它能够访问 Internet，则使用 Amazon S3 公有终端节点进行连接。如果您的实例无法访问 Internet，请使用出站安全组规则或访问控制列表 (ACL) 策略为您的 VPC 配置网关终端节点，该策略允许您通过 HTTPS 访问 Amazon S3。有关更多信息，请参阅 [网关 VPC P P 中的 Amazon VPC User Guide](#)。如果您在连接到 S3 存储桶时遇到任何问题，请参阅 [为什么我无法使用网关 VPC 终端节点连接到 S3 存储桶？](#) 中的 Amazon 知识中心。
- 它有权访问所需的 Amazon S3 对象的访问权限。将以下 Amazon S3 访问策略添加到实例的 IAM 角色，并将##占位符和你的 Amazon 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": [  
    {  
      "Action": "s3:*",  
      "Resource": "arn:aws:s3:::##",  
      "Effect": "Allow",  
      "Principal": "*" } ] }
```

```
{
  "Effect": "Allow",
  "Action": "s3:GetObject",
  "Resource": "arn:aws:s3:::dcv-license.region/*"
}
```

- 如果您使用的是 Windows 实例，请确保该实例可以访问实例元数据服务。需要对此服务的访问权限以确保可以正确授权 NICE DCV 服务器。有关实例元数据服务的更多信息，请参阅[实例元数据和用户数据](#)中的适用于 Windows 实例 Amazon EC2 用户指南。

如果您使用的是自定义 Windows AMI，则必须安装 EC2Config 服务（Windows Server 2012 R2 及以前的版本）或 EC2Launch（Windows Server 2016 及更高版本）。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[使用 EC2Config 服务配置 Windows 实例](#)要么[使用 EC2Launch 配置 Windows 实例](#)中的适用于 Windows 实例 Amazon EC2 用户指南。

如果您正在 Amazon EC2 实例上安装和使用 NICE DCV 服务器，可以跳过本章的其余部分。本章的其余部分仅适用于在本地服务器或其他基于云的服务器上使用 NICE DCV 服务器。

## 本地服务器和其他基于云的服务器上的 NICE DCV

需要有许可证才能在本地服务器或其他基于云的服务器上安装和使用 NICE DCV 服务器。提供了以下许可选项：

- 自动评估许可证— 当您安装 NICE DCV 服务器时，会自动安装此类许可证。这种类型的许可证在安装之后的 30 天内有效。在许可证过期后，您不再能够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

如果没有配置任何其他许可证，NICE DCV 服务器默认为使用自动评估许可证。

- 扩展评估许可证— 延期评估许可证是延长自动评估许可证提供的初始 30 天评估期的评估许可证。该期限由 NICE 在 case-by-case 基础。在到达到期日期后，延期评估许可证将变得无效，您此时不再能够在服务器上创建和托管 NICE DCV 会话。必须向上列出的 NICE 分销商或经销商申请延期评估许可证[如何购买 NICE 网站](#)的页面。这些许可证以许可证文件的形式提供，必须安装在 NICE DCV 服务器上。
- 生产许可— 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。借助浮动许可证，您可以在网络中运行多个 NICE DCV 服务器。同时，您还可以限制您可以跨所有服务器创建的并发 NICE DCV 会话的数量。每个并发的 NICE DCV DCV 会话都需要一个许可证。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— 永久许可证没有到期日期，可以不限时间地使用。
  - 订阅— 订阅的有效期限有限，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，您不再能够在 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[如何购买](#)在 NICE 网站上找到您所在地区的 NICE 分销商或经销商。

## 许可证 (e)

- NICE DCV 客户端不需要许可证。
- NICE DCV 服务器许可证文件向后兼容以前版本的 NICE DCV 服务器。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 结合使用。
- NICE DCV 服务器版本至少需要相同版本的 NICE DCV 服务器许可证。例如，如果您使用的是 NICE DCV 服务器版本 2021，则需要许可证版本为 2021 或更高版本。如果升级到更高版本的 NICE DCV 服务器，则必须请求兼容的许可证文件。有关更多信息，请联系您的 NICE DCV 分销商或经销商。

## Note

有关 NICE DCV 服务器兼容性的信息，请参阅[兼容性注意事项](#) (p. 43)。

## 微软远程访问 Windows 服务器的许可要求

微软要求，除了 Windows 服务器客户端访问许可证 (CAL) 外，还必须为每位远程访问服务器图形用户界面 (GUI) 的用户的 Windows Server 版本提供 Windows 服务器远程桌面服务 (RDS) CAL。无论您使用哪种远程显示协议，都需要这样做。如果您使用 NICE DCV 访问远程 Windows 服务器主机的 GUI，也需要此许可证。

如果您在 Amazon EC2 实例上运行 NICE DCV 服务器，并且使用[Windows ServerAMI](#)，亚马逊负担 Windows Server CAL 的许可费用，并提供两个仅用于管理目的的 Windows Server RDS CAL。这仅用于测试、维护和管理。

有关更多信息，请参阅 [微软产品条款网站](#)。如果您对 Microsoft 软件的授权或权利有任何疑问，请咨询您的法律团队、Microsoft 或 Microsoft 分销商。

## 安装延期评估许可证

当您从 NICE 请求延期评估许可证时，您会收到一个 `license.lic` 定义许可证的文件。

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，将 `license.lic` 在服务器上的其他文件夹中，必须更新 `license-file` 配置参数，以便它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](#) (p. 35)
- [更改 Linux 服务器上的许可证路径](#) (p. 36)

## 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

本主题中的以下部分说明了如何购买和使用生产许可证（永久许可证或订阅）。

主题

- [第 1 步：安装动 RLM 服务器 \(p. 36\)](#)
- [第 2 步：获取 RLM 服务器的主机 ID \(p. 37\)](#)
- [第 3 步：购买永久许可证或订阅 \(p. 38\)](#)
- [第 4 步：修改许可证文件 \(p. 38\)](#)
- [第 5 步：配置动 RLM 服务器 \(p. 39\)](#)
- [第 6 步：配置 NICE DCV 服务器 \(p. 42\)](#)

### 第 1 步：安装动 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 \(p. 36\)](#)
- [在 Linux 上安装 RLM 服务器 \(p. 37\)](#)

### 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

建议安装最新稳定版本的 RLM 许可证管理程序包。

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

建议安装最新稳定版本的 RLM 许可证管理程序包。

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根账户用于此值。

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

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

## 第 2 步：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，您需要提供此主机 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
```

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。您在下一个步骤中需要用到它。

### 在 Linux 上获取 RLM 服务器主机 ID

获取服务器的主机 ID

导航到 `/opt/nice/rlm/` 并运行以下命令。

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

该命令返回 RLM 服务器的主机 ID，如下所示。

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

记录主机 ID。您在下一个步骤中需要用到它。

## 第 3 步：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[如何购买](#)在 NICE 网站上找到您所在地区的 NICE 分销商或经销商。

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

## 第 4 步：修改许可证文件

当您购买 NICE DCV 永久许可证或订阅时，您会收到一个 `license.lic` 定义许可证的文件。`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 服务器的主机 ID。
- RLM 服务器的 TCP 端口号。默认为 5053。
- ISV 端口号。这是 RLM 服务器用来侦听 NICE DCV 许可证请求的可选端口。如果未指定，RLM 会在启动时选择一个随机端口。
- 许可证授权的 NICE DCV 产品，以及每种产品的以下详细信息：
  - 许可证所涵盖的主要版本（例如，2017 适用于 2017 年 NICE DCV 产品）。
  - 到期日期。`Permanent` 指示许可证不过期。
  - 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 `dcv-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. 使用首选文本编辑器打开该文件。
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) 在文件中的行中添加 ISV 端口号, 该行以开头 `ISV`, 通过添加 `port=port_number`。要启用与 DCV 服务器的通信, 需要使用此端口。

如果您不想指定 ISV 端口, 请省略 `port=port_number`。如果不指定 ISV 端口, RLM 在每次启动时使用随机端口。

### Warning

如果防火墙设置阻止使用随机选择的端口, 则除了在 `HOSTLins`。

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。

## 第 5 步 : 配置动 RLM 服务器

在修改许可证文件后, 您必须将该文件放置到 RLM 服务器上, 然后启动 RLM 服务。

### 主题

- [在 Windows 上配置 RLM 服务器 \(p. 39\)](#)
- [在 Linux 上配置 RLM 服务器 \(p. 40\)](#)

## 在 Windows 上配置 RLM 服务器

### 在 Windows 上配置 RLM 服务器

1. Connect 您的 RLM 服务器。
2. 将编辑后的 `license.lic` 文件复制到 `C:\RLM\license\`。
3. 复制 `C:\Program Files\NICE\DCV\Server\license\nice.set` 来自你的 NICE DCV 服务器的文件并将其放在 `C:\RLM\RLM 服务器上的文件夹`。
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 [RLM 许可证管理手册](#)。

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
```

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. 复制/usr/share/dcv/license/nice.set来自你的 NICE DCV 服务器的文件然后把它放进去/opt/nice/rlm在您的 RLM 服务器上的。
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"
    fi
}
```

```
        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。

## 第 6 步：配置 NICE DCV 服务器

将 NICE DCV 服务器配置为使用 RLM 服务器。为此，您必须配置 `license-file` NICE DCV 服务器上的配置参数。

#### 主题

- [Windows NICE DCV 服务器配置 \(p. 42\)](#)
- [Linux NICE DCV 服务器配置 \(p. 42\)](#)

### 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 `5053@RLM_server_hostname` 格式的 RLM 服务器的端口号和主机名。

#### Note

您可以使用 RLM 服务器 IP 地址而不是主机名。

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 `5053@RLM_server_hostname` 格式的日期和时间。

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 = "5053@RLM_server_hostname"
```

#### Note

您可以使用 RLM 服务器 IP 地址而不是主机名。

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 服务器上更新了许可证，NICE DCV 服务器会自动更新正在运行的会话的已用许可证。以下过程详细说明了如何在 RLM 上更新 DCV 许可证。

#### 更新 RLM 服务器上的 DCV 许可证

1. 更新之前的许可证文件 [已安装 \(p. 39\)](#)。在 Linux 上，它应该放在 `/opt/dcv/rlm/license/license.lic` 在 Windows 上的 `C:\RLM\license\license.lic`。
2. 运行 `rlmutil rlmreread` 命令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 服务器应在几分钟内（通常为 5 分钟或更短）检查新许可证的使用情况。

从 NICE DCV 版本 2021.0 开始，您可以使用以下命令作为管理员为了立即强制更新使用授权：

```
$ dcv reload-licenses
```

## 升级 NICE DCV 服务器

以下主题介绍如何升级 NICE DCV 服务器。

#### 目录

- [兼容性注意事项 \(p. 43\)](#)
- [在 Windows 上升级 NICE DCV 服务器 \(p. 43\)](#)
- [在 Linux 上升级 NICE DCV 服务器 \(p. 44\)](#)

## 兼容性注意事项

NICE DCV 服务器版本 2017 及更高版本与 NICE DCV 客户端版本 2017 及更高版本兼容。

#### Note

有关本地和非基于 EC2 的服务器上的 NICE DCV 服务器许可兼容性要求的信息，请参阅 [许可证 \(e\) \(p. 34\)](#)。

## 在 Windows 上升级 NICE DCV 服务器

在 Windows 上升级 NICE DCV 服务器

1. 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` 用于检查所有正在运行的会话的 NICE DCV 命令。如果存在正在运行的会话，请使用 `dcv close session` NICE DCV 命令来阻止他们。
3. 在确认没有正在运行的会话后，停止 NICE DCV 服务器。有关更多信息，请参阅 [在 Windows 上停止 NICE DCV 服务器 \(p. 47\)](#)。
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. 从下载最新版本的 NICE DCV 服务器 [NICE 网站](#)。
6. 执行 [使用向导 \(p. 8\)](#) 中所述的步骤，从步骤 3 开始。
7. 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. 通过启动新的 NICE DCV 会话来测试 NICE DCV 服务器。有关更多信息，请参阅 [启动 NICE DCV 会话 \(p. 77\)](#)。

## 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` 用于检查所有正在运行的会话的 NICE DCV 命令。如果存在正在运行的会话，请使用 `dcv close session` NICE DCV 命令来阻止他们。
3. 在确认没有正在运行的会话后，停止 NICE DCV 服务器。有关更多信息，请参阅 [在 Linux 上停止 NICE DCV 服务器 \(p. 48\)](#)。
4. 对 NICE DCV 服务器配置进行备份。复制 `/etc/dcv/dcv.conf` 文件包到安全的位置。
5. 执行 [安装 NICE DCV 服务器 \(p. 19\)](#) 中所述的步骤。
6. 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. 通过启动新的 NICE DCV 会话来测试 NICE DCV 服务器。有关更多信息，请参阅 [启动 NICE DCV 会话 \(p. 77\)](#)。

## 卸载 NICE DCV 服务器

以下主题介绍如何卸载 NICE DCV 服务器。

### 目录

- [在 Windows 上卸载 NICE DCV 服务器 \(p. 44\)](#)
- [在 Linux 上卸载 NICE DCV 服务器 \(p. 45\)](#)

## 在 Windows 上卸载 NICE DCV 服务器

### 在 Windows 上卸载 NICE DCV 服务器

1. 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` 用于检查所有正在运行的会话的 NICE DCV 命令。如果存在正在运行的会话，请使用 `dcv close session` NICE DCV 命令来阻止他们。
3. 在确认没有正在运行的会话后，停止 NICE DCV 服务器。有关更多信息，请参阅 [在 Windows 上停止 NICE DCV 服务器 \(p. 47\)](#)。

4. 打开窗口设置应用程序并导航到应用程序包和功能面板。
5. 选择 NICE DCV 服务器，然后按卸载。
6. (可选) 您可能还希望删除由 NICE DCV 服务器生成的所有日志文件。在卸载完成后，导航到 C : ProgramData\不错\dcv\并删除日志 folder。

## 在 Linux 上卸载 NICE DCV 服务器

NICE DCV 服务器的安装使用了一系列 RPM 或 .deb 程序包，具体取决于您的主机服务器的操作系统。

### Note

您必须以 root 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` 用于检查所有正在运行的会话的 NICE DCV 命令。如果存在正在运行的会话，请使用 `dcv close session` NICE DCV 命令来阻止他们。
2. 在确认没有正在运行的会话后，停止 NICE DCV 服务器。有关更多信息，请参阅 [在 Linux 上停止 NICE DCV 服务器 \(p. 48\)](#)。
3. 卸载 NICE DCV 服务器程序包。根据您的执行安装的方式，某些软件包可能未安装在系统上，可以从命令中省略。有关可选程序包的列表，请参阅 [在 Linux 上安装 NICE DCV 服务器 \(p. 10\)](#)。

### Amazon Linux 2 and RHEL/CentOS 7.x/8.x

```
$ 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 18.04/20.04

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

4. (可选) 您可能还希望删除由 NICE DCV 服务器生成的所有日志文件。在卸载完成后，导航到 `/var/package` 并删除 DCV folder。

# 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 \(p. 46\)](#)
- [停止 NICE DCV 服务器 \(p. 47\)](#)
- [启用 QUIC UDP 传输协议 \(p. 48\)](#)
- [更改 NICE DCV 服务器 TCP/UDP 端口和监听地址 \(p. 49\)](#)
- [管理 TLS 证书 \(p. 52\)](#)
- [断开空闲客户端连接 \(p. 53\)](#)
- [在 Linux NICE DCV 服务器上启用 GPU 共享 \(p. 55\)](#)
- [启用触摸屏和触控笔支持 \(p. 56\)](#)
- [启用游戏手柄支持 \(p. 58\)](#)
- [启用 USB 远程控制 \(p. 58\)](#)
- [配置智能卡缓存 \(p. 60\)](#)
- [启用会话存储 \(p. 60\)](#)
- [在 Linux NICE DCV 服务器上配置打印机 \(p. 62\)](#)
- [在 Linux NICE DCV 服务器上配置剪贴板 \(p. 63\)](#)
- [配置多通道音频 \(p. 64\)](#)
- [配置 HTTP 标头 \(p. 67\)](#)
- [配置 NICE DCV 身份验证 \(p. 68\)](#)
- [配置 NICE DCV 授权 \(p. 70\)](#)

## 启动 NICE DCV 服务器

NICE DCV 服务器必须正在运行才能承载会话。

默认情况下，NICE DCV 服务器在所托管于的服务器启动时启动。如果您选择禁止服务器在安装 NICE DCV 服务器时自动启动，则必须手动启动服务器或重新设置自动启动。要执行任一选项，请遵循以下步骤之一。

## 主题

- [在 Windows 上启动 NICE DCV 服务器 \(p. 46\)](#)
- [在 Linux 上启动 NICE DCV 服务器 \(p. 47\)](#)

## 在 Windows 上启动 NICE DCV 服务器

使用微软管理控制台的“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. 选择开始。

#### Note

如果服务器已启动并运行，则该服务器已在运行中，启动按钮被禁用。

使用 Microsoft 管理控制台的“服务”管理单元配置自动启动。

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。

## 在 Linux 上启动 NICE DCV 服务器

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 7.x/8.x、CentOS 7.x/8.x Enterprise 12 和 Ubuntu 18.x

```
$ sudo systemctl start dcvserver
```

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 7.x/8.x、CentOS 7.x/8.x 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。

主题

- [在 Windows 上停止 NICE DCV 服务器 \(p. 47\)](#)
- [在 Linux 上停止 NICE DCV 服务器 \(p. 48\)](#)

## 在 Windows 上停止 NICE DCV 服务器

使用微软管理控制台的“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 Stop (停止)。

#### Note

如果服务器已停止，则停止按钮处于禁用状态。

通过 Microsoft 管理控制台的“服务”管理单元禁止自动启动。

### 在 Windows 上阻止 NICE DCV 服务器自动启动

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，选择手动。

## 在 Linux 上停止 NICE DCV 服务器

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：

- RHEL 7.x/8.x、CentOS 7.x/8.x 和 SUSE Linux Enterprise 12

```
$ sudo systemctl stop dcvserver
```

使用命令行禁用自动 NICE DCV 服务器启动。

在 Linux 上阻止 NICE DCV 服务器自动启动

使用以下命令：

- RHEL 7.x/8.x、CentOS 7.x/8.x 和 SUSE Linux Enterprise 12

```
$ sudo systemctl disable dcvserver
```

## 启用 QUIC UDP 传输协议

默认情况下，NICE DCV 使用 WebSocket 协议，基于 TCP，用于数据传输。

您可以将 NICE DCV 配置为使用 QUIC 协议进行数据传输。此传输协议基于 UDP。如果您的网络出现高延迟和数据包丢失，使用 QUIC 可能会提高性能。如果启用 QUIC，NICE DCV 服务器将使用 QUIC 协议进行数据传输。但是，它继续使用 WebSocket 用于身份验证流量。

#### Note

仅当网络和安全配置允许 UDP 流量时，才能使用 QUIC。

如果启用 QUIC，则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，则使用 WebSocket。有关 QUIC 协议的更多信息，请参阅[正在连接到 NICE DCV 会话](#)中的 NICE DCV 用户指南。

#### Windows NICE DCV server

将 NICE DCV 配置为使用 QUIC (UDP) 进行数据传输

1. 打开 Windows 注册表编辑器，并导航到 HKEY\_USERS/GSettings/com/Software/dcv/connectivity//键。
2. 打开 enable-quick-frontend 参数。适用于 Value data，输入 1。

#### Note

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

3. 关闭 Windows 注册表编辑器。
4. [停止 \(p. 47\)](#)和[重新启动 \(p. 46\)](#)NICE DCV 服务器。

#### Linux NICE DCV server

将 NICE DCV 配置为使用 QUIC (UDP) 进行数据传输

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 \(p. 47\)](#)和[重新启动 \(p. 46\)](#)NICE DCV 服务器。

## 更改 NICE DCV 服务器 TCP/UDP 端口和监听地址

默认情况下，NICE DCV 服务器配置为监听 TCP 端口 8443 并在其运行的主机上的任何网络接口上进行通信。

在安装 NICE DCV 服务器后。如果您将 NICE DCV 服务器配置为[启用 QUIC \(p. 48\)](#)，您还可以为 QUIC 流量指定自定义 UDP 端口。该端口号必须高于 1024。

您可以指定 NICE DCV 服务器侦听的网络地址。例如，这允许您指定是只使用 IPv4 还是 IPv6。它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

每当您对 NICE DCV 服务器的网络配置应用更改时，请确保将更改传达给客户端，例如，客户端需要知道用于连接会话的端口号。

#### Tip

控制向客户端公开的网络地址和端口的另一种方法是使用[NICE DCV 连接网关](#)或作为服务器前端的另一个 Web 代理或负载均衡器。通过网关访问您的 NICE DCV 服务器主机可以让您的服务器拥有一个单一的地址。它还允许使用低于 1024 的端口号，包括 443 ( HTTPS 的标准端口号 )。

有关配置网关的网络地址和端口的更多信息，请参阅网关的文档。

#### 主题

- [更改 NICE DCV 服务器的 TCP 端口 \(p. 50\)](#)
- [监听特定的端点 \(p. 51\)](#)

## 更改 NICE DCV 服务器的 TCP 端口

### Windows NICE DCV server

要更改 NICE DCV 服务器使用的端口，请配置 `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。

如果没有 `web-port` 该注册表项中的参数，创建一个：

- a. 在导航窗格中，打开上下文 (右键单击) 菜单。连接键。然后，选择 `New`、`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。如果未配置此参数，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。

如果没有 `quic-port` 该注册表项中的参数，创建一个：

- a. 在导航窗格中，打开上下文 (右键单击) 菜单。连接键。然后，选择 `New`、`DWORD (32 位)` 值。
  - b. 对于名称，输入 `quic-port` 并按 `Enter`。
6. 打开 `quic-port` 参数。适用于 `Value data` 中，输入新的 UDP 端口号。如果您未配置此参数并且启用了 QUIC 支持，则 NICE DCV 服务器默认使用 UDP 端口 8443。

#### Note

UDP 端口号必须高于 1024。

7. 选择 `确定`，并关闭 Windows 注册表编辑器。
8. [停止 \(p. 47\)](#) 和 [重新启动 \(p. 46\)](#) NICE DCV 服务器。

### Linux NICE DCV server

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

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 TCP 端口号替换为新的 TCP 端口号。

如果没有web-port参数[connectivity]部分中，使用以下格式手动添加此参数：

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

#### Note

TCP 端口号必须高于 1024。

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。

如果没有web-port参数[connectivity]部分中，使用以下格式手动添加此参数：

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

#### Note

UDP 端口号必须高于 1024。

4. 保存并关闭 文件。
5. [停止 \(p. 47\)](#)和[重新启动 \(p. 46\)](#)NICE DCV 服务器。

## 监听特定的端点

要仅监听特定的网络地址，可以将web-listen-endpoints和quic-listen-endpointsNICE DCV 服务器配置中的参数。

每个端点都由一个 IPv4 或 IPv6 地址表示，后面可以选择用一个端口号分隔。终端节点中指定的端口号优先于在web-port和quic-port参数。

由于可以指定多个端点，因此一组端点由逗号分隔的列表表示，该列表用方括号括起来，其中每个端点都放在单引号之间。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eth1Linux 主机上的网络接口。

#### Note

这些配置参数只能从 NICE DCV 服务器 2022.0 开始使用。

### Windows NICE DCV server

在 Windows 上更改服务器的端点

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 端点，请选择web-listen-endpoints参数。

如果没有web-listen-endpoints该注册表项中的参数，创建一个：

- a. 在导航窗格中，打开上下文 (右键单击) 菜单。连接键。然后，选择New、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。适用于Value data中，输入终端节点的列表。

5. 如果启用了 QUIC，要配置 UDP 端点，请选择 `quic-listen-endpoints` 参数。

如果没有 `quic-listen-endpoints` 该注册表项中的参数，创建一个：

- a. 在导航窗格中，打开上下文 (右键单击) 菜单。连接键。然后，选择 New、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。适用于 Value data，输入终端节点的列表。
  7. 选择确定，并关闭 Windows 注册表编辑器。
  8. [停止 \(p. 47\)](#) 和 [重新启动 \(p. 46\)](#) NICE DCV 服务器。

## Linux NICE DCV server

### 在 Linux 上更改服务器的端点

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，替换现有的端点列表。

如果没有 `web-listen-endpoints` Parameter `[connectivity]` 部分中，使用以下格式手动添加此参数：

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

3. 在 `[connectivity]` 部分中找到 `quic-listen-endpoints` 参数。然后，替换现有的端点列表。

如果没有 `quic-listen-endpoints` Parameter `[connectivity]` 部分中，使用以下格式手动添加此参数：

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

4. 保存并关闭文件。
5. [停止 \(p. 47\)](#) 和 [重新启动 \(p. 46\)](#) NICE DCV 服务器。

## 管理 TLS 证书

NICE DCV 自动生成自签名证书，该证书用于保护 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。默认证书包括两个文件。它们本身就是证书 (`dcv.pem`) 还有一把钥匙 (`dcv.key`)。

当 DCV 客户端用户连接到服务器时，他们可能会在建立连接之前收到服务器证书警告，他们可以根据这些警告进行验证。

如果他们使用 Web 浏览器进行连接，浏览器可能会警告客户端用户信任服务器的证书，他们应联系管理员以确认证书的真实性。

同样，如果他们使用的是 Windows、Linux 或 macOS 客户端，则可能建议他们向 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` 并根据他们的证书指纹检查输出。

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。

生成自己的证书时，请选择符合您特定需求的证书属性。这些区域有：CN (Common Name)属性在大多数情况下必须与主机的公有主机名匹配。您可能还想指定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，则必须为证书命名 `dcv.pem` 而且你必须命名密钥 `dcv.key`。

#### Windows NICE DCV server

在 Windows 上更改服务器的 TLS 证书

- 将您的证书及其密钥置于 Windows NICE DCV 服务器上的以下位置：

```
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. 将您的证书及其密钥置于 Linux NICE DCV 服务器上的以下位置：

```
/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，则新证书将自动重新加载。对于以前版本的 NICE DCV，您需要手动 [停止 \(p. 47\)](#) 和 [重新启动 \(p. 46\)](#) NICE DCV 服务器。

## 断开空闲客户端连接

您可以将 NICE DCV 配置为断开空闲客户端的连接。更具体地说，对于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，您可以执行此操作。默认情况下，NICE DCV 服务器在空闲 60 分钟 (一小时) 后断开 NICE DCV 客户端的连接。

您还可以将 NICE DCV 服务器配置为向空闲客户端发送通知。通知是通知他们他们的会话即将断开连接。仅 NICE DCV 服务器和客户端版本 2017.4 及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。

#### 主题

- 在 [Windows 上更改空闲超时期限 \(p. 54\)](#)

- [在 Linux 上更改空闲超时期限 \(p. 54\)](#)

## 在 Windows 上更改空闲超时期限

要更改 NICE DCV 服务器的空闲超时期限，您必须配置 `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。

如果找不到参数：

- a. 在导航窗格中，打开上下文 (右键单击) 菜单。连接键。然后，选择 New、DWORD (32 位) 值。
  - b. 对于名称，输入 `idle-timeout` 并按 Enter。
3. 打开 `idle-timeout` 参数。适用于 Value data，为空闲超时期限输入一个值 (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/GSettings/com/Software/dcv/connectivity//` 键并选择 `idle-timeout-warning` 参数。

如果找不到参数：

- a. 在导航窗格中，打开上下文 (右键单击) 菜单。连接键。然后，选择 New、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。适用于 Value data 中，输入在断开连接之前发送相关警告通知的秒数 (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。

## 在 Linux 上更改空闲超时期限

要更改 NICE DCV 服务器的空闲超时期限，您必须配置 `idle-timeoutParameterdcv.conf` 文件。

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，将现有超时期限替换为新的超时期限 (以分钟为单位，十进制)。

如果没有 `idle-timeoutParameter[connectivity]` 部分中，使用以下格式手动添加此参数：

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

要避免断开空闲客户端的连接，请输入 0。

3. (可选) 要将 NICE DCV 服务器配置为向空闲客户端发送超时通知，请在 `idle-timeout-warningParameter[connectivity]` 部分并指定在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。

4. 保存并关闭 文件。

## 在 Linux NICE DCV 服务器上启用 GPU 共享

利用 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 \(p. 75\)](#)。GPU 共享使您可以使用一个 NICE DCV 服务器并托管多个共享服务器的物理 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。
- 安装 NICE DCV `dcv-g1` 和 `nice-xdcv` 服务器上的程序包。
- 确保服务器至少有一个支持的 NVIDIA GPU。
- 在服务器上安装 NVIDIA GPU 驱动程序。需要正式的 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 实用程序可以自动创建 `xorg.conf` 文件并为所有可用的 NVIDIA GPU 配置此参数。

- 确保 X 服务器正在运行。
- ( 可选 ) 通过运行以下命令验证 NICE DCV 服务器配置 `dcv-g1diag` 工具。有关更多信息，请参阅 [安装后检查 \(p. 31\)](#)。

您也可以安装 `nice-dcv-g1test` 打包并运行 `dcv-g1test` 测试应用程序以检查是否已为 GPU 共享正确配置服务器。

要启用 GPU 共享，您必须指定要由虚拟会话使用的 GPU 的列表。如果不指定 GPU，则仅指定标准 X 服务器所使用的 GPU ( 带显示名称 ) :0.0，被使用。

在中指定 `GPUg1-displays` 参数 `dcv.conf` 文件后，请完成本主题前面所述的先决条件。

### 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 `g1-displays` 参数。然后，采用以下格式指定可用的 GPU：

```
[display/linux]  
g1-displays = [':xserver_port.screen_number_1', ':xserver_port.screen_number_2', ...]
```

其中 `xserver_port` 是服务器和 `####` 是与 GPU 相关的屏幕关联的数字。 `####` 开始于 0。

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 \(p. 47\)](#)和[重新启动 \(p. 46\)](#)NICE DCV 服务器。

## 启用触摸屏和触控笔支持

- Windows 服务器支持

所有受支持的 Windows 操作系统都支持触摸屏。从 Windows 10 和 Windows 2019 开始，所有受支持的 Windows 操作系统都支持手写笔，但在 Windows 2016、Windows 8.1 及更早版本上不支持手写笔。默认情况下，Windows NICE DCV 服务器上已启用这些功能。无需其他配置。

- Linux 服务器支持

所有受支持的 Linux 操作系统都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

如果支持 NICE DCV，则启用触摸屏和触控笔与 NICE DCV 配合使用在客户端和服务器的服务器上，并在服务器上启用。有关客户端支持的信息，请参阅[客户机功能](#)中的 NICE DCV 用户指南。

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
```

- 保存更改并关闭该文件。
- 重新启动 X 服务器。
  - RHEL Linux EnCentOS e 12.x、Ubuntu 18.x 和 SUSE Linux Enterprise 12.x

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

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

- 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)]
```

## 配置测针压力范围

有些应用要求您将触控笔的压力范围减小到 0 到 2048 之间。您可以设置压力范围 `Pressure2k` 选项设置为 `true` / `etc/X11/xorg.conf` 文件。

### 配置光笔压力

- 使用所需的文本编辑器打开 `/etc/X11/xorg.conf`。
- 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
```

- 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，游戏手柄设备可以在连接到任何受支持的 Windows 或 Linux 操作系统时使用。

支持以下游戏手柄设备：

- 能力
- DualShock 4 控制器

与上面列出的设备兼容，或者可以配置为模拟其中一个受支持设备的其他设备也可以工作。

### Note

只有在使用 Windows 本机 NICE DCV 客户端时才支持 Gamepad 设备。确保您使用的是 NICE DCV 客户端 2022.0 或更高版本。

要启用游戏手柄支持，请确保您安装的是最新的 NICE DCV 服务器版本，并且已选择安装 GamePad 驱动程序。有关更多信息，请参阅 [在 Windows 上安装 NICE DCV 服务器 \(p. 6\)](#)。安装驱动程序时，默认情况下，Windows NICE DCV 服务器上已启用此功能。

## 支持 Xbox 360 控制器

Xbox 360 控制器需要安装其 Windows 驱动程序。此驱动程序不会自动安装在 Windows 上，需要从 Windows 更新官方网站上检索。

下载并安装 Xbox 360 控制器驱动程序：

1. 在 Microsoft 更新目录页上搜索驱动程序：<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驱动程序。
3. 打开 .cab 文件并解压其内容：

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

4. 使用以下命令安装 .inf 文件：

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

## 启用 USB 远程控制

NICE DCV 使客户可以使用各种专门的 USB 设备，如 3D 指点设备或身份验证设备。设备物理连接到其计算机，以便与在 NICE DCV 服务器上运行的应用程序交互。

### Important

NICE DCV 提供了一种用于重定向 USB 设备的通用机制。某些对网络延迟敏感的设备可能会遇到问题。此外，由于驱动程序兼容性问题，某些设备可能无法按预期运行。在部署到生产环境之前，请确保您的设备按预期工作。

## Note

仅 Windows 客户端支持 USB 远程处理。便携式 Windows 客户端或 Web 浏览器客户端不支持 USB 远程处理。NICE DCV 客户端上可能需要其他配置。有关更多信息，请参阅 [使用 USB 远程处理](#) 中的 NICE DCV 用户指南。

NICE DCV 服务器使用允许列表来确定客户端可使用哪些 USB 设备。默认情况下，一些常用的 USB 设备会添加到允许列表。这意味着客户端可以将这些 USB 设备连接到他们的计算机，并在服务器上使用它们而无需任何其他配置。

但默认情况下，某些专用设备可能不会添加到允许列表中。必须先将这些设备手动添加到 NICE DCV 服务器上的允许列表中，之后它们才可供客户端使用。添加后，它们将出现在 Windows 客户端的设置菜单中。

## 主题

- [将 USB 设备添加到 Windows NICE DCV 服务器上的允许列表 \(p. 59\)](#)
- [在 Linux NICE DCV 服务器上 将 USB 设备添加到允许列表 \(p. 59\)](#)

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

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. 确保您安装的是最新的 NICE DCV 服务器版本，并且已选择安装 USB 远程处理驱动程序。有关更多信息，请参阅 [在 Windows 上安装 NICE DCV 服务器 \(p. 6\)](#)。
2. 在 NICE DCV 服务器上安装 USB 设备的硬件驱动程序。
3. 从客户端请求筛选器字符串。有关更多信息，请参阅 [使用 USB 远程处理](#) 中的 NICE DCV 用户指南。
4. 使用首选文本编辑器打开 `C:\Program Files\NICE\DCV\Server\conf\usb-devices.conf`，并将筛选器字符串添加到文件底部的新行。
5. 保存并关闭文件。
6. [停止和重新启动](#) NICE DCV 服务器。

## 在 Linux NICE DCV 服务器上 将 USB 设备添加到允许列表

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

### 在 Linux NICE DCV 服务器上 将 USB 设备添加到允许列表

1. 确保您已安装 NICE DCV 服务器和 DCV USB 驱动程序的最新版本。有关更多信息，请参阅 [在 Linux 上安装 NICE DCV 服务器 \(p. 10\)](#)。
2. 在 NICE DCV 服务器上安装 USB 设备的硬件驱动程序。
3. 从客户端请求筛选器字符串。有关更多信息，请参阅 [使用 USB 远程处理](#) 中的 NICE DCV 用户指南。
4. 使用首选文本编辑器打开 `/etc/dcv/usb-devices.conf`，并将筛选器字符串添加到文件底部的新行。
5. 保存并关闭文件。

6. [停止和重新启动](#)NICE DCV 服务器。

## 配置智能卡缓存

智能卡缓存功能使 NICE DCV 服务器能够缓存智能卡值。启用此功能时，NICE DCV 服务器将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。

默认情况下禁用智能卡缓存。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[使用智能卡](#)中的NICE DCV 用户指南。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，而不考虑为`DCV_PCSC_ENABLE_CACHE`环境变量。

### 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'`。

如果没有`enable-cacheParameter[smartcard]`部分中，使用以下格式手动添加此参数：

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

3. 保存并关闭 文件。
4. [停止](#) (p. 47)和[重新启动](#) (p. 46)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/18/Software/GSettings/com/dcv/Software/dcv/Software/`键并选择`enable-cache`参数。

如果此参数不存在：

- a. 在左窗格中，打开智能卡键，然后选择New、字符串值。
  - b. 对于名称，输入 `enable-cache` 并按 Enter。
3. 打开`enable-cache`参数。适用于Value data, 输入`always-on`永久启用智能卡缓存，或输入`always-off`永久禁用智能卡缓存。
  4. 选择确定，并关闭 Windows 注册表编辑器。

## 启用会话存储

会话存储是 NICE DCV 服务器上的一个文件夹，客户端在连接到特定的NICE DCV 会话时可访问该文件夹。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 (p. 61)
- 在 Linux NICE DCV 服务器上启用会话存储 (p. 61)

## 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_USERSSoftware/Software/dcv-session-management/automatic-console-session` 键并选择 `storage-root` 参数。

如果没有 `storage-root` 参数，请创建一个此参数，如下所示：

- i. 在导航窗格中，打开上下文（右键单击）菜单。会话管理/`automatic-console-session` 键。然后，选择 `New`、字符串。
  - ii. 对于名称，输入 `storage-root` 并按 `Enter`。
- c. 打开 `storage-root` 参数。适用于 `Value data`，输入步骤 1 中创建的文件夹的完整路径。

您还可以使用 `%home%` 指定当前已登录的用户的主目录。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，则将禁用会话存储。

- d. 选择 `确定`，并关闭 Windows 注册表编辑器。
  - e. [停止 \(p. 47\)](#) 和 [重新启动 \(p. 46\)](#) 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 \(p. 77\)](#)。

## 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，然后配置 `storage-rootParameterdcv.conf` 文件。

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。

如果没有 `storage-rootParameter[session-management/automatic-console-session]` 部分中，使用以下格式手动添加此参数。

```
[session-management/automatic-console-session]
```

```
storage-root="/opt/session-storage/"
```

您还可以使用%home%指定当前已登录的用户的主目录。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，则将禁用会话存储。

3. 保存并关闭 文件。
4. [停止 \(p. 47\)](#)和[重新启动 \(p. 46\)](#)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 \(p. 77\)](#)。

## 在 Linux NICE DCV 服务器上配置打印机

如果您正在使用受支持的 Linux 发行版，则必须将 NICE DCV 服务器配置为支持打印。Windows NICE DCV 服务器不需要其他配置。

在 Linux NICE DCV 服务器上启用打印机重定向

1. 安装CUPS服务器上的服务。

- Amazon Linux 2 和 CentOS

```
$ sudo yum install cups
```

- Ubuntu

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

- SUSE Linux Enterprise

```
$ sudo zypper install cups
```

2. 添加dcv打印机管理员组的用户。打印机管理员组的名称可以因操作系统。例如，如果您的打印机管理员组名为lpadmin，运行以下命令：

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

3. 确保打印机管理员组在SystemGroupcups 配置文件中的参数。例如，如果您的打印机管理员组名为lpadmin，使用文本编辑器打开/etc/cups/cups-file.conf并查找以下行。

```
SystemGroup lpadmin
```

如果配置文件中出现该行，则说明安装已完成。继续下一步。

如果该行未出现在配置文件中，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 企业版) 确保打印机管理员组具有读取 cups 本地证书的权限。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名为lpadmin，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 (p. 47)和重新启动 (p. 46)NICE DCV 服务器。

## 排查打印机问题

SUSE Linux Enterprise 和 RHEL 8 可能会阻止连接到打印机插座。如果您正在运行这些操作系统之一，并且存在打印问题，请检查日志文件以确定是否是造成这种情况的原因。

使用文本编辑器打开 `/var/log/audit/audit.log` 并检查你的日志是否有类似于以下内容的行：

```
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
```

如果日志文件中出现类似的行，则说明操作系统正在阻止访问打印机插座。

要解决此问题，必须创建允许访问打印机插座的 cups 策略。为此，请执行以下步骤：

1. 创建所需策略文件。使用首选文本编辑器，创建一个名为的新文件 `cupsd_policy` 并添加以下内容。

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

2. 安装策略。

```
$ cat cupsd_policy | audit2allow -M cupsd_policy_module
```

```
$ semodule -i cupsd_policy_module.pp
```

## 在 Linux NICE DCV 服务器上配置剪贴板

Linux 操作系统提供了两个可用于复制和粘贴内容的缓冲区。缓冲区是主要选择和剪贴板。要将内容复制到主选区，请通过拖动指针突出显示内容。要从主选区粘贴内容，请使用指针或 Shift+插入键盘快捷键。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 (p. 64)
- 将主要选择内容复制到客户端剪贴板 (p. 64)

## 将客户端剪贴板内容粘贴到主选区

默认情况下，客户端中复制的内容将置于剪贴板中。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，用户可以使用上下文（右键单击）菜单粘贴从两个剪贴板复制的内容。或者，他们可以使用鼠标的中间按钮或Shift+插入键盘快捷键。

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`。

如果没有 `primary-selection-paste` Parameter `[clipboard]` 部分中，使用以下格式手动添加此参数：

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

3. 保存并关闭 文件。
4. [停止 \(p. 81\)](#)和[重新启动 \(p. 77\)](#)NICE DCV 会话。

## 将主要选择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`。

如果没有 `primary-selection-copy` Parameter `[clipboard]` 部分中，使用以下格式手动添加此参数：

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

3. 保存并关闭 文件。
4. [停止 \(p. 81\)](#)和[重新启动 \(p. 77\)](#)NICE DCV 会话。

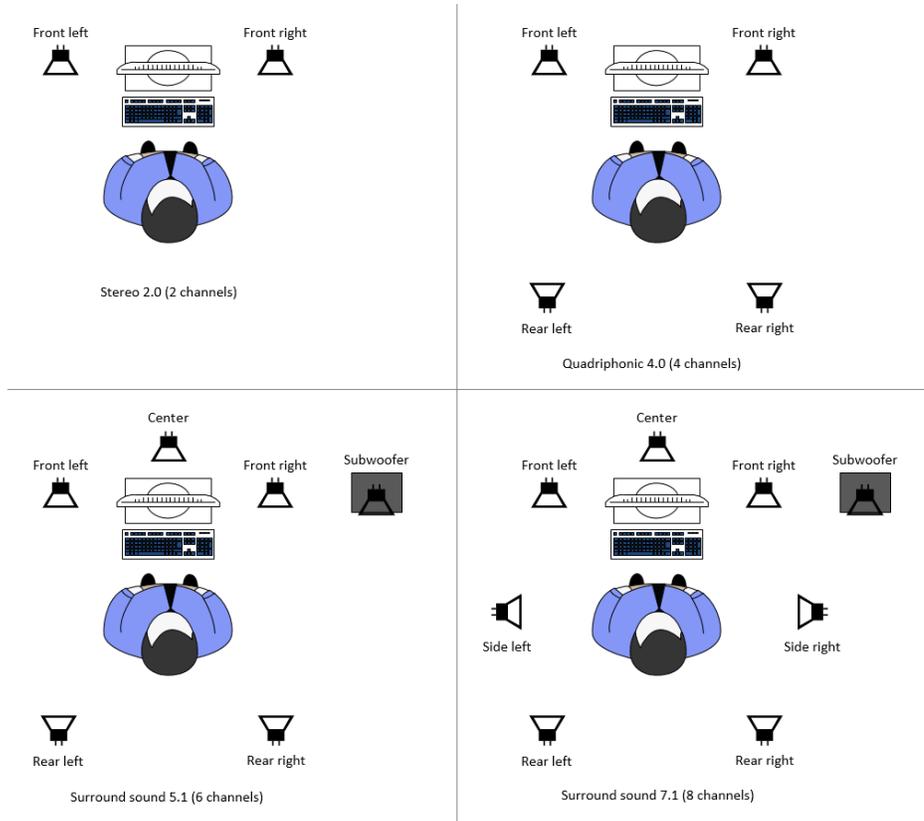
## 配置多通道音频

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

NICE DCV 支持以下多声道音频配置：

- 立体声 2.0 ( 双声道 )
- Quadriphonic 4.0 ( 四个通道 )

- 环绕声 5.1 ( 六个声道 )
- 环绕声 7.1 ( 八个声道 ) — 仅限 Windows NICE DCV 服务器



如果客户端请求的音频通道数少于服务器提供的通道数，则服务器会缩混声道数。这是为了匹配客户端请求的频道数。例如，假设客户端请求环绕声 5.1，而服务器最多支持 7.1 环绕声。服务器将音频缩混到 5.1。

服务器不会自动缩混音频以匹配源应用程序的音频输出。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。只有左前和右前音频通道会流式传输到客户端。剩下的通道会丢失。如果这是真的，为防止音频通道丢失，请将 NICE DCV 服务器配置为缩混音频通道。

#### 主题

- [在 Windows NICE DCV 服务器上配置音频通道 \(p. 65\)](#)
- [在 Linux NICE DCV 服务器上配置音频通道 \(p. 66\)](#)

## 在 Windows NICE DCV 服务器上配置音频通道

Windows 服务器支持环绕声 7.1 ( 八个音频通道 )。默认配置为立体声。但是，您可以将服务器配置为使用其他配置。

在 Windows 服务器上配置音频通道：

1. 打开声音控制面板。在桌面的任务栏中，右键单击“扬声器”图标，然后选择 SOUN.
2. 打开“播放”选项卡，然后选择 NICE DCV 扬声器。
3. 选择 Configure ( 配置 )。
4. 选择首选通道配置。

5. 选择 OK ( 确定 )。

## 在 Linux NICE DCV 服务器上配置音频通道

默认情况下，Linux 服务器支持立体声 2.0 ( 两个音频通道 )，并且需要一些额外的配置才能支持多声道音频。

您需要执行以下操作：

1. 配置 PulseAudio 声音服务器。
2. 配置 NICE DCV 服务器使用 PulseAudio 设备。
3. 配置要使用的通道数。

### 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。

在您配置了 PulseAudio 声音服务器，您必须将 NICE DCV 服务器配置为从 PulseAudio 声音服务器。

### 配置 NICE DCV 服务器使用 PulseAudio 设备

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，将现有值替换为您在上一步中检索的设备名称。

如果没有 `grab-device` Parameter `[audio]` 部分中，使用以下格式手动添加此参数：

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

3. 保存并关闭 文件。

配置 NICE DCV 服务器以捕获来自 PulseAudio 声音服务器，则可以指定要使用的通道数。

### 配置要使用的通道数

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，将现有频道数替换为以下各项之一：2 对于 2.0 或 6 对于 5.1。

如果没有 `source-channels` Parameter `[audio]` 部分中，使用以下格式手动添加此参数：

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

3. 保存并关闭 文件。
4. [停止 \(p. 47\)](#) 和 [重新启动 \(p. 46\)](#) NICE DCV 服务器。

## 配置 HTTP 标头

您可以将 NICE DCV 服务器配置为在用户使用 Web 浏览器客户端连接到会话时，向 NICE DCV 客户端发送其他 HTTP 响应标头。响应标头可以提供有关用户正在连接到的 NICE DCV 服务器的其他信息。

主题

- 在 [Windows NICE DCV 服务器上配置 HTTP 标头 \(p. 67\)](#)
- 在 [Linux NICE DCV 服务器上配置 HTTP 标头 \(p. 67\)](#)

### 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，打开上下文 (右键单击) 菜单。连接键。然后，选择 New、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。适用于 Value data，请输入以下格式的 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 \(p. 47\)](#)和[重新启动 \(p. 46\)](#)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 \(p. 47\)](#)和[重新启动 \(p. 46\)](#)NICE DCV 服务器。

## 配置 NICE DCV 身份验证

默认情况下，客户端在连接到NICE DCV会话之前需要针对托管 NICE DCV 的服务器进行身份验证。如果客户端未能进行身份验证，则可能是因为它被阻止连接到会话。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，身份验证工作委派给 WinLogon。对于 Linux NICE DCV 服务器，身份验证工作委派给 PAM。客户端在连接到NICE DCV 会话时提供其系统凭证。确保您的客户端拥有 NICE DCV 服务器上的相应用户账户的凭证。
- **none**— 连接到NICE DCV 会话时不需要客户端身份验证。NICE DCV 服务器授予对所有尝试连接到会话的客户端的访问权。

确保您的客户端了解 NICE DCV 服务器使用的身份验证方法。他们还应确保自己拥有连接到会话所需的信息。

主题

- [在 Windows 上配置身份验证 \(p. 68\)](#)
- [在 Lux 上配置身份验证 \(p. 69\)](#)

## 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。

如果没有 `authentication` 该注册表项中的参数，创建一个：

- a. 在导航窗格中，打开上下文 (右键单击) 菜单。身份验证键。然后，选择 New、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

Windows 凭证提供程序后，如果用户可以针对 DCV 服务器进行身份验证，则可以绕过 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\CurrentVersion\ Authenticaties\ {8A2C93D0-D55F-4045-99D7-B27F5E263407}键。
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\CurrentVersion\ Authenticaties\ {8A2C93D0-D55F-4045-99D7-B27F5E263407}键。
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 Lux 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，将现有值替换为 `system` 或 `none`。

如果没有 `authenticationParameter[security]` 部分中，使用以下格式添加此参数。

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

3. 保存并关闭 文件。

## Paramy 服务

在 Linux 上，当 NICE DCV `authentication` 参数设置为 `system`，则通过执行 PAM 服务执行身份验证。

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 PAM 在通过 NICE DCV 对用户进行身份验证时所执行的步骤，可以将 `pam-serviceParameterauthentication` 的部分 `dcv.conf`。

#### 更改 PAM 服务

1. 以 root 身份导航到 `/etc/pam.d` 目录并创建一个新文件，例如 `dcv-custom`。
2. 编辑 `dcv-custom` 文件使用首选文本编辑器。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。

如果没有 `pam-serviceParameter[authentication]` 部分中，使用以下格式手动添加此参数：

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

### Note

PAM 服务名称必须与您在中创建的文件名称匹配。/etc/pam.d.

5. 保存并关闭 文件。

## 配置 NICE DCV 授权

授权用于授予或拒绝 NICE DCV 客户端对特定的 NICE DCV 功能的权限。在 NICE DCV 中，授权是使用权限文件。权限文件定义了特定用户在连接到会话时可使用的特定的 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 \(p. 70\)](#)
- [自定义权限文件 \(p. 70\)](#)
- [使用权限文件 \(p. 70\)](#)

## 默认权限文件

如果在创建会话时未指定自定义权限文件，则为所有会话使用默认权限文件。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 \(p. 70\)](#)。

## 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 \(p. 77\)](#)。

有关创建自定义权限文件的信息，请参阅 [使用权限文件 \(p. 70\)](#)。

## 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 \(p. 71\)](#)
- [创建组 \(p. 71\)](#)
- [创建别名 \(p. 72\)](#)
- [添加权限 \(p. 72\)](#)

## 导入权限文件

`imports` 部分通常是权限文件的第一个部分。您可以使用此部分引用和包含现有权限文件。您还可以将之前定义的 NICE DCV 权限并入您的权限文件中。

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` 语句并使用绝对路径或相对路径指定文件的位置
  - Windows NICE DCV 服务器：

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

- Linux NICE DCV 服务器：

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

示例

以下语句导入一个名为的权限文件 `dcv-permissions.file` 使用绝对路径。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 `[groups]` 部分定义具有类似使用案例或权限要求的用户的用户组。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，您必须先将在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

以下示例添加组部分标题并创建一个名为的组 `my-group`。该组包括个人用户。他们被命名为 `john` 和 `jane`。其中一个现有的群组，名为 `observers`。另一个是名为的操作系统用户组 `guests`：

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

## 创建别名

您可以使用 `[aliases]` 部分以创建 NICE DCV 功能集。在定义别名后，您可以授予或拒绝组或单个用户对该别名的使用权限。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，您必须先将其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。别名成员可以是单个 NICE DCV 功能或其他别名。

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

## 示例

以下示例添加别名部分标题并创建一个名为的别名 `file-management`。它包括 `file-upload` 和 `file-download` 功能和名为的现有别名 `clipboard-management`。

```
[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、要应用的规则和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 `osgroup:`。NICE DCV 包括一个内置的 `%owner%` 引用，该参照可用于引用会话所有者。它也可以用来引用内置的 `%any%` 可用于引用任何用户的引用。

以下规则可用于权限语句：

- `allow`— 授予对功能的访问权。
- `disallow`— 拒绝对功能的访问权，但可被后续权限覆盖。
- `deny`— 拒绝对功能的访问权，但无法被后续权限覆盖。

功能可以包含单个 NICE DCV 功能和/或别名。功能列表必须用空格分隔。NICE DCV 包括一个内置的 `builtin` 别名，其中包括所有 NICE DCV 功能。

可在权限文件中引用以下功能：

- `audio-in`— 将音频插入到 NICE DCV 服务器。
- `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。
- `file-download`— 从会话存储中下载文件。
- `file-upload`— 将文件上传到会话存储。
- `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 及更高版本上受支持。
- `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 版及更高版本上受支持。
- `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 版及更高版本上受支持。
- `printer`— 创建 PDF 或 XPS 文件。
- `screenshot`— 保存远程桌面的屏幕截图。它在 NICE DCV 2021.2 及更高版本上受支持。

移除时 `screenshot` 授权，我们建议您禁用 `clipboard-copy` 权限。这样可以防止用户在服务器的剪贴板上捕获屏幕截图，然后将其粘贴到客户端。当 `screenshot` 授权被拒绝，Windows 和 macOS 还将阻止外部工具捕获客户端的屏幕截图。例如，在 NICE DCV 客户端窗口中使用 Windows 截图工具将生成黑色图像。

- `smartcard`— 从客户端读取智能卡。
- `stylus`— 从专门的 USB 设备 (如 3D 指点设备或绘图板) 输入。
- `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`— 用于在协作会话中设置用户的无所有者访问权限。
- `usb`— 使用来自客户端的 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 及更高版本上受支持。

### 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

您的客户端必须先在 NICE DCV 服务器上创建 NICE DCV 会话，然后才能连接到该会话。如果有活动会话，客户端只能连接到 NICE DCV 服务器。

每个 NICE DCV 会话都具有以下属性：

- 会话ID— 用于标识 NICE DCV 服务器上的特定会话。
- 所有者— 创建了会话的 NICE DCV 用户。默认情况下，只有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 会议简介 \(p. 75\)](#)
- [使用命令行工具管理 NICE DCV 会话 \(p. 76\)](#)
- [启动 NICE DCV 会话 \(p. 77\)](#)
- [停止 NICE DCV 会话 \(p. 81\)](#)
- [管理正在运行的 NICE DCV 会话 \(p. 82\)](#)
- [查看 NICE DCV 会话 \(p. 87\)](#)
- [获取 NICE DCV 会话屏幕截图 \(p. 89\)](#)

## NICE DCV 会议简介

NICE DCV 提供两种类型的会话：控制台会话和虚拟会话。下表总结了两种类型的会话之间的差异。

会话类型	支持	多个会话	所需权限	直接截屏	GPU 加速的 OpenGL 支持
控制台	Linux 和 Windows NICE DCV 服务器	不能，每台服务器上只允许一个控制台会话	只有管理员用户可以启动和关闭会话	是	是的，没有其他软件
虚拟	仅 Linux NICE DCV	是，一台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	不，每个虚拟会话都运行一个专用的 X 服务器 (Xdcv)。屏幕是从 X 服务器捕获的。	是，但需要 DCV-GL 软件包

Note

您不能同时在一个 NICE DCV 服务器上运行控制台和虚拟会话。

## 控制台会话

Windows 和 Linux NICE DCV 服务器上支持控制台会话。如果你使用的是 Windows NICE DCV 服务器，则只能使用控制台会话。

在 NICE DCV 服务器上，一次只能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。

通过控制台会话，NICE DCV 可以直接捕获桌面屏幕的内容。如果服务器配置了 GPU，则 NICE DCV 控制台会话可以直接访问 GPU。

## 虚拟会话

仅 Linux NICE DCV 服务器上支持虚拟会话。

可以在同一个 NICE DCV 服务器上同时托管多个虚拟会话。虚拟会话由 NICE DCV 用户创建和管理。NICE DCV 用户只能管理他们自己创建的会话。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。

通过虚拟会话，NICE DCV 启动一个 X 服务器实例，`xdcv`，并在 X 服务器内运行桌面环境。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

而 NICE DCV 确保每个虚拟会话都有一个独立的 `xdcv` 显示，许多其他系统资源（包括用户主文件夹中的文件、D-Bus 服务和设备）都是按用户进行的，因此可以在同一个用户的多个虚拟会话中共享和访问。

您不应在同一个 NICE DCV 服务器上同时为同一个用户运行多个虚拟会话，除非您已将操作系统设置为缓解对共享资源可能存在的担忧。

如果 `dcv-g1` 包已安装并获得许可，NICE DCV 虚拟会话将共享对服务器 GPU 的访问权限。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-g1.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 服务器包括一个命令行工具，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，则可以从任何目录使用命令。

从 NICE DCV 安装目录使用命令行工具

导航到文件夹 `dcv.exe` 文件位于 `C:\Program Files\NICE\DCV\Server\bin\` 然后打开命令提示窗口。

或者，您可以在从其他目录运行命令时指定完整路径。

```
C:\> "C:\Program Files\NICE\DCV\Server\bin\dcv.exe" list-sessions
```

将 NICE DCV 目录添加到 PATH 环境变量中

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New并指定到binNICE DCV 安装目录中的文件夹 (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使用该工具。打开终端窗口并输入要运行的命令。

## 启动 NICE DCV 会话

当你使用默认值为 [安装 Windows NICE DCV 服务器 \(p. 8\)](#)，[a控制台会话 \(p. 75\)](#) 在安装服务器后自动创建并处于活动状态。默认控制台会话由 Administrator 拥有，其默认会话 ID 为 `console`。你可以使用这个会话或者你可以 [关闭它 \(p. 81\)](#) 并创建新的会话。

在安装 NICE DCV 服务器时，如果您已选择退出自动控制台会话创建，则必须手动创建一个。安装 NICE DCV 服务器后，您可以启用或禁用 [自动控制台会话创建 \(p. 80\)](#) 所需。

### Note

Linux NICE DCV 服务器在安装后不会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，并且您超出了许可证支持的最大并行会话数。您可能得到一个 `no licenses` 错误消息。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以启动会话。有关更多信息，请参阅 [启动 NICE DCV 服务器 \(p. 46\)](#)。

### 主题

- [手动启动控制台和虚拟会话 \(p. 77\)](#)
- [启用自动控制台会话 \(p. 80\)](#)

## 手动启动控制台和虚拟会话

您可以随时启动 NICE DCV 会话。一次只能运行一个控制台会话。如果您使用的是 Linux NICE DCV 服务器，您可以同时运行多个虚拟会话。

跑步是必需的 `dcv list-sessions` 在创建会话之前，尤其是在使用 Windows NICE DCV 服务器的情况下。

要在 Windows 或 Linux NICE DCV 服务器上创建控制台或虚拟会话，请使用 `dcv create-session` 命令。

### 主题

- [语法 \(p. 77\)](#)
- [选项 \(p. 78\)](#)
- [示例 \(p. 80\)](#)

## 语法

启动会话的命令的最低语法为：

```
dcv create-session session_ID
```

包含所有选项的完整语法为：

```
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

这些区域有：\  
symbol 表示将命令分成多行的语法。

您还可以使用 `dcv create-session --help` 以显示语法的快速参考。

## 选项

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。会话名称可以是长度最多为 256 个字符的字符串。如果字符串超过 256 个字符，命令将失败。会话名称不必唯一。

您可以随时使用更改会话名称 `dcv set-name` 命令。有关更多信息，请参阅 [管理会话名称 \(p. 86\)](#)。

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，则将禁用会话存储。

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

以下命令创建拥有的控制台会话由拥有 `dcv-user` 唯一的会话 ID 为 `my-session`，会话名称为 `my graphics session`。它还指定一个名为的权限文件 `perm-file.txt`。

- Windows NICE DCV

```
C:\> dcv create-session^
--owner dcv-user^
--name "my graphics session"^
--permissions-file perm-file.txt^
my-session
```

- Linux NICE DCV

```
$ 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 )

以下命令使用创建虚拟会话 `root` 用户模拟预期的会话所有者，`dcv-user`。会话由 `dcv-user` 拥有，即使它是由根用户创建。

```
$ 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确保每次 NICE DCV 服务器启动时都自动创建一个控制台会话。自动控制台会话由指定的 NICE DCV 用户拥有 `owner` 配置参数。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 \(p. 117\)](#)。

#### Note

NICE DCV 不支持自动创建虚拟会话。

#### 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，打开上下文 (上下单击) 菜单，然后打开会话管理键然后选择 New、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，打开上下文 (上下单击) 菜单，然后打开 `automatic-console-session` 键然后选择 New、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 \(p. 47\)](#)和[重新启动 \(p. 46\)](#)NICE DCV

#### 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 \(p. 47\)](#)和[重新启动 \(p. 46\)](#)NICE DCV

## 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。

要在 Windows 或 Linux NICE DCV 服务器上停止控制台或虚拟会话，请使用 `dcv close-session` 命令并指定唯一的会话 ID。

#### 主题

- [语法 \(p. 82\)](#)
- [示例 \(p. 82\)](#)

## 语法

```
dcv close-session session-id
```

## 示例

例如，以下命令可停止具有唯一 ID 的会话my-session。

```
dcv close-session my-session
```

# 管理正在运行的 NICE DCV 会话

以下部分提供了有关如何管理正在运行的 NICE DCV 会话的信息。

### 主题

- [管理 NICE DCV 会话存储 \(p. 82\)](#)
- [管理 NICE DCV 会话授权 \(p. 83\)](#)
- [管理 NICE DCV 会话显示布局 \(p. 84\)](#)
- [管理会话名称 \(p. 86\)](#)

## 管理 NICE DCV 会话存储

会话存储是 NICE DCV 服务器上的一个目录，客户端在连接到 NICE DCV 会话时可以访问该目录。

如果在 NICE DCV 服务器上启用了会话存储，则可以使用 `dcv set-storage-root` 命令指定服务器上要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 \(p. 60\)](#)。

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 \(p. 82\)](#)
- [选项 \(p. 82\)](#)
- [示例 \(p. 83\)](#)

## 语法

```
dcv set-storage-root --session session_id /path_to/directory
```

对于目录路径，可以使用 `%home%` 指定当前已登录的用户的主目录。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/`。

## 选项

以下选项可以与 `dcv set-storage-root` 命令

### **--session**

要为其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:\> dcw set-storage-root --session my-session c:\session-storage
```

Linux NICE DCV

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)。

```
$ dcw set-storage-root --session my-session %home%/session-storage/
```

## 管理 NICE DCV 会话授权

授权用于授予或拒绝 NICE DCV 客户端对特定 NICE DCV 功能的权限。通常，在 NICE DCV 会话启动时配置授权。但是，可以编辑正在运行的会话的权限。有关 NICE DCV 授权的更多信息，请参阅[配置 NICE DCV 授权 \(p. 70\)](#)。

要修改正在运行的会话的权限，请使用 `dcw set-permissions` 命令。

主题

- [语法 \(p. 83\)](#)
- [选项 \(p. 83\)](#)
- [示例 \(p. 84\)](#)

## 语法

```
dcw set-permissions --session session-id --none | --reset-builtin | --file /path_to/permissions_file
```

您必须指定以下任一 `--none`、`--reset-builtin`，或者 `--file`。

## 选项

以下选项可以与 `dcw set-permissions` 命令一起使用。

`--session`

指定要为其设置权限的会话的 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，则撤销所有权限。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 \(p. 70\)](#)。

## 示例

示例 1 — 撤销所有权限

以下示例撤销 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

以下示例指定了一个名为的自定义权限文件`perm-file.txt`对于 ID 为的会话`my-session`. 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

您可以设置正在运行的 NICE DCV 会话的显示布局。显示布局指定在客户端连接到会话时使用的默认配置。但是，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来手动覆盖布局。

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，则 NICE DCV 服务器不会应用指定的显示布局。

主题

- [限制显示布局 \(p. 84\)](#)
- [指定显示布局 \(p. 85\)](#)
- [查看显示布局 \(p. 86\)](#)

## 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 \(p. 106\)](#)要防止客户端更改显示布局，请将此参数设置为`false`。
- [min-head-resolution \(p. 107\)](#)和[max-head-resolution \(p. 107\)](#)分别指定允许的最小分辨率和最大分辨率。
- [web-client-max-head-resolution \(p. 108\)](#)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。指定较高的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的规格。
- [max-num-heads \(p. 107\)](#)— 指定最大显示数。

- [max-layout-area \(p. 107\)](#)— 指定屏幕区域允许的最大像素数。屏幕区域总大小 (以像素为单位表示) 超过指定值的请求将被忽略。

有关这些参数的更多信息, 请参阅“参数参考”中的[display 参数 \(p. 105\)](#)。

## 指定显示布局

为正在运行的 NICE DCV 会话配置显示布局

使用 `dcv set-display-layout` 命令并指定要为其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
```

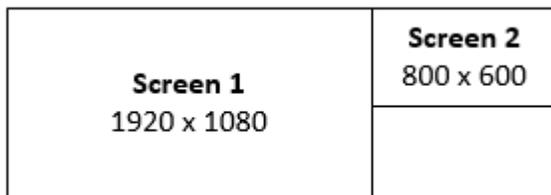
如果指定多个屏幕, 请用逗号将屏幕描述符分隔开来。屏幕位置偏移指定屏幕左上角相对于屏幕 1 的位置。如果没有为屏幕指定位置偏移, 则它默认为  $x=0$  和  $y=0$ 。

### 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:

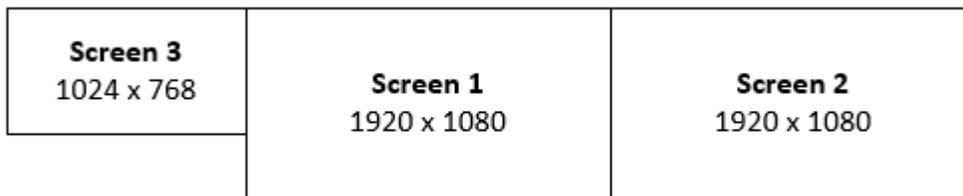
- 屏幕 1: 1920x1080 分辨率, 偏移为  $x=0, y=0$
- 屏幕 2: 800x600 分辨率, 偏移为  $x=1920, y=0$ , 以便它显示在屏幕 1 的右侧。



```
1920x1080+0+0,800x600+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。



```
1920x1080+0+0,1920x1080+1920+0,1024x768-1024+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 \(p. 87\)](#)。

## 管理会话名称

您可以随时更改正在运行的会话的名称。您可以使用会话的特定名称根据会话的名称快速识别会话。会话名称不必唯一。

要更改正在运行的会话的名称，请使用 `dcv set-name` 命令。

主题

- [语法 \(p. 86\)](#)
- [选项 \(p. 86\)](#)
- [示例 \(p. 82\)](#)

## 语法

```
$ dcv set-name --session session_id --none | --name "session-name"
```

您必须指定 `--name` 或 `--none`。

## 选项

以下选项可以与 `dset-name` 命令一起使用。

### **--session**

要为其设置名称的会话的 ID。

类型: 字符串

必需 是

### **--name**

分配会话的名称。只有在要为会话分配名称时才指定此选项。如果要删除名称，请忽略该参数。会话名称最多可以有 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**

指定此参数可从会话中删除现有名称。如果您不想删除会话名称，请忽略该选项。

必需 否

## 示例

示例 1-更改会话名称

以下示例设置 ID 为的会话的名称my-session到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 ID 为的会话的名称my-session.

```
$ dcv set-name --session my-session --none
```

## 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的所有活动会话。NICE DCV 用户只能查看他们自己创建的会话。

主题

- [列出所有活动会话 \(p. 87\)](#)
- [查看有关特定会话的信息 \(p. 87\)](#)

## 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 \(p. 87\)](#)
- [输出 \(p. 87\)](#)

## 语法

```
dcv list-sessions
```

## 输出

该命令使用以下格式返回活动会话的列表。

```
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 \(p. 88\)](#)
- [输出 \(p. 87\)](#)

## 语法

```
$ dcv describe-session session_id
```

## 输出

在以下示例输出中，`display-layout`元素表示会话的显示布局设置为使用两个 800x600 屏幕。其中，第二个屏幕相对于第一个屏幕的偏移为 `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
    }
  ]
}
```

```
{
  "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` 命令获取正在运行的会话的桌面的屏幕截图。

### 语法

```
dcv get-screenshot --max-width pixels --max-height pixels --format JPEG|PNG --primary --
json --output /path_to/destination session_name
```

### 选项

#### **--max-width**

指定屏幕截图的最大宽度（以像素为单位）。如果您未指定宽度或高度，则屏幕截图将使用会话的显示分辨率。如果仅指定高度，则会自动缩放宽度以保持纵横比。

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）。如果不指定宽度或高度，屏幕截图将使用会话的显示分辨率。如果仅指定宽度，则会自动缩放高度以保持纵横比。

类型: 整数

必需 否

#### **--format**

屏幕截图的文件格式。目前仅需 JPEG 和 PNG 支持格式。如果您指定了冲突的文件类型 `--format` 和 `--output` 选项，为指定的值 `--format` 优先级。例如，如果您指定 `--format JPEG` 和 `--output myfile.png`，NICE DCV 会创建一个 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的屏幕截图。要仅获取主显示的屏幕截图，请指定 `--primary`。要获取所有显示的屏幕截图，请忽略该选项。如果您选择获取所有显示屏的屏幕截图，则所有显示将合并为一个屏幕截图。

必需 否

### **--json, -j**

指示是否以 base64 编码的 JSON 格式交付输出。要获取 JSON 输出，请指定 `--json`。否则，请省略它。

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，请指定 `c:\directory\filename.format`，对于 Linux，请指定 `/directory/filename.format`。格式必须为 `.png` 要么 `.jpeg`。如果您指定了冲突的文件类型 `--format` 和 `--output` 选项，为指定的值 `--format` 优先级。例如，如果您指定 `--format JPEG` 和 `--output myfile.png`，NICE DCV 会创建一个 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的会话的屏幕截图 `my-session`。屏幕截图使用服务器的分辨率。

```
dcv get-screenshot --output myscreenshot.png my-session
```

### 示例 2

以下示例命令截取的屏幕截图是 200 像素宽 100 像素高。它从一个名为的会话中获取它 `my-session`。它使用文件名将屏幕截图保存在当前目录中 `myscreenshot.png`。

```
dcv get-screenshot --max-width 200 --max-height 100 --output myscreenshot.png my-session
```

### 示例 3

以下示例命令截取的名为的会话的屏幕截图 `my-session`。屏幕截图仅是主显示屏的屏幕截图。它将文件保存到当前目录中，并命名屏幕截图 `myscreenshot.png`。

```
dcv get-screenshot --primary --output myscreenshot.jpeg my-session
```

### 示例 4。

以下示例命令获取名为的会话的屏幕截图 `my-session`。该命令输出以 base64 和 JSON 格式编码的文件。

```
dcv get-screenshot --json --format png my-session
```

## 如何...

### 主题

- [使用外部验证 \(p. 91\)](#)
- [查找并停止空闲会话 \(p. 93\)](#)
- [启用到 X 服务器的远程 X 连接 \(p. 94\)](#)
- [在 iFrame 中嵌入 NICE DCV Web 浏览器客户端 \(p. 95\)](#)

## 使用外部验证

默认情况下，NICE DCV 客户端身份验证工作委派给底层操作系统。使用 Windows NICE DCV 服务器，身份验证工作委派给 WinLogon。使用 Linux NICE DCV 服务器，身份验证工作委派给 Linux PAM。

您可以配置 NICE DCV 使用外部身份验证服务器对客户端进行身份验证。这使您能够使用现有的身份验证系统。利用外部身份验证，NICE DCV 利用您现有的登录机制并将身份验证委托给外部身份验证服务器。

要使用外部身份验证服务器，您必须具有以下准备：

- **登录机制**— 这是用户用来登录的前端机制。它应该能够使用现有的凭据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 \(p. 92\)](#)。
- **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。此服务器应该能够从 NICE DCV 服务器接收 HTTP (S) POST 请求，其中包括令牌，执行必要的身份验证，然后将响应发送回 NICE DCV 服务器。有关实施身份验证服务器的更多信息，请参阅 [服务要求 \(p. 92\)](#)。
- **配置 NICE DCV 服务器**— 必须将 NICE DCV 服务器配置为使用外部身份验证服务器。有关更多信息，请参阅 [配置 NICE DCV 服务器 \(p. 91\)](#)。

### 主题

- [配置 NICE DCV 服务器 \(p. 91\)](#)
- [使用令牌 \(p. 92\)](#)
- [服务要求 \(p. 92\)](#)

## 配置 NICE DCV 服务器

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

要在 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 `auth-token-verifier` 中的参数 `[security]` 部分，然后用外部身份验证服务器的 URL 和要通信的端口替换现有值，格式如下：`url:port`。例如，如果您使用的是 `dcvSimpleExternal` 身份验证器，则指定以下内容：`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 注册表编辑器并导航到HKEY\_USERS/S-1-5-18/Software/GSettings/com/nictware/dcV/安全/键。
2. 打开auth-token-verifier参数。适用于Value data中，输入外部身份验证服务器的 URL 和要通信的端口，格式如下：**url:port**。例如，如果您使用的是 dcvSimpleExternal 身份验证器，则指定以下内容：**http://127.0.0.1:8444**。

#### Note

如果该参数不存在，则创建一个新的 String 参数并将其命名。auth-token-verifier。

3. 关闭 Windows 注册表编辑器。
4. [停止 \(p. 47\)](#)和[重新启动 \(p. 46\)](#)NICE DCV 服务器。

## 使用令牌

生成令牌后，必须能够将其发送到 NICE DCV 服务器。使用 Web 浏览器客户端，将令牌附加到连接 URL，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
```

## 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。

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。根据身份验证过程的结果，响应正文必须包含以下内容之一：

- 如果身份验证成功，则身份验证服务会返回yes和用户标识符。例如：

```
<auth result="yes"><username>username</username></auth>
```

- 如果身份验证不成功，身份验证服务将返回no。例如：

```
<auth result="no"/>
```

## dcvSimple外部身份验证器

NICE DCV 附带一个名为的参考外部身份验证服务器，`DcvSimpleExternalAuthenticator`。DCV `SimpleExternal` 身份验证器是一个单一的 Python 脚本，您可以用作自行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，并且必须在安装 NICE DCV 服务器的同一服务器上运行。默认情况下，`dcvSimpleExternal` 身份验证器侦听端口上的请求 8444。如果需要，您可以更改端口。要做到这一点，请打开 `/etc/dcv/simpleextauth.conf` 使用首选文本编辑器，找到 `EXTAUTH_PORT` 参数，并将现有值替换为所需的端口号。

要使用 `dcvSimpleExternal` 身份验证器，您必须安装 `nice-dcv-simple-external-authenticator` 程序包。有关更多信息，请参阅 [安装 NICE DCV 服务器 \(p. 19\)](#)。

启动 `DcvSimpleExternalAuthenticator` 安装完成后，请运行以下命令：

```
$ dcvsimpleextauth start-server
```

## 查找并停止空闲会话

您 NICE DCV 以使用 `dcv describe-sessions` 使用 CLI 命令 `-j` 命令选项。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` (p. 81) 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，由于内在安全风险，NICE DCV 2017 会阻止使用 X 转发。NICE DCV 从较新的 Xorg 服务器版本继承了这一行为。NICE DCV 服务器实施了以下默认继承的缓解措施来最大限度降低安全风险：

- NICE DCV 服务器会阻止来自网络的 X 连接。NICE DCV 服务器配置为开始 `-nolisten tcp` 命令行选项。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](#) (p. 94)。
- 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，NICE DCV 配置为从 `-nolisten tcp` 降低暴露在安全风险下的命令行选项。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. Connect NICE DCV 会话。
2. 确认 NICE DCV 服务器正在 6000-6063 范围内的端口上侦听。

```
$ 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 会话显示号码。

```
$ 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. 从远程服务器上，导出显示环境变量以指向 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（本例中为 xterm）应出现在 NICE DCV 服务器的桌面环境中。

## 在 iFrame 中嵌入 NICE DCV Web 浏览器客户端

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 iFrame 中。但是，您可以覆盖此默认行为，以允许 Web 浏览器客户端在 iFrame 内运行。

有关防止点击劫持攻击的更多信息，请参阅[备注单：内容安全策略](#)。

要允许 Web 浏览器在 iFrame 内运行，必须配置 NICE DCV 服务器以向 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

我们建议您添加两个标头文件，以确保跨 Web 浏览器的最佳兼容性。

### Windows server

1. 打开 Windows 注册表编辑器并导航到 HKEY\_USERS/S-1-5-18/Software/GSettings/com/nictware/dcV/连接性/键。
2. 打开 web x 框架选项参数。适用于 Value data，输入 "ALLOW-FROM https://*server\_hostname*".

#### Note

如果该参数不存在，请创建一个新的 String 参数并命名它 web-x-frame-options。

3. 打开 web-extra-http-headers 参数。适用于 Value data，输入 ["Content-Security-Policy", "frame-ancestors https://*server\_hostname*"]。

#### Note

如果该参数不存在，请创建一个新的 String 参数并命名它 web-extra-http-headers。

4. 关闭 Windows 注册表编辑器。
5. [停止 \(p. 47\)](#)和[重新启动 \(p. 46\)](#)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 \(p. 47\)](#)和[重新启动 \(p. 46\)](#)NICE DCV 服务器。

默认情况下，大多数浏览器都禁止访问某些功能，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，请按如下方式修改 iFrame 元素：

```
<iframe src="..." allow="microphone; fullscreen">/iframe>
```

# 解决 NICE DCV

本章说明如何标识和解决您可能遇到的与 NICE DCV 相关的问题。

## 主题

- [使用日志文件 \(p. 97\)](#)
- [在 Linux 上创建虚拟会话故障排除 \(p. 98\)](#)
- [UID 更改后 Linux 会话无法启动 \(p. 100\)](#)
- [修复 Windows 上的光标问题 \(p. 100\)](#)
- [修复复制并粘贴到 IntelliJ IDEA \(p. 100\)](#)

如果您是 NICE DCV 本地客户并且需要其他帮助，请联系您的 NICE DCV 经销商以获得第一级的支持。如果您在 Amazon EC2 上使用 NICE DCV，则可以使用 Amazon 支持。如果您没有 Amazon 支持计划，您可以在 NICE DCV 社区寻求帮助，方法是在 [Amazon 论坛](#)。

## 使用日志文件

NICE DCV 日志文件可用于标识和解决 NICE DCV 服务器的问题。NICE DCV 日志文件可以在 NICE DCV 服务器上的以下位置找到：

- 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 服务器使您能够配置详细程度等级的日志文件。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。
- warning— 包括错误和警告。
- info— 默认详细程度等级。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、warning、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、warning、info 或 debug。

```
[log]
level="verbosity_level"
```

3. 保存并关闭文件。

## 在 Linux 上创建虚拟会话故障排除

主题

- [调查 Linux 上的虚拟会话创建失败 \(p. 98\)](#)
- [在 Linux 上创建故障安全虚拟会话 \(p. 99\)](#)

如果连接到虚拟会话会导致 No session available 要么 The sessionId `session` is not available 错误，这可能是由于虚拟会话创建失败并已终止。

你可以检查会话是否存在 dcv list-sessions 命令。请参阅 [the section called “查看会话” \(p. 87\)](#) 有关检查正在运行的会话的更多信息。如果该会话不在列表中，那么它可能已经失败。

## 调查 Linux 上的虚拟会话创建失败

虚拟会话是 [创建 \(p. 77\)](#) 在 Linux 上使用命令：

```
$ dcv create-session session
```

只有在创建会话失败时，此命令才会返回错误。但是，可能发生会话最初成功创建的情况，但会在用户连接之前终止。你可能会注意到这一点是因为当你检查现有会话时，例如使用命令 dcv list-sessions 或者 dcv describe-session `session`，你可能没有收到列出的会话。

在大多数情况下，发生这种情况的原因是桌面会话已创建但随后立即失败，例如，由 init 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 包含与创建新会话进程的 dcv 组件相关的日志的文件。
- 检查 /var/log/dcv/dcv-session.`user.session`.log 包含与 dcv init 脚本相关的日志的文件。
- 检查 \$HOME/.xsession-errors 文件位于与会话所有者对应的主目录中。此文件包含由系统 X 会话初始化脚本生成的日志，通常包含桌面会话管理器或脚本调用的其他应用程序生成的日志。

- 查看系统日志以获取有关故障系统和组件的更多信息。作为一开始，请检查的输出 `dmesg` (例如在流程失败的情况下) 和 `journalctl -xe`。
- 使用故障安全会话进行测试 (p. 99) 验证问题不取决于正在使用的会话管理器。

如果失败只发生在特定用户身上，您也可以尝试以下操作：

- 检查用户配置，特别是删除或重命名用户配置时会发生什么情况。

根据桌面环境和版本，配置目录可能是 `.gnome` 要么 `.kde` 要么 `.config` 在用户目录中。

- 检查影响用户的特定用户配置 `PATH` 或者环境。很多时候，特定用户的会话启动失败是由于以下框架造成的：`anaconda` 覆盖一些可能导致的标准本机命令 `dbus` 会话初始化中的连接将失败。
- 检查权限问题。在本地设置了错误的权限 `~/.dbus` 要么 `~/.Xauthority` (例如，它们可能是由 `root` 而不是用户) 可能会导致桌面会话立即终止。

## 在 Linux 上创建故障安全虚拟会话

验证会话创建失败是否与桌面环境的启动有关的常见策略是创建最小的会话。我们将本次会议称为“故障安全”会话。如果创建故障安全会话正常工作，那么我们可以推断出你的正常会话因为默认的系统桌面环境无法启动而失败。相反，如果故障安全会话也失败，那么问题更可能与 NICE DCV 服务器的设置有关。

故障安全会话通常由只包含简单窗口管理器和终端的桌面会话组成。这允许用户检查是否存在与使用中的特定会话环境 (通常是 `gnome` 或 `KDE`) 相关的会话创建问题。

为了创建故障安全会话，你需要为用户创建一个 `init` 脚本，其中包含以下内容：

```
#!/bin/sh
metacity &
xterm
```

这将启动 `metacity` 窗口管理器然后启动 `xterm` 终端，只要 `xterm` 进程已终止会话也将终止。

如果系统上可用，您可以使用自己选择的其他会话管理器或终端。

### Note

你必须确保脚本不会立即终止。为此，你需要在脚本末尾启动一个非立即终止的程序。随着最后一个命令终止 (`xterm` 在示例中)，初始会话也会终止。同时，当你在 Windows 管理器之后启动另一个工具时，你需要确保它在后台运行 (通过添加 `&` 在示例中)，以确保调用下一个命令。

然后，您需要确保 `init` 脚本可执行：

```
$ chmod a+x init.sh
```

要从用户 shell 使用指定 `init` 脚本创建会话，请运行此命令，其中 `init.sh` 之前创建的脚本是：

```
$ dcv create-session dummy --init init.sh
```

要为另一个用户作为超级用户创建会话，可以改为运行以下命令：

```
$ sudo dcv create-session test --user user --owner user --init init.sh
```

最后，你可以启动测试应用程序，例如 `dcv gltest` (只有在你有 `nice-dcv-gltest` 已安装软件包) 或 `glxgears` 验证 OpenGL 或任何其他应用程序是否正常工作。

## UID 更改后 Linux 会话无法启动

在 Linux 主机上，更改用户的用户 ID (UID) 或使用修改用户 UID 的其他 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，使用新的 UID 运行的 DCV 会话的进程无权访问仍保留以前 UID 的文件和文件夹。具体而言：

- 这些区域有：[日志文件 \(p. 97\)](#)在 NICE DCV 日志目录中
- 用户的主文件夹

该问题影响控制台和虚拟会话。

要解决此问题，请确保用户的主文件夹及其包含的文件具有正确的 UID 并删除旧的 UID。[NICE DCV 日志文件 \(p. 97\)](#)那里有以前的 UID。

## 修复 Windows 上的光标问题

如果在 Windows Server 2012 或 Windows 8 及更高版本上运行 NICE DCV 服务器，则会始终显示为箭头。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

当尝试将 macOS NICE DCV 客户端中的文本复制到 IntelliJ IDEA 时，无法粘贴文本。IntelliJ 不能接受 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便将文本粘贴到 IntelliJ 中，请修改 `disabled-targets` 在 NICE DCV 服务器上的字段。

此更改将防止复制和粘贴与 NICE DCV Web 客户端一起使用。在进行此更改之前，请确保您希望复制并粘贴 IntelliJ IDEA 仅在 NICE DCV 客户端上工作。

将服务器配置为将文本粘贴到 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` 要么 `[clipboard]` 部分，请手动添加它们。
3. 添加以下内容以定义 `disabled-targets`。

```
[clipboard]
disabled-targets = ['dcv/text', 'JAVA_DATAFLAVOR:application/x-java-jvm-local-objectref; class=com.intellij.codeInsight.editorActions.FoldingData']
```

4. 保存并关闭文件。
5. [停止 \(p. 81\)](#)和[重新启动 \(p. 77\)](#)NICE DCV 会话。

# NICE DCV 服务器参数参考

下表列出了可配置用于自定义 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`在服务器启动时将加载一次该参数。如果更新参数值，则在重新启动服务器时将加载新值。
- `session`在创建会话时将加载该参数。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将加载该参数。如果更新参数值，则新值将用于后续客户端连接。
- `custom`参数的加载对于此参数是唯一的。有关更多信息，请参阅参数说明。

## 主题

- [audio 参数 \(p. 101\)](#)
- [clipboard 参数 \(p. 102\)](#)
- [connectivity 参数 \(p. 103\)](#)
- [display 参数 \(p. 105\)](#)
- [display/linux 参数 \(p. 108\)](#)
- [input 参数 \(p. 109\)](#)
- [license 参数 \(p. 109\)](#)
- [log 参数 \(p. 110\)](#)
- [security 参数 \(p. 111\)](#)
- [session-management 参数 \(p. 115\)](#)
- [session-management/automatic-console-session 参数 \(p. 117\)](#)
- [session-management/defaults 参数 \(p. 118\)](#)
- [smartcard 参数 \(p. 118\)](#)
- [webcam 参数 \(p. 118\)](#)
- [windows 参数 \(p. 119\)](#)
- [修改配置参数 \(p. 119\)](#)

## audio 参数

下表描述了代码示例目录的配置参数[`audio`]的部分`/etc/dcv/dcv.conf`Linux NICE DCV 服务器的文件，以及`audioWindows` NICE DCV 服务器的注册表项。

参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
<code>avsync-支持</code>	字符串	<code>session</code>	' <code>auto</code> '	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“已启用”、“已禁用”或“ <code>auto</code> ”（默认值=' <code>auto</code> '）。如果指定了“ <code>auto</code> ”，则仅在控制台会话中启用音频/视

参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
				频同步，并且只有在加速视频压缩可用时才会启用。—since 版本 2021.1-10557 (p. 131).
source-channels	inice-32 位 ( 32 位 )	session	2	在 Linux 上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所支持的通道数。允许的值为：2 ( 立体声 )、4 ( 4.0 四声道立体声 )、6 ( 5.1 环绕立体声 )、6 ( 5.1 环绕立体声 )。默认值为 2 ( 立体声 ) —since 版本 2020.0-84 (p. 134).

## clipboard 参数

下表描述了代码示例目录的配置参数[clipboard]的部分/etc/dcv/dcv.confLinux NICE DCV 服务器的文件，以及clipboardWindows NICE DCV 服务器的注册表项。

参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
enabled	true or false — DWORD ( 32 位 )	session	Linux的： trseWindows : 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处理。剪贴板监控也将被禁用。—since 版本 2020.0-8428 (p. 134).
max-image-area	inice-32 位 ( 32 位 )	session	-1	剪贴板图像的最大面积— 指定可从服务器传输到客户端的剪贴板图像的最大面积 ( 像素数 )。如果此值缺失或设置为 -1，则无限制。—since 版本 2017.0-4334 (p. 140).
max-payload-size	inice-32 位 ( 32 位 )	session	20971520	剪贴板数据的最大大小— 指定可从服务器传输到客户端的剪贴板数据的最大大小 ( 以字节为单位 )。如果缺少此值，则默认限制为 20 MB。—since 版本 2017.0-4334 (p. 140).
max-text-len	inice-32 位 ( 32 位 )	session	-1	剪贴板文本的最大字符数— 指定可从服务器传输到客户端的剪贴板文本的最多字符数。如果此值缺失或设置为 -1，则无限制。—since 版本 2017.0-4334 (p. 140).

参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
primary-selection-copy	true or false — DWORD (32 位)	session	Linux的 : seWindows : 0	启用来自 linux 的主选副本— Linux 桌面支持多个剪贴板：通用剪贴板和主选区域。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。—since 版本 2019.0-7318 (p. 135).
primary-selection-paste	true or false — DWORD (32 位)	session	Linux的 : seWindows : 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域。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。—since 版本 2019.0-7318 (p. 135).
更新超时	incc-32 位 (32 位)	session	200	更新事件通知超时— 指定从上次更新事件开始等待向客户端发送通知的时间（以毫秒为单位）。默认值 200 毫秒。—since 版本 2020.1-8942 (p. 133).

## connectivity 参数

下表描述了代码示例目录的配置参数[connectivity]的部分/etc/dcv/dcv.confLinux NICE DCV 服务器的文件，以及connectivityWindows NICE DCV 服务器的注册表键。

参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
enable-quic-frontend	true or false — DWORD (32 位)	server	Linux的 : seWindows : 0	是否启用 QUIC 前端— 指定是否应启用 QUIC 前端。—since 版本 2020.2-9508 (p. 132).
idle-timeout	incc-32 位 (32 位)	自定义	60	空闲超时— 指定在断开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。—since 版本 2017.0-4100 (p. 140).
idle-timeout-warning	incc-32 位 (32 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相对于空闲超时的秒数。指定 0 表示从不向空



参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
web-port	inice-32 位 ( 32 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。请参阅 'web-listen-endpoints' 设置，了解有关如何应用此设置的更多详细信息。—since 版本 2017.0-4100 (p. 140).
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。—since 版本 2017.0-4100 (p. 140).
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。—since 版本 2017.0-4100 (p. 140).
web-use-hsts	true or false — DWORD ( 32 位 )	server	Linux 的 : trseWindows : 1	是否使用 HSTS— 启用此项可强制浏览器阻止将通过 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。—since 版本 2017.0-4100 (p. 140).
ws-keepalive-interval	inice-32 位 ( 32 位 )	server	10	Websocket 保活间隔— 指定发送 keepalive 消息的时间间隔 ( 以秒为单位 )。如果设置为 0，则禁用 keepalive 消息。—since 版本 2017.0-4100 (p. 140).

## display 参数

下表描述了代码示例目录的配置参数 [display] 的部分 /etc/dcv/dcv.conf Linux NICE DCV 服务器的文件，以及 displayWindows NICE DCV 服务器的注册表键。

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	session	[]	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

参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
				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 <800><600><0><0><1024><768><800><0> : s 本2017.0-5600 (p. 139).
cuda-devices	字符串	连接	[]	用于流编码的 CUDA 设备— 指定 DCV 用于分发编码和 CUDA 工作负载的本地 CUDA 设备的列表。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。—since 版本2017.2-6182 (p. 138).
enable-client-resize	true or false — DWORD ( 32 位 )	session	Linux的 : trseWindows : 1	是否允许客户设置显示布局— 指定是否允许客户设置显示布局。—since 版本2017.0-4100 (p. 140).
enable-qu	true or ford ( 32 位 )	session	Linux的 : trseWindows : 1	是否发送质量更新— 指定是否发送质量更新。—s 版本2017.0-4100 (p. 140).
enable-yuv444 编码	字符串	session	“default-off”	是否启用 YUV444 编码— 启用或禁用 YUV444 编码。如果“始终在线”, 则服务器将更喜欢为实现高色彩精度而优化的 YUV444 格式。如果“始终关闭”, 则服务器将首选针对流媒体性能进行了优化的格式。“default-on” 和 “default-off” 的语义相同, 即让客户端决定。允许的值 : always-on、always-off、default-on、default-off。—s 版本2022.0-11954 (p. 127).
grabber-target-fps	ince-32 位 ( 32 位 )	session	0	帧抓取器的每秒目标帧数— 设置每秒抓取帧数的上限。值为 0 默认为每个特定帧缓冲区读取器的标准行为, 例如回退到 target-fps 或不限制抓取。并非所有帧捕获后端都支持此设置。—s 版本2017.1-570 (p. 139).

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	inice-32 位 ( 32 位 )	session	4	最大压缩线程— 指定最多压缩程序线程数。—s版本2017.0-4100 (p. 140).
max-head-resolution	字符串	自定义	(4096, 2160)	最大头分辨率— 设置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。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。—s版本2017.0-4100 (p. 140).
max-layout-area	inice-32 位 ( 32 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面积 ( 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。—s版本2019.1-74 (p. 135).
max-num-heads	inice-32 位 ( 32 位 )	自定义	4	最大磁头数— 指定客户端可请求的最大显示头的最大数量。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。—s版本2017.0-4100 (p. 140).
min-head-resolution	字符串	自定义	(640, 480)	Min Head 分— 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。—s版本2017.0-4100 (p. 140).
target-fps	inice-32 位 ( 32 位 )	session	-1	每秒目标帧数— 指定允许的每秒最多允许的帧数。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。如果版本小于 2020.2，则无法识别 -1 值，默认值为 25。—s版本2017.0-4100 (p. 140).

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 or ford ( 32 位 )	session	Linux的： trseWindows : 1	是否使用脏区— 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。—s版本2017.0-4100 (p. 140).
web-client-max-head-se	字符串	自定义	(1920, 1080)	Web 客户端的最大头部分辨率— 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。这些区域有：max-head-resolution 限制选项应用于此选项设置的最大宽度和最大高度值。如果此值设置为 (0, 0)，则忽略该选项。—s版本2020.0-84 (p. 134).

## display/linux 参数

下表描述了代码示例目录的配置参数[display/linux]的部分/etc/dcv/dcv.confLinux NICE DCV 服务器的文件，以及display/linuxWindows NICE DCV 服务器的注册表项。

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	session	[:0.0]	3D 加速 X 显示屏— 指定 DCV 用于虚拟会话中 OpenGL 渲染的本地 3D 加速 X 显示器和屏幕的列表。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。—s版本2017.0-4100 (p. 140).
h264-encoder-displays	字符串	连接	[]	H.264 编码器 X 显示屏— 指定支持加速 H.264 编码的本地 X 显示器和屏幕的列表。如果为空，则 DCV 使用为 OpenGL 渲染选定的同一显示器。仅当系统上安装的某些 GPU 不使用受支持的技术之一为 H.264 编码提供加速时，此设置很有用。—s版本2017.0-4100 (p. 140).

## input 参数

下表描述了代码示例目录的配置参数[`input`]的部分`/etc/dcv/dcv.conf`Linux NICE DCV 服务器的文件，以及`inputWindows` NICE DCV 服务器的注册表项。

参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
<code>enable-autorepeat</code>	true or false ( 32 位 )	session	Linux的： <code>trseWindows : 1</code>	是否允许在 Linux 上自动重复— 指定是否允许自动重复单个键。—s版本2017.2-61 (p. 138).
<code>enable-gampad</code>	true or false — DWORD ( 32 位 )	session	Linux的： <code>trseWindows : 1</code>	是否允许游戏手柄输入— 指定是否启用游戏手柄。—s版本2022.0-11954 (p. 127).
<code>enable-relative- mouse</code>	true or false — DWORD ( 32 位 )	session	Linux的： <code>trseWindows : 1</code>	是否允许相对鼠标移动— 指定是否允许鼠标的相对移动。—s版本2017.0-51 (p. 140).
<code>enable-stylus</code>	true or false — DWORD ( 32 位 )	session	Linux的： <code>trseWindows : 1</code>	是否允许手写笔输入— 指定是否启用触控笔。—s版本2019.0-741 (p. 135).
<code>enable-touch</code>	true or false — DWORD ( 32 位 )	session	Linux的： <code>trseWindows : 1</code>	是否允许触摸输入— 指定是否启用触摸。—s版本2017.3-68 (p. 137).

## license 参数

下表描述了代码示例目录的配置参数[`license`]的部分`/etc/dcv/dcv.conf`Linux NICE DCV 服务器的文件，以及`licenseWindows` NICE DCV 服务器的注册表项。

参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
<code>license-file</code>	字符串	session	"	许可证— 指定试用许可证文件或 RLM 服务器端口和主机名。如果在 RLM 服务器上使用的是浮动许可证，请使用此参数以 <code>port@hostname</code> 格式指定 RLM 服务器的端口和主机名。如果您使用的是延期试用许可证，并且没有在默认位置放置 <code>license.lic</code> 文件，请使用此参数指定 <code>license.lic</code> 许可证文件的完整路径。如果该默认文件不存在，则使用试用许

参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
				可证。从配置中读取此值，并在每次创建新会话时更新。—s版本2017.0-4100 (p. 140).

## log 参数

下表描述了代码示例目录的配置参数[log]的部分/etc/dcv/dcv.confLinux NICE DCV 服务器的文件，以及logWindows NICE DCV 服务器的注册表项。

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—指定保存日志的目标位置。如果未指定，则默认为“C:\ProgramData在Linux上)。—since版本2017.0-4100 (p. 140).
level	字符串	自定义	“info”	日志级别—指定日志文件的详细程度级别。详细程度有（按提供的信息量顺序排列）：“错误”、“警告”、“信息”和“调试”。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。—since版本2017.0-4100 (p. 140).
max-file-size	ince-32位（32位）	server	0	日志文件大小的最大值 MegaBytes 轮换—指定触发轮换之前的最大日志文件大小。如果值为'0'，则禁用按大小旋转，而是在重新启动生成文件的进程时轮换文件。—since版本2022.1-13067 (p. 125).
rotate	ince-32位（32位）	server	10	日志文件轮换次数—指定日志文件在被删除前分割的次數。如果值为0，则删除而不分割旧版本。—since版本2017.0-4100 (p. 140).
轮换间隔	字符串	server	“none”	连续两次日志文件轮换之间的最大时间间隔—指定连续两次日志文件轮换之间的最大时间间隔。如果该值为“none”，则文件不会根据时间旋转。其他可能的值包括“每分钟”、“每

参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
				小时”和“每天”。—since 版本 2022.1-13067 (p. 125).
轮换	字符串	server	'计数器'	要附加到轮换日志文件中的后缀—指定要附加到轮换日志文件的后缀。如果指定了“counter”，则会在每个轮换的日志文件中附加一个简单的递增计数器后缀。如果指定了“时间戳”，则会对日志文件应用形式为“YYYY-MM-DD-HH-MM”的时间戳。如果日志文件夹中已存在带有该时间戳的轮换文件，则会在时间戳后附加一个额外的数字计数器。—since 版本 2022.1-13067 (p. 125).
transfer-audit	字符串	server	“none”	将方向移交给审计—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。容许值为：“无”、“server-to-client”、“client-to-server”和“全部”。如果此值缺失或等于“none”，则禁用传输审核且不创建文件。—since 版本 2017.0-4100 (p. 140).

## security 参数

下表描述了代码示例目录的配置参数[security]的部分/etc/dcv/dcv.conf Linux NICE DCV 服务器的文件，以及security Windows NICE DCV 服务器的注册表项。

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—指定表示此 DCV 服务器可以服务的正则表达式模式。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。例如：'^ (www \.)? 示例 \.com\$'。—since 版本 2017.0-4100 (p. 140).

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://.+ \$"	允许的起源—指定表示此 DCV 服务器接受的正则表达式模式。在建立 WebSocket connection，客户端的握手时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。这是防止跨站点的一种安全手段 WebSocket 劫持 (CSWSH) 攻击。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> ]。例如：'^ https://(www\.)? (\.\.)? 示例 \.com (: 443)? \$'。—since 版本 2017.0-4100 (p. 140).
auth-connection- setup-timeout	inice-32 位 ( 32 位 )	server	120	身份验证通道连接设置超时—指定在超时之前允许完成身份验证通道连接设置过程所用的时间 (以秒为单位)。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。—since 版本 2017.0-4100 (p. 140).
auth-token-verifier	字符串	server	"	身份验证令牌验证器的终端节点—指定 DCV 服务器使用的身份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。—since 版本 2017.0-4100 (p. 140).
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。—since 版本 2017.0-4100 (p. 140).
authentication- threshold	inice-32 位 ( 32 位 )	server	3	身份验证阈值—指定在服务器关闭连接之前，每个客户端的身份验证可以失败的次数。要允许无限制的身份验证尝试，请使用 0。—since 版本 2017.0-4100 (p. 140).

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。—since 版本 2017.0-4100 (p. 140).
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。—since 版本 2022.0-11954 (p. 127).
ciphers	字符串	server	"ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-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。—since 版本 2017.0-4100 (p. 140).
connection-estab-timeout	inice-32 位 ( 32 位 )	server	5	连接建立超时— 指定在超时之前允许完成连接过程所用的时间 ( 以秒为单位 ) 。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。—since 版本 2017.0-4100 (p. 140).
connection-setup-timeout	inice-32 位 ( 32 位 )	server	5	频道连接设置超时— 指定在超时之前允许完成通道连接设置过程所用的时间 ( 以秒为单位 ) 。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。—since 版本 2017.0-4100 (p. 140).
CRL false	字符串	自定义	"	CRL 文件— 指定包含证书吊销列表 (CRL)。—since 版本 2022.0-11954 (p. 127).
enable-gssapi	true or false — DWORD ( 32 位 )	server	Linux 的 : seWindows : 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。—since 版本 2017.3-68 (p. 137).

参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
max-connections- per-user	inice-32 位 ( 32 位 )	server	10	用户的最大连接数— 指定 每个用户所允许的并发连 接的最多数量。超出的连 接将被拒绝。—since版 本2017.0-4100 (p. 140).
no-tls-strict	true or false — DWORD ( 32 位 )	server	Linux的 : seWindows : 0	启用或禁用严格的证书验证— 在 连接到外部身份验证令牌验证 程序时, 启用或禁用严格的证书 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。—since版 本2017.0-4100 (p. 140).
os-auto-lock	true or false — DWORD ( 32 位 )	session	Linux的 : trseWindows : 1	是否在上次客户端连接结束 时锁定操作系统会话— 如果 启用此项, 则在关闭最后 一个客户端连接时, 操作系统 会话将被锁定。—since版 本2017.1-577 (p. 139).
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。—since版 本2017.0-4100 (p. 140).
passwd-file	字符串	server	”	密码文件— 指定用于检查 用户证书的密码文件 ( 仅限 于 dcv 身份验证模式 )。 如果为空, 请对 Linux 使用 \${XDG_CONFIG_HOME}/ NICE/dcv/passwd 中的默认 文件, 或对 Windows 使用 %CSIDL_LOCAL_APPDATA %NICE\dcv\passwd 中 的默认文件。—since版 本2017.0-4100 (p. 140).
server-fqdn	字符串	server	”	SQDN— 指定服务器完 全限定域名。空白表示 gethostname()。—since版 本2017.3-698 (p. 137).
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 )。—since版 本2020.0-8428 (p. 134).

参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
监督控制	字符串	自定义	'已禁用'	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“禁用”和“强制执行”。如果将此值设置为“强制”，则可以将非监督访问权限设置为允许或拒绝协作会话中用户的无所有者访问。如果允许用户进行非监督访问，则该用户可以在没有所有者的情况下访问会话。默认情况下，除所有者之外的所有用户都被拒绝此权限。当此值设置为“disabled”（默认）时，服务器不会强制执行此监督控制和权限。新值在配置上进行更改后立即生效。—since版本2021.3-11591 (p. 128).
user-realm	字符串	server	"	服务器用户领域— 指定服务器的用户领域。—since版本2017.3-698 (p. 137).

## session-management 参数

下表描述了代码示例目录的配置参数[session-management]的部分/etc/dcv/dcv.confLinux NICE DCV 服务器的文件，以及session-managementWindows NICE DCV 服务器的注册表项。

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 or false — DWORD ( 32 位 )	server	Linux的： seWindows : 0	在服务器启动时创建控制台会话— 指定在服务器启动时是否自动创建控制台会话（使用 ID “控制台”）。—since版本2017.0-4100 (p. 140).
enable-gl-in-virtual-会话	字符串	session	“default-on”	是否使用 dcv-gl 功能— 指定是否使用 dcvs 功能（需要许可证）。允许的值：always-on、always-off、default-on、default-off。—since版本2017.0-4100 (p. 140).
max-concurrent-clients	incc-32 位 ( 32 位 )	session	-1	每个会话最多并发客户端的最大数量— 指定每个会话最多并发客户端的最大数量。如果设置为 -1，则表示无限限制。要仅为自动会话设置限制，请使用max-concurrent-clients'会话管理/automatic-

参数	类型- Windows 注册表类 型	重新加载 上下文	默认值	描述
				console-session'。—since版本2017.0-4100 (p. 140).
max-concurrent-sessions	inice-32位 ( 32位 )	server	0	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。—since版本2019.0-7318 (p. 135).
max-sessions-per-user	inice-32位 ( 32位 )	server	0	每个用户的最大会话数— 指定每个用户可以拥有的允许的最大并发会话数。此限制当前仅适用于虚拟会话。指定 0 表示不强制施加任何限制。—since版本2021.0-10242 (p. 131).
virtual-session-default-layout	字符串	session	[]	虚拟会话的默认布局— 如果设置此项，则将 Xdcv 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 <800><600><0><0><1024><768> : [<800><0>[] virtual-session-monitors setting) 的优先级高于数组中的元素的数量。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，则表示已启用监视器数 ( 在 virtual-session-monitors 设置 ) 将被忽略。—since版本2017.0-5600 (p. 139).
virtual-session-font-path	字符串	session	"	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。—since版本2017.0-4100 (p. 140).
virtual-session-source-profile	true or false — DWORD ( 32位 )	session	Linux的 : seWindows : 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 是否应提供用户配置文件。默认情况下这是假的，DCV 将使用 "bash —noprofile —norc" 运行会话启动器脚本 —since版本2021.3-11591 (p. 128).
virtual-session-xdcv-args	字符串	session	"	要传递给 Xdcv 的额外参数— 指定要传递到 Xdcv 的任何其他参数。—since版本2017.0-4334 (p. 140).

## session-management/automatic-console-session 参数

下表描述了代码示例目录的配置参数[session-management/automatic-console-session]的部分/etc/dcv/dcv.confLinux NICE DCV 服务器的文件，以及session-management/automatic-console-sessionWindows NICE DCV 服务器的注册表项。

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'	指定在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连接。允许的值包括'reject-new-connection' (传入的连接将被关闭) 和'same-user-oldest-connection' (服务器将关闭与会话交互时间最长的同一个用户的连接，或者如果没有该信息，则关闭连接时间最早的同一个用户的连接)。—since版本2022.1-13067 (p. 125).
max-concurrent-clients	inice-32位 (32位)	server	-1	每个会话最多并发客户端的最大数量— 指定每个会话所允许的并发客户端的最大数量。如果设置为 -1，则表示无限制。—since版本2017.0-5600 (p. 139).
owner	字符串	server	"	自动创建的“控制台”会话的所有者— 指定“控制台”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。—since版本2017.0-5600 (p. 139).
permissions-file	字符串	server	"	自动“控制台”会话的权限文件— 指定用于检查用户的 DCV 功能访问权限文件的路径。如果为空，则只有所有者才对会话拥有完全访问权限。—since版本2017.0-5600 (p. 139).
storage-root	字符串	server	"	文件存储根文件夹的路径— 指定要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。—since版本2017.0-5600 (p. 139).

## session-management/defaults 参数

下表描述了代码示例目录的配置参数[session-management/defaults]的部分/etc/dcv/dcv.confLinux NICE DCV 服务器的文件，以及session-management/defaultsWindows NICE DCV 服务器的注册表项。

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	session	"	所有会话中都包含默认权限—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\Server\conf）。—since版本2017.0-5600 (p. 139).

## smartcard 参数

下表描述了代码示例目录的配置参数[smartcard]的部分/etc/dcv/dcv.confLinux NICE DCV 服务器的文件，以及smartcardWindows NICE DCV 服务器的注册表项。

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。—since版本2017.2-6182 (p. 138).

## webcam 参数

下表描述了代码示例目录的配置参数[webcam]的部分/etc/dcv/dcv.confLinux NICE DCV 服务器的文件，以及webcamWindows NICE DCV 服务器的注册表项。

参数	类型- Windows 注册表类型	重新加载 上下文	默认值	描述
max-repth	字符串	连接	(1280, 720)	网络摄像头最大分辨率— 指定向应用程序公开的最大网络摄像机分辨率。—since版本2021.0-10242 (p. 131).
首选分辨率	字符串	连接	(640, 480)	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。如果指定的值之一为 0，则禁用网络摄像头共享。—since版本2021.0-10242 (p. 131).

## windows 参数

下表描述了代码示例目录的配置参数[windows]的 部分/etc/dcv/dcv.confLinux NICE DCV 服务器的文件，以及windowsWindows NICE DCV 服务器的注册表键。

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 or false — DWORD ( 32 位 )	session	Linux的 : trseWindows : 1	防止显示器进入省电模式— 指定是否防止显示器进入节能模式。—since版本2017.0-4100 (p. 140).
printer	字符串	session	“DCV printer”	要设置为默认打印机— 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，DCV 将不会更改当前的默认打印机。—since版本2017.0-4100 (p. 140).

## 修改配置参数

本节介绍如何修改 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 \(p. 101\)](#)。

主题

- [Windows NICE DCV 服务器服务器服务器 \(p. 120\)](#)
- [Linux NICE DCV 服务器 \(p. 120\)](#)

## Windows NICE DCV 服务器服务器服务器

对于 Windows NICE DCV 服务器，请使用 Windows 注册表编辑器修改配置参数。PowerShell，或命令行。

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 \(p. 101\)](#) 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 \(p. 101\)](#) 中所述的类型和名称添加该参数。

使用修改配置参数 PowerShell

1. 运行 PowerShell 作为管理员。
2. 使用 [NICE DCV 服务器参数参考 \(p. 101\)](#) 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 \(p. 101\)](#) 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 \(p. 101\)](#) 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 DCV 服务器，可使用文本编辑器或命令行工具修改配置参数，例如 `crudini`。

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 \(p. 101\)](#) 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 \(p. 101\)](#)中所述的参数名称添加该参数。

```
parameter_name="parameter_value"
```

4. 保存并关闭 文件。

使用 `crudini` 修改配置参数

使用 [NICE DCV 服务器参数参考 \(p. 101\)](#)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 终止支持寿命

NICE DCV 终止 Support 寿命 (EOSL) 定义了在此时间点之后，NICE DCV 的特定主要版本（及其所有次要版本）将不再获得支持，并且不再经过与较新版本的兼容性测试。

在 EOSL 日期之前，NICE DCV 支持团队将继续为配置问题提供全面支持。缺陷解决方案和功能请求仅针对最新版本的 NICE DCV 服务器和 NICE DCV 客户端实施。它们不适用于旧版本。

在 EOSL 日期之后，不提供进一步的支持或维护。我们还将停止测试兼容性问题。要获得持续支持，您必须升级到最新的 NICE DCV 版本。

主题

- [EOSL 时间线 \(p. 122\)](#)
- [面向客户的 EOSL 路径 \(p. 122\)](#)
- [EOSL 常见问题 \(p. 123\)](#)

## EOSL 时间线

下表显示了 NICE DCV 主要版本的 EOSL 时间表。

NICE DCV 主要版本	初始发布日期	EOSL 日期
NICE DCV 2016.x	2015 年 12 月 31 日	2021 年 3 月 31 日
NICE DCV	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 2021 年	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

## 面向客户的 EOSL 路径

如果你正在运行 NICE DCV Amazon，你不需要 NICE DCV 的许可证。您只需为底层证券付费 Amazon 用于工作负载的资源。如果您当前使用的是超过 EOSL 日期的 NICE DCV 版本，请使用 [下载页面](#) 或者利用 [不错的 DCV AMI](#) 来自的 Amazon Web Services Marketplace 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，且您当前使用的 NICE DCV 版本超过了 EOSL 日期，请联系您的经销商或分销商以评估可用的升级路径。如果你有有效的支持合同，你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 [网站 NICE](#)。

## EOSL 常见问题

1. 我使用的是 NICE DCV 版本，该版本已在本地或与第三方云服务提供商达到了 EOSL，但我现有的支持合同。我会受到 EOSL 的影响吗？

如果您有有效的支持合同，那么 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，无需额外付费。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完整支持：

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，这为您提供了升级到 NICE DCV 最新版本的途径。
3. 通过支付恢复费恢复旧的支持合同，这相当于自您的支持合同到期以来一段时间内支持服务当前收费的 70%。

2. 我使用的是在 Amazon EC2 上达到了 EOSL 的 NICE DCV 版本，我该怎么办才能升级到受支持的版本？

升级到完全支持的 NICE DCV 版本以便在 Amazon EC2 上使用，可随时向客户提供，无需额外费用。

3. 我可以已经到达其 EOSL 且支持的 NICE DCV 服务器的 NICE DCV 客户端版本，反之亦然？

是的，但我们强烈建议您将客户端和服务器软件升级到最新版本，因为错误修复不再适用于已达到 EOSL 的版本。

# NICE DCV 的发布说明和文档历史记录

## 录

本页面提供了NICE DCV 的发布说明和文档历史记录。

### 主题

- [NICE DCV 发行说明 \(p. 124\)](#)
- [文档历史记录 \(p. 141\)](#)

## NICE DCV 发行说明

本节概述了 NICE DCV 的主要更新、功能版本和错误修复。所有更新均按发布数据进行组织。我们会经常更新文档来处理您发送给我们的反馈意见。

### 主题

- [DCV 2022.1-13300— 2022 年 8 月 4 日 \(p. 125\)](#)
- [DCV 2022.1-13216— 2022 年 7 月 21 日 \(p. 125\)](#)
- [DCV 2022.1-13067— 2022 年 6 月 29 日 \(p. 125\)](#)
- [DCV 2022.0-12760— 2022 年 5 月 23 日 \(p. 126\)](#)
- [DCV 2022.0-12627— 2022 年 5 月 19 日 \(p. 126\)](#)
- [DCV 2022.0-12123— 2022 年 3 月 23 日 \(p. 127\)](#)
- [DCV 2022.0-11954 — 2022 年 2 月 23 日 \(p. 127\)](#)
- [DCV 2021.3-11591— 2021 年 12 月 20 日 \(p. 128\)](#)
- [DCV 2021.2-11445— 2021 年 11 月 18 日 \(p. 128\)](#)
- [DCV 2021.2-11190— 2021 年 10 月 11 日 \(p. 129\)](#)
- [DCV 2021.2-11135— 2021 年 9 月 24 日 \(p. 129\)](#)
- [DCV 2021.2-11048— 2021 年 9 月 1 日 \(p. 129\)](#)
- [DCV 2021.1-10851— 2021 年 7 月 30 日 \(p. 130\)](#)
- [DCV 2021.1-10598— 2021 年 6 月 10 日 \(p. 131\)](#)
- [DCV 2021.1-10557— 2021 年 5 月 31 日 \(p. 131\)](#)
- [DCV 2021.0-10242 — 2021 年 4 月 12 日 \(p. 131\)](#)
- [DCV 2020.2-9662— 2020 年 12 月 4 日 \(p. 132\)](#)
- [DCV 2020.2-9508— 2020 年 11 月 11 日 \(p. 132\)](#)
- [DCV 2020.1-9012— 2020 年 9 月 30 日 \(p. 133\)](#)
- [DCV 2020.1-9012— 2020 年 8 月 24 日 \(p. 133\)](#)
- [DCV 2020.1-8942— 2020 年 8 月 3 日 \(p. 133\)](#)
- [DCV 2020.0-8428 \(p. 134\)](#)
- [DCV 2019.1-744 \(p. 135\)](#)
- [DCV 2019.1-7423 2019 年 10 月 10 日 9 月 10 日 9 月 10 日 \(p. 135\)](#)
- [DCV 2019.0-718 — 2019 年 8 月 5 日 5 日 5 日 5 日 \(p. 135\)](#)
- [DCV 2017.4-698 \(p. 136\)](#)
- [DCV 2017.3-698 \(p. 137\)](#)
- [DCV 2017.2-6182 \(p. 138\)](#)

- DCV 2017.1-5870 (p. 139)
- DCV 2017.1-5777 (p. 139)
- DCV 2017.0-5600 — 2018 年 6 月 4 日 (p. 139)
- DCV 2017.0-5121 2018 (p. 140)
- DCV 2017.0-434 (p. 140)
- DCV 2017.0-4100 2017 (p. 140)

## 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"> <li>• nice-dcv-server: 13300</li> <li>• nice-dcv-client(Windows): 8261</li> <li>• nice-dcv-viewer (macOS) 4279</li> <li>• nice-dcv-viewer (Linux) 4251</li> <li>• nice-xdcv : 433</li> <li>• nice-dcv-gl : 973</li> <li>• nice-dcv-gltest : 295</li> <li>• nice-dcv-simple-externalauthenticatata : 193</li> </ul>	<ul style="list-style-type: none"> <li>• 当有多个协作者连接到会话时，请勿自动解锁 Windows。</li> <li>• 修复了服务器无法加载指定的证书文件时的问题。</li> <li>• 修复了导致 macOS 客户端音频失真的问题。</li> </ul>

## 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"> <li>• nice-dcv-server: 13216</li> <li>• nice-dcv-client(Windows): 8261</li> <li>• nice-dcv-viewer (macOS) 4251</li> <li>• nice-dcv-viewer (Linux) 4251</li> <li>• nice-xdcv : 433</li> <li>• nice-dcv-gl : 966</li> <li>• nice-dcv-gltest : 295</li> <li>• nice-dcv-simple-externalauthenticatata : 193</li> </ul>	<ul style="list-style-type: none"> <li>• 修复了所有客户端中导致无法连接到 NICE DCV 服务器 2019.1 及更早版本的问题。</li> <li>• 修复了一个问题 SmartCard 在 Windows 服务器上进行重定向。</li> <li>• 修复了在具有 GPU 的主机上连接到 NICE DCV 服务器时可能导致流式传输失败的问题。</li> </ul>

## 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"> <li>• nice-dcv-server: 13067</li> <li>• nice-dcv-client(Windows) 8248</li> </ul>	<p>NICE DCV 添加了以下功能：</p> <ul style="list-style-type: none"> <li>• 为服务器添加了对 Ubuntu 22.04 和 Rocky Linux 8.5 及更高版本的支持。</li> <li>• 为本机客户端添加了对 Ubuntu 22.04 的支持。</li> <li>• 改进了 Windows、macOS 和 Linux 本机客户端的协作体验。</li> </ul>	<ul style="list-style-type: none"> <li>• 提高了性能，使非 GPU 服务器上的整体 CPU 消耗最多减少了 30%。</li> <li>• 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。</li> <li>• 修复了 QUIC 传输中可能导致初始握手失败的问题。</li> </ul>

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li>nice-dcv-viewer (macOS) 4241</li> <li>nice-dcv-viewer (Linux) 4241</li> <li>nice-xdcv : 433</li> <li>nice-dcv-gl : 966</li> <li>nice-dcv-gltest : 295</li> <li>nice-dcv-simple-externalauthenticatataate : 193</li> </ul>		<ul style="list-style-type: none"> <li>修复了可能导致某些应用程序在 Linux 服务器上的相对鼠标移动无法按预期工作的问题。</li> </ul>

## 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"> <li>nice-dcv-server: 12760</li> <li>nice-dcv-client(Windows): 8145</li> <li>nice-dcv-viewer (macOS): 4131</li> <li>nice-dcv-viewer (Linux): 4131</li> <li>nice-xdcv : 424</li> <li>nice-dcv-gl : 961</li> <li>nice-dcv-gltest : 291</li> <li>nice-dcv-simple-externalauthenticatataate : 188</li> </ul>	<p>变化:</p> <p>修复了在指定 web-url-path 选项。</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"> <li>nice-dcv-server: 12627</li> <li>nice-dcv-client(Windows): 8145</li> <li>nice-dcv-viewer (macOS): 4131</li> <li>nice-dcv-viewer (Linux): 4131</li> </ul>	<p>变化:</p> <ul style="list-style-type: none"> <li>修复了 QUIC 传输中的一些问题，这些问题可能导致带宽估计不正确和视觉伪影。</li> </ul>

内部版本号	更改和错误修复
<ul style="list-style-type: none"> <li>nice-xdcv : 424</li> <li>nice-dcv-gl : 961</li> <li>nice-dcv-gltest : 291</li> <li>nice-dcv-simple-externalauthenticatataatate : 188</li> </ul>	<ul style="list-style-type: none"> <li>修复了 Windows 服务器安装程序中的音频服务可能导致更新过程失败的问题。</li> <li>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。</li> <li>修复了在 macOS 和 Linux 客户端中保存屏幕截图时出现的问题。</li> <li>更新了 OpenSSL、zlib 和 gdk-pixbuf 第三方库。</li> </ul>

## 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"> <li>nice-dcv-server: 12123</li> <li>nice-dcv-client(Windows): 7920</li> <li>nice-dcv-viewer (macOS): 3973</li> <li>nice-dcv-viewer (Linux): 3973</li> <li>nice-xdcv : 424</li> <li>nice-dcv-gl : 961</li> <li>nice-dcv-gltest : 291</li> <li>nice-dcv-simple-external-身份验证器 : 188</li> </ul>	<p>NICE DCV 添加了以下功能：</p> <ul style="list-style-type: none"> <li>添加了在 macOS 和 Linux 客户端上启用高色彩精度的选项。</li> </ul>	<p>中的变化：</p> <ul style="list-style-type: none"> <li>改进了使用 QUIC 传输时的带宽估计和图像质量。</li> </ul> <p>修复：</p> <ul style="list-style-type: none"> <li>修复了使用 NVIDIA 驱动程序 510.xx 时在 Linux 上控制台会话中的视觉伪影。</li> <li>修复了问题 DualShock 在 Windows 本机客户端中通过蓝牙连接的 4 个控制器。</li> <li>修复了启用网络摄像头时 macOS 客户端可能崩溃的问题。</li> </ul>

## 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"> <li>nice-dcv-server: 11954</li> <li>nice-dcv-client(Windows): 7866</li> <li>nice-dcv-viewer (macOS) 3929</li> <li>nice-dcv-viewer (Linux) 3929</li> <li>nice-xdcv : 424</li> <li>nice-dcv-gl : 961</li> <li>nice-dcv-gltest : 291</li> <li>nice-dcv-simple-externalauthenticatataatate : 188</li> </ul>	<p>NICE DCV 添加了以下功能：</p> <ul style="list-style-type: none"> <li>游戏控制器支持 Windows 服务器和 Windows 本机客户端。</li> <li>NICE DCV Web 客户端现在可以利用 WebCodecs 在支持它的浏览器上。</li> <li>添加了在 Windows 和 Web 客户端上启用高色彩精度的选项。</li> <li>改进的协作体验：当有人加入会话时，用户会收到通知</li> <li>将 CentOS 8 Stream 添加到支持的 Linux 发行版列表中。</li> </ul>	<p>中的变化：</p> <ul style="list-style-type: none"> <li>现在，无需重新启动 NICE DCV 服务器即可更新 TLS 证书。</li> <li>现在可以将 NICE DCV 服务器配置为监听特定的网络接口或特定的 IPv4 或 IPv6 地址。</li> <li>“DCV 打印机”现在也可以在 Linux 系统上自动配置。</li> <li>Windows 上的 NICE DCV 进程现在以更高的优先级执行。</li> </ul> <p>修复：</p>

内部版本号	新功能	更改和错误修复
		<ul style="list-style-type: none"> <li>修复了在 Windows 2016 上使用带有 GPU 的实例时代理重启时崩溃的问题。</li> <li>修复了从 NICE DCV 客户端重定向某些 USB 设备时注销会话时在 Windows 上崩溃的问题。</li> <li>执行授权检查时，规范包含 Windows 域的用户名。</li> <li>改进了 Windows 客户端中的相对鼠标模式。</li> <li>修复了同步的问题 CapsLock 键。</li> </ul>

## 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"> <li>nice-dcv-server: 11591</li> <li>nice-dcv-client(Windows): 7801</li> <li>nice-dcv-viewer (macOS) 3829</li> <li>nice-dcv-viewer (Linux) 3829</li> <li>nice-xdcv : 415</li> <li>nice-dcv-gl : 952</li> <li>nice-dcv-gltest : 284</li> <li>nice-dcv-simple-external-身份验证器 : 176</li> </ul>	<p>NICE DCV 添加了以下功能：</p> <ul style="list-style-type: none"> <li>Web 客户端的用户界面已更新。</li> <li>现在支持 EC2 G5 和 G5G 实例。</li> <li>Windows Server 2022 和 Windows 11 现在是受支持的操作系统。</li> </ul>	<ul style="list-style-type: none"> <li>Linux 虚拟会话的 init 脚本不再加载用户的 bash 配置文件，从而避免了环境变量覆盖系统默认值时反复出现的问题。</li> <li>这些区域有：nice-dcv-external-authenticator 现在需要 Python 3。</li> </ul>

## 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"> <li>nice-dcv-server: 11445</li> <li>nice-dcv-client(Windows ) : 779999999</li> <li>nice-dcv-viewer macOS ) : 379999999</li> <li>nice-dcv-viewer Linux) : 3799999</li> <li>nic-xdcv : 411</li> <li>nice-dcv-gl: 946</li> <li>nice-dcv-gltest: 279</li> <li>nice-dcv-simple-externalauthenticatataata : 160</li> </ul>	<p>修复：</p> <ul style="list-style-type: none"> <li>修复了阻止客户端在 macOS Monterey 上正常工作的问题。</li> <li>改进了 Windows 上服务器的安全性。</li> <li>修复了可能导致无法正确应用多显示器布局的错误，尤其是在使用 Web 客户端时。</li> <li>修复了一个可能导致 Delete 键无法在某些 Windows 应用程序中正常工作。</li> <li>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。</li> </ul>

## 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"><li>nice-dcv-server: 11190</li><li>nice-dcv-client(Windows) : 778888888</li><li>nice-dcv-viewer macOS) : 377777777</li><li>nice-dcv-viewer Linux) : 3777777</li><li>nic-xdcv : 411</li><li>nice-dcv-gl: 946</li><li>nice-dcv-gltest: 279</li><li>nice-dcv-simple-externalauthenticatataate : 160</li></ul>	<p>修复：</p> <ul style="list-style-type: none"><li>修复了 Windows 客户端中的一个问题，该问题阻止用户在连接到具有过期证书的服务器时关闭证书验证对话框。</li><li>修复了触控笔上的中键单击按钮在原生客户端上无法按预期工作的问题。</li><li>修复了 Xdcv 中导致无法加载旧版 X11 字体的回归。</li><li>修复了 macOS 和 Linux 客户端在使用死键的键盘布局时键盘组合无法正常工作的问题。</li></ul>

## 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"><li>nice-dcv-server: 11135</li><li>nice-dcv-client(Windows): 7781</li><li>nice-dcv-viewer (macOS) 3740</li><li>nice-dcv-viewer (Linux) 3740</li><li>nice-xdcv : 408</li><li>nice-dcv-gl : 944</li><li>nice-dcv-gltest : 279</li><li>nice-dcv-simple-external-身份验证器 : 160</li></ul>	<p>修复：</p> <ul style="list-style-type: none"><li>修复了 QUIC 数据包大小协商问题，该问题在使用 2021.2 客户端连接较旧的服务器时可能导致连接和性能问题。</li><li>修复了可能导致 NVENC 编码器失效的 NVENC 设备选择错误。</li><li>修复了装有 Windows 和 NVIDIA GPU 的计算机上可能导致压缩伪影和色彩精度伪影的问题。</li><li>修复了 Linux 服务器上修改键的错误，该错误可能导致某些键盘组合无法按预期工作。</li><li>修复了配备 M1 CPU 的计算机上的 macOS 客户端的性能下降问题。</li><li>修复了 macOS 客户端中的一个错误，该错误会导致某些键盘组合无法按预期工作。</li><li>修复了在 Linux 虚拟会话中如何处理触摸事件的问题，该问题可能导致会话终止。</li></ul>

## 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"><li>nice-dcv-server: 11048</li><li>nice-dcv-client(Windows): 7774</li><li>nice-dcv-viewer (macOS) 3690</li><li>nice-dcv-viewer (Linux) 3690</li><li>nice-xdcv : 406</li></ul>	<p>NICE DCV 添加了以下功能：</p> <ul style="list-style-type: none"><li>Web 客户端剪贴板改进。有了这些改进，你现在可以使用谷歌浏览器和微软 Edge 上的</li></ul>	<p>中的变化：</p> <ul style="list-style-type: none"><li>现在，NICE DCV Web 客户端在 Linux 上是一个单独的软件包，也是 Windows 安装程序中的可选组件。通过此更改，客</li></ul>

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li>nice-dcv-gl : 944</li> <li>nice-dcv-gltest : 279</li> <li>nice-dcv-simple-external-身份验证器 : 160</li> </ul>	<p>NICE DCV 网络客户端复制和粘贴 PNG 格式的图像。</p> <ul style="list-style-type: none"> <li>一个适用于 Windows 和 macOS 客户端截图屏蔽功能。此功能可防止用户截取 NICE DCV 会话内容的屏幕截图，从而增加了一层额外的安全保护。启用后，用户捕获的任何屏幕截图都会显示空白屏幕。</li> <li>流媒体质量改进。直播质量特别通过“更好”得到改善build-to-lossless”使用 QUIC 协议时的性能。</li> <li>一个certificate-validation-policy选项来指定你的客户端的行为。当服务器提供不受信任的 X.509 证书（如自签名证书）时，您可以使用它。</li> <li>可以更改运行时在音频驱动程序中配置的声道数。</li> <li>Pressure2K 选项已添加到dcvinputXorg 模块。您可以使用它来将触控笔的压力灵敏度范围从0-65335更改为0-2048，以便与Mari和Nuke等应用程序兼容</li> <li>Support 实验 WebCodecs 添加了谷歌浏览器和微软 Edge 上的 API。当您在浏览器中启用此 API 时，NICE DCV Web 客户端可以使用它来加速视频解码并提供更高的帧速率。</li> </ul>	<p>用户可以决定是否部署 Web 客户端。</p> <ul style="list-style-type: none"> <li>现在，使用 NVENC 编码器时，支持 H.264 高配置文件。将 NVENC 编码器与 NVIDIA GPU 配合使用，可以减少带宽使用量，同时保持相同的图像质量。</li> <li>NICE DCV 服务器现在使用所有可用的 GPU 在具有多个 GPU 的计算机上进行压缩。</li> <li>NICE DCV 附带的所有 Windows 驱动程序现已通过 WHQL 认证。</li> <li>OpenSSL 已更新为 1.1.1.1.1.1.1.1.1.1.1.1.</li> <li>Xdv 已更新为 xServer 1.13。</li> </ul> <p>修复：</p> <ul style="list-style-type: none"> <li>修复了 macOS 客户端上的数字小键盘键的问题。</li> <li>修复了导致某些 USB 设备（例如游戏手柄）无法正确重定向到 Windows 服务器的问题。</li> <li>修复了断开连接时无法正确释放修改键的错误。</li> <li>修复了使用 Ubuntu 20.04 和英特尔 GPU 时 Linux 本机客户端崩溃的问题。</li> </ul>

## 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"> <li>nice-dcv-server : 10851</li> <li>nice-dcv-client(Windows ) : 774444444</li> <li>nice-dcv-viewer(macOS) 3590</li> <li>nice-dcv-viewer(Linux) 3560</li> <li>nice-xdcv : 392</li> <li>nice-dcv-gl : 937</li> <li>nice-dcv-gltest : 275</li> <li>nice-dcv-simple-external-身份验证器 : 154</li> </ul>	<p>中的变化：</p> <ul style="list-style-type: none"> <li>我们改进了 Windows、Linux 和 macOS 客户端上的稳定性。</li> </ul> <p>修复：</p> <ul style="list-style-type: none"> <li>修复了在 Windows 服务器上使用 AMD 和 NVIDIA 图形适配器导致屏幕闪烁的错误。</li> <li>修复了连接到运行多个会话的 Linux 服务器时偶尔出现的问题。</li> <li>修复了与在 Linux 服务器上处理非西方键盘布局相关的错误。</li> </ul>

内部版本号	更改和错误修复
	<ul style="list-style-type: none"> <li>修复了 Windows 客户端中连接窗口中的视觉伪影。</li> <li>修复了 Windows 上 USB 重定向驱动程序中的几个错误并改进了设备兼容性。</li> </ul>

## 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"> <li>nice-dcv-server: 10598</li> <li>nice-dcv-client(Windows) : 771313</li> <li>nice-dcv-viewer(macOS) : 34777777</li> <li>nice-dcv-viewer(Linux) : 3477777</li> <li>nic-xdcv : 39992</li> <li>nice-dcv-gl: 937</li> <li>nice-dcv-gltest: 275</li> <li>nice-dcv-simple-external-身份验证器 : 154</li> </ul>	<ul style="list-style-type: none"> <li>修复了服务器的 Windows 安装程序中预填充的问题 session owner authenticated-当前用户。</li> <li>提高了 macOS 和 Linux 客户端的整体稳定性。</li> </ul>

## 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"> <li>nice-dcv-server: 10557</li> <li>nice-dcv-client(Windows): 7713</li> <li>nice-dcv-viewer (macOS) 3450</li> <li>nice-dcv-viewer (Linux) 3454</li> <li>nice-xdcv : 392</li> <li>nice-dcv-gl : 937</li> <li>nice-dcv-gltest : 275</li> <li>nice-dcv-simple-externalauthenticatataate : 154</li> </ul>	<ul style="list-style-type: none"> <li>NICE DCV 添加了客户端选项，以便在连接到具有 GPU 的服务器时实现精确的音频/视频同步。</li> <li>NICE DCV 在 Linux 控制台会话中添加了对麦克风的支持。</li> </ul>	<ul style="list-style-type: none"> <li>降低了没有 GPU 的 Windows 服务器主机上的 CPU 使用率。</li> <li>修复了阅读问题 .dcv macOS 和 Linux 客户端中的连接文件。</li> <li>为不支持硬件加速解码的 macOS 计算机添加了软件解码的备用功能。</li> <li>添加了对 macOS 客户端读取存储在系统钥匙串中的 CA 证书的支持。</li> </ul>

## 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"> <li>nice-dcv-server: 10242</li> <li>nice-dcv-client(Windows): 7643</li> <li>nice-dcv-viewer (macOS) 3186</li> <li>nice-dcv-viewer (Linux) 3294</li> <li>nice-xdcv : 380</li> <li>nice-dcv-gl : 912</li> </ul>	<ul style="list-style-type: none"> <li>添加了对 Windows NICE DCV 服务器的网络摄像头重定向支持。</li> <li>添加了对 Linux NICE DCV 服务器的打印机重定向支持。</li> </ul>	<ul style="list-style-type: none"> <li>优化了 Linux 服务器和配备 NVIDIA GPU 的 Amazon EC2 实例上的 GPU 和 CPU 资源使用情况。</li> <li>添加了对在 Linux NICE DCV 服务器的 Amazon EC2 G4aD</li> </ul>

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li>nice-dcv-glttest : 266</li> <li>nice-dcv-simple-external-身份验证器 : 134</li> </ul>	<ul style="list-style-type: none"> <li>在 macOS 客户端上添加了对 M1 处理器的支持。</li> <li>添加了针对 macOS 客户端的多显示器显示支持。</li> </ul>	<ul style="list-style-type: none"> <li>实例上使用 AMD GPU 进行 GPU 加速视频编码的支持。</li> <li>优化音频处理以减少音频延迟</li> <li>如果在服务器上启用了 QUIC 协议，则将客户端的默认值更改为 QUIC 协议。</li> <li>在 DCV 命令行工具中添加了新的获取屏幕截图命令。</li> <li>添加了强制注销选项，该选项使用 <code>--logout-user</code> 的选项 <code>close-session</code> 命令。关闭控制台会话时可以使用此选项。</li> </ul>

## 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"> <li>nice-dcv-server: 9662</li> <li>nice-dcv-client(Windows): 7490</li> <li>nice-dcv-viewer (macOS) 2117</li> <li>nice-dcv-viewer (Linux) 3007</li> <li>nice-xdcv : 359</li> <li>nice-dcv-gl : 881</li> <li>nice-dcv-glttest : 259</li> <li>nice-dcv-simple-externalauthenticatataate : 125</li> </ul>	<ul style="list-style-type: none"> <li>增强了 Web 浏览器客户端中使用的安全协议。</li> <li>提高了与 Windows 客户端配合使用的 Amazon EC2 G4ad 实例的性能和稳健性。</li> <li>修复了在 Windows 客户端的连接设置对话框中选择端口的问题。</li> </ul>

## 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"> <li>nice-dcv-server: 9508</li> <li>nice-dcv-client(Windows): 7459</li> <li>nice-dcv-viewer (macOS) 2078</li> <li>nice-dcv-viewer (Linux) 1737</li> <li>nice-xdcv : 359</li> <li>nice-dcv-gl : 881</li> <li>nice-dcv-glttest : 259</li> <li>nice-dcv-simple-external-身份验证器 : 125</li> </ul>	<ul style="list-style-type: none"> <li>增加了对 QUIC ( 基于 UDP ) 传输协议的支持。</li> <li>增加了对 SLES 15 和 Ubuntu 20.4 的支持。</li> <li>增加了对 Windows NICE DCV 服务器的智能卡支持。</li> </ul>	<ul style="list-style-type: none"> <li>对于托管在具有 NVIDIA GPU 的服务器和 EC2 实例上的控制台会话，将默认的 NICE DCV 帧速率限制器更改为 60 FPS。</li> <li>优化了 Windows NICE DCV 服务器上使用的 GPU 和 CPU 资源，这些服务器托管在具有 NVIDIA GPU 的 EC2 实例上。</li> <li>增加了 <code>list-endpoints</code> NICE DCV CLI 命令。这列出了当前的活动端点。</li> <li>这些区域有：<code>version</code> NICE DCV CLI 命令支持 <code>--json</code> 选项。</li> <li>在 Linux 服务器上，<code>create-session</code> NICE DCV CLI 命令</li> </ul>

内部版本号	新功能	更改和错误修复
		<p>现在支持<code>--disable-login-monitor</code>选项。</p> <ul style="list-style-type: none"> <li>• 改进了与 Linux NICE DCV 服务器上不同显示管理器的兼容性。</li> <li>• 修复了键盘输入处理中的几个问题。</li> <li>• USB 设备允许列表文件现在动态重新加载。</li> </ul>

## 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"> <li>• nice-dcv-server: 9012</li> <li>• nice-dcv-client (Windows): 7342</li> <li>• nice-dcv-viewer (macOS) 1986</li> <li>• nice-dcv-viewer (Linux) 1545</li> <li>• nice-xdcv : 338</li> <li>• nice-dcv-gl : 840</li> <li>• nice-dcv-gltest : 246</li> <li>• nice-dcv-simple-externalauthenticatata : 111</li> </ul>	<ul style="list-style-type: none"> <li>• 添加了缺失的 macOS 客户端图标。</li> </ul>

## 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"> <li>• nice-dcv-server: 9012</li> <li>• nice-dcv-client (Windows): 7342</li> <li>• nice-dcv-viewer (macOS) 1910</li> <li>• nice-dcv-viewer (Linux) 1545</li> <li>• nice-xdcv : 338</li> <li>• nice-dcv-gl : 840</li> <li>• nice-dcv-gltest : 246</li> <li>• nice-dcv-simple-externalauthenticatata : 111</li> </ul>	<ul style="list-style-type: none"> <li>• 修复了 Amazon S3 访问权限 Amazon GovCloud 区域</li> <li>• 基于 Web 的客户端</li> </ul>

## 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"> <li>• nice-dcv-server: 8942</li> <li>• nice-dcv-client (Windows): 7342</li> </ul>	<ul style="list-style-type: none"> <li>• Linux NICE DCV 服务器现在支持 Amazon 基于 Graviton2 的 Arm 实例，例如 m6g、c6g</li> </ul>	<ul style="list-style-type: none"> <li>• 在没有 GPU 的 Amazon EC2 实例上添加了对新 NICE DCV 虚拟显示驱动程序的支持。</li> </ul>

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li>nice-dcv-viewer (macOS) 1910</li> <li>nice-dcv-viewer (Linux) 1545</li> <li>nice-xdcv : 338</li> <li>nice-dcv-gl : 840</li> <li>nice-dcv-gltest : 246</li> <li>nice-dcv-simple-externalauthenticatataate : 111</li> </ul>	<p>和 R6g。有关更多信息，请参阅 <a href="#">Amazon Graviton 处理器</a>。</p> <ul style="list-style-type: none"> <li>在 Linux NICE DCV 服务器上添加了对 RHEL 8.x 和 CentOS 8.x 的支持。</li> <li>添加了使用 Windows NICE DCV 服务器和 Windows NICE DCV 客户端时对打印机重定向的支持。</li> <li>在 macOS 和 Linux 原生 NICE DCV 客户端上添加了对压力敏感的手写笔支持。</li> <li>添加了对 Linux NICE DCV 服务器和 Linux NICE DCV 客户端的环绕声 5.1 支持。</li> <li>添加了对 Linux NICE DCV 本机客户端的触摸屏支持。</li> <li>现在，您可以将自定义名称关联到 NICE DCV 会话。</li> <li>Support 在 macOS 原生 NICE DCV 客户端上进行硬件加速解码和渲染。</li> </ul>	<ul style="list-style-type: none"> <li>解决了使用 NVENC 编码器时由于色彩空间转换而导致视觉伪影的问题。</li> <li>这些区域有：dcv list-sessions 命令现在始终包含控制台会话（如果存在）</li> <li>在较新的 Linux 发行版中，控制台会话代理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。</li> <li>现在，使用以下命令激活 URL 时，本机客户端会自动打开 dcv:// 方案。</li> <li>改进了 macOS 本机客户端和 Web 客户端处理键盘修改器的方式。</li> <li>改进了 DCV-GL 中的可视化和 fbconfig 选择，以改进对某些应用程序的支持。</li> <li>文件传输期间的 CPU 使用率降低</li> <li>改进了 Web 浏览器客户端中的 WebGL 渲染，以减少资源使用。</li> </ul>

## DCV 2020.0-8428

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li>nice-dcv-server : 8428</li> <li>nice-dcv-client (Windows): 7238</li> <li>nice-dcv-viewer (macOS) 1716</li> <li>nice-dcv-viewer (Linux) 1358</li> <li>nice-xdcv : 296</li> <li>nice-dcv-gl : 759</li> <li>nice-dcv-gltest : 229</li> <li>nice-dcv-simple-externalauthenticatataate : 87</li> </ul>	<ul style="list-style-type: none"> <li>增加了 Linux 服务器上的屏幕触控笔和触摸支持。</li> <li>增加了 Windows 服务器到 Windows 本机客户端上的 7.1 环绕立体声播放支持。</li> <li>增加了 Linux 本机客户端上的硬件加速和触控笔支持。</li> <li>增加了一个用于在服务器端设置显示布局。</li> <li>在 Microsoft Edge 浏览器（版本 79.0.309 或更高版本）上添加了多显示器 Web 客户端显示器的多显示器 Web 客户端显示支持。</li> </ul>	<ul style="list-style-type: none"> <li>在全屏模式下，现在可以隐藏 Windows 客户端上的工具栏控制。</li> <li>增加了 Windows 本机客户端上的 NTLM 代理支持。</li> <li>改进了对使用 NVIDIA 适配器的 Windows 无头物理主机的支持。</li> <li>删除了对传统 NVIDIA NvIFR 库的支持。</li> <li>增加了对最新的 Windows 10 上的 Windows 图形捕获 API 的支持。</li> <li>增加了对 Amazon EC2 的支持</li> </ul> <p>EC2 实例上的实例元数据服务 (IMDS) v2。</p> <ul style="list-style-type: none"> <li>DCV CLI 提供了新的 on-client-connected/disconnected</li> </ul>

内部版本号	新功能	更改和错误修复
		<p>命令来检测客户端与会话连接或断开连接的时间。</p> <ul style="list-style-type: none"> <li>增加了对指定主机名以绑定外部身份验证器证书的支持。</li> <li>DCV-GL 现在在支持 GL 厂商中立的分派库 (GLvnd) 的系统上使用它。</li> </ul>

## DCV 2019.1-744

内部版本号	更改和错误修复
<ul style="list-style-type: none"> <li>nice-dcv-server : 7644</li> <li>nice-dcv-client (Windows): 7114</li> <li>nice-dcv-viewer (macOS) 1535</li> <li>nice-dcv-viewer (Linux) 1124</li> <li>nice-xdcv : 226</li> <li>nice-dcv-gl : 544</li> <li>nice-dcv-gltest : 220</li> <li>nice-dcv-simple-externalauthenticatataate : 77</li> </ul>	<ul style="list-style-type: none"> <li>修复了 NICE 使用的集成 API 中的问题 EnginFrame 和其他会话管理器。</li> <li>修复了 32 位版本的 Windows 本机客户端的问题。</li> </ul>

## DCV 2019.1-7423 2019 月 10 日 9 月 10 日 9 月 10 日

内部版本号	更改和错误修复
<ul style="list-style-type: none"> <li>nice-dcv-server : 7423</li> <li>nice-dcv-client (Windows): 7087</li> <li>nice-dcv-viewer (macOS) 1535</li> <li>nice-dcv-viewer (Linux) 1124</li> <li>nice-xdcv : 226</li> <li>nice-dcv-gl : 544</li> <li>nice-dcv-gltest : 220</li> <li>nice-dcv-simple-externalauthenticatataate : 77</li> </ul>	<ul style="list-style-type: none"> <li>改进了 Windows 上的 DCV 服务器的安全性。</li> <li>修复了 Linux 上的 Autodesk Maya 的渲染问题。</li> <li>增加了与键盘操作相关的改进和错误修复。</li> </ul>

## DCV 2019.0-718 — 2019 8 月 5 日 5 日 5 日 5 日

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li>nice-dcv-server : 7318</li> <li>nice-dcv-client (Windows): 7059</li> <li>nice-dcv-viewer (macOS) 1530</li> <li>nice-dcv-viewer (Linux) 968</li> </ul>	<ul style="list-style-type: none"> <li>Web 客户端支持多显示器。</li> <li>Windows Server 2019 上支持触控笔输入。</li> <li>macOS 和 Linux 本机客户端上音频输入/输出。</li> </ul>	<ul style="list-style-type: none"> <li>添加了针对 Windows 触摸输入的压力灵敏度的改进兼容性。</li> <li>改进了 Windows 上具有异构图形适配器的系统的行为。</li> <li>减少了检测非活动连接所需的时间 (例如, 响应客户端上</li> </ul>

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li>nice-xdcv : 224</li> <li>nice-dcv-gl : 529</li> <li>nice-dcv-gltest : 218</li> <li>nice-dcv-simple-externalauthenticatataate : 72</li> </ul>	<ul style="list-style-type: none"> <li>Linux 服务器上的增强剪贴板功能 ( 中键单击粘贴 )。</li> </ul>	<p>从有线网络到 Wi-Fi 网络的更改)。</p> <ul style="list-style-type: none"> <li>减少了无法在 Linux 上捕获光标图标时的日志记录。</li> <li>支持在虚拟会话 Xdcv 组件中禁用复合扩展。</li> <li>增加了一个选项来限制并发虚拟会话数。</li> <li>改进了安装了 Bash 5 的系统的脚本兼容性。</li> <li>更改了 OpenGL 和 GLES 的默认值, 以便自动检测并使用 OpenGL 和 GLES 以在 Linux 客户端上进行渲染。</li> <li>更新了当 GL 窗口的可见性发生变化时的 DCV-GL 屏幕缓冲区。</li> <li>修复了 Windows 7 上的 Windows 客户端中的鼠标滚轮检测。</li> <li>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。</li> <li>改进了横向打印文档时 Windows 客户端上的打印。</li> </ul>

## DCV 2017.4-698

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li>nice-dcv-server : 6898</li> <li>nice-dcv-client (Windows): 6969</li> <li>nice-dcv-viewer (macOS) 1376</li> <li>nice-dcv-viewer (Linux) 804</li> <li>nice-xdcv : 210</li> <li>nice-dcv-gl : 490</li> <li>nice-dcv-gltest : 216</li> <li>nice-dcv-simple-externalauthenticatataate : 70</li> </ul>	<ul style="list-style-type: none"> <li>适用于 macOS 的新本机客户端。</li> </ul>	<ul style="list-style-type: none"> <li>Windows 本机客户端现在使用硬件加速进行解码和渲染 ( 如果它在系统中可用 )。</li> <li>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。</li> <li>dcv 命令行工具现在报告有关许可证的信息。</li> <li>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。</li> <li>改进了对使用多个修饰符的键盘组合的支持。</li> <li>针对通信失败改进了与 Reprise License Manager 的交互的稳健性。</li> <li>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。</li> </ul>

内部版本号	新功能	更改和错误修复
		<ul style="list-style-type: none"> <li>在 Linux 上使用带多个 GPU 的 NVENC 硬件编码器时，遵循 <code>nvidia-smi</code> 工具使用的相同顺序。</li> <li>Linux 客户端现在接收和处理来自 Windows DCV 打印机的打印文件。</li> </ul>

## DCV 2017.3-698

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li><code>nice-dcv-server</code> : 6698</li> <li><code>nice-dcv-client</code> : 5946</li> <li><code>nice-dcv-viewer</code> (Linux) 683</li> <li><code>nice-xdcv</code> : 207</li> <li><code>nice-dcv-gl</code> : 471</li> <li><code>nice-dcv-gltest</code> : 210</li> <li><code>nice-dcv-simple-external-身份验证器</code> : 66</li> </ul>	<ul style="list-style-type: none"> <li>增加了对 Kerberos (GSSAPI) 身份验证的支持。</li> <li>增加了对支持触摸的 Windows 版本上的触摸事件的支持。</li> <li>在使用系统身份验证 (Windows 凭证提供程序) 时自动解锁 Windows 会话。</li> </ul>	<ul style="list-style-type: none"> <li>增加了选择使用 Y'UV444 编码的选项。</li> <li>EL6 RPM 现在包括 NVENC 编码器模块。</li> <li>Windows 系统身份验证现在接受 <code>name@domain</code> 格式。</li> <li>Yubikey USB 设备现已添加到允许列表中。</li> <li>改进了日语键盘支持。</li> <li>输入授权权限更为精细。增加了操作虚拟光标的 <code>pointer</code> 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 <code>keyboard-sas</code> 权限以在 Windows 上操作 SAS (Ctrl+Alt + Del)。 <code>keyboardsas</code> 依赖于 <code>keyboard</code> 权限。</li> <li>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。</li> <li>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。</li> <li>改进了对并发文件存储传输的处理。</li> <li>使用较新的 NVIDIA 驱动程序在 Windows 上修复了 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。</li> <li>永不停止重试重新获取 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。</li> </ul>

内部版本号	新功能	更改和错误修复
		<ul style="list-style-type: none"> <li>• 增加了一个选项来在 Windows 客户端中设置全屏键盘快捷键。</li> <li>• 改进了 Windows 客户端中跨多个显示器拖动窗口时的自适应逻辑。</li> <li>• 修复了 Windows 客户端中未触发断开连接时的 prompt-reconnect 选项。</li> <li>• 修复了 DCV-GL 与 NVIDIA 驱动程序 410.xx 不兼容的问题。</li> <li>• 使用 Matlab 和 Blender 应用程序修复了 DCV-GL 中的性能下降问题。</li> </ul>

## DCV 2017.2-6182

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li>• nice-dcv-server : 6182</li> <li>• nice-dcv-client : 5890</li> <li>• nice-dcv-viewer (Linux) 503</li> <li>• nice-xdcv : 180</li> <li>• nice-dcv-gl : 427</li> <li>• nice-dcv-gltest : 201</li> <li>• nice-dcv-simple-externalauthenticatataatate : 58</li> </ul>	<ul style="list-style-type: none"> <li>• 增加了对 Linux 虚拟会话的音频播放支持。</li> <li>• 改进了智能卡性能。</li> <li>• 增加了 Linux 客户端的文件传输支持。</li> </ul>	<ul style="list-style-type: none"> <li>• 与键盘操作相关的改进和错误修复。</li> <li>• 更改配置中的日志级别不再需要重新启动服务器。</li> <li>• Windows 服务器安装程序现在在跳过 Microsoft C 运行时可再分发组件的安装 ( 如果它已安装 )。</li> <li>• 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。</li> <li>• 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。</li> <li>• 增加了一个 cuda-devices 设置在 display 部分。这可将服务器配置为在不同的 CUDA 设备上分发 NVENC 编码。</li> <li>• 改进了处理多个并发命令时的会话创建代码的可靠性。</li> <li>• 将默认剪贴板限制增加到了 20 MB。</li> <li>• Windows 客户端现在检测旧 .dcv 文件并启动 DCV 2016 Endstation ( 如果已安装 )。</li> <li>• 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。</li> </ul>

内部版本号	新功能	更改和错误修复
		<ul style="list-style-type: none"> <li>• 改进了 DCV-GL 的回读策略，以提高性能和可靠性。</li> <li>• 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。</li> </ul>

## DCV 2017.1-5870

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li>• nice-dcv-server : 5870</li> <li>• nice-dcv-client : 5813</li> <li>• nice-dcv-viewer (Linux) 450</li> <li>• nice-xdcv : 170</li> <li>• nice-dcv-gl : 366</li> <li>• nice-dcv-gltest : 198</li> <li>• nice-dcv-simple-externalauthenticatataate : 53</li> </ul>	<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"> <li>• 现在可在创建会话时读取许可证设置。这使管理员能够更改此设置，而无需重新启动服务器。</li> <li>• 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。</li> <li>• 减少了可能出现的错误情况下的日志记录。</li> </ul>

## DCV 2017.1-5777

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li>• nice-dcv-server : 5777</li> <li>• nice-dcv-client : 5777</li> <li>• nice-dcv-viewer (Linux) 438</li> <li>• nice-xdcv : 166</li> <li>• nice-dcv-gl : 366</li> <li>• nice-dcv-gltest : 189</li> <li>• nice-dcv-simple-externalauthenticatataate : 51</li> </ul>	<ul style="list-style-type: none"> <li>• 增加了 Linux 本机客户端。</li> <li>• 增加了对 3DConnexion 鼠标和 USB 存储设备的支持。</li> <li>• Windows 会话在最后一个客户端断开连接时自动锁定。</li> </ul>	<ul style="list-style-type: none"> <li>• Linux 版本中的性能改进。</li> <li>• 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。</li> <li>• 改进了 Linux 上的智能卡支持。</li> <li>• 修复了使用 Linux 控制台会话时上传文件的文件权限。</li> </ul>

## 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"> <li>• nice-dcv-server : 5600</li> <li>• nice-dcv-client : 5600</li> <li>• nice-xdcv : 160</li> <li>• nice-dcv-gl : 279</li> <li>• nice-dcv-gltest : 184</li> </ul>	<ul style="list-style-type: none"> <li>• 增加了对 Linux 上的多个显示器的支持。</li> <li>• Windows 客户端性能改进。</li> <li>• 在 Chrome 66+ 上使用了新的剪贴板 API。</li> <li>• 增加了适用于 Windows 的 NVENC 编码器。</li> </ul>	<ul style="list-style-type: none"> <li>• EC2 上的使用现在需要能够从运行 DCV 服务器的实例访问 S3。</li> <li>• 针对服务器帧处理和 Windows 客户端解码的性能改进。</li> </ul>

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li>nice-dcv-simple-external-身份验证器 : 48</li> </ul>		<ul style="list-style-type: none"> <li>修复了与以下内容相关的键盘问题 NumPad 和封锁的修饰符。</li> <li>防止在 Linux 上使用外部身份验证器时发生文件描述符泄露。</li> <li>修复了可能出现的智能卡连接错误。</li> </ul>

## DCV 2017.0-5121 2018

内部版本号	新功能	更改和错误修复
<ul style="list-style-type: none"> <li>nice-dcv-server : 5121</li> <li>nice-dcv-client : 5121</li> <li>nice-xdcv : 146</li> <li>nice-dcv-gl : 270</li> <li>nice-dcv-gltest : 184</li> <li>nice-dcv-simple-externalauthenticatataate : 46</li> </ul>	<ul style="list-style-type: none"> <li>Windows 本机客户端现在可以识别 DPI。</li> <li>增加了对相对鼠标移动模式的支持。</li> </ul>	<ul style="list-style-type: none"> <li>阻止了在 Linux 上的 Ansys cfx5solve 挂起。</li> <li>修复了 Windows 10 上可能出现的代理挂起情况。</li> <li>改进了 Web 客户端用户界面。</li> <li>指定域时规范化的 Windows 用户名。</li> <li>修复了 RHEL6 上的外部验证器。</li> </ul>

## DCV 2017.0-434

内部版本号	更改和错误修复
<ul style="list-style-type: none"> <li>nice-dcv-server : 4334</li> <li>nice-dcv-client : 4334</li> <li>nice-xdcv : 137</li> <li>nice-dcv-gl : 254</li> <li>nice-dcv-gltest : 184</li> <li>nice-dcv-simple-externalauthenticatataate : 45</li> </ul>	<ul style="list-style-type: none"> <li>改进了键盘操作。</li> <li>修复了 RHEL6 上的导致关闭会话后无法创建新会话时无法创建新会话时无法创建新会话。</li> <li>改进了对本机客户端上的 SOCKS5 代理的支持。</li> <li>修复了在虚拟会话上运行时导致 Headwave 崩溃和在虚拟会话上运行时导致 Chimera 崩溃的错误。</li> <li>改进了对虚拟会话的字体支持。</li> </ul>

## DCV 2017.0-4100 2017

内部版本号
<ul style="list-style-type: none"> <li>nice-dcv-server : 4100</li> <li>nice-dcv-client : 4100</li> <li>nice-xdcv : 118</li> <li>nice-dcv-gl : 229</li> <li>nice-dcv-gltest : 158</li> </ul>

内部版本号

- nice-dcv-simple-externalauthenticatataate : 35

## 文档历史记录

下表介绍了此版本的NICE DCV。

更改	说明	日期
NICE DCV 版本 2022.1	NICE DCV 2022.1 现已上市。 有关更多信息，请参阅 <a href="#">DCV 2022.1-13067—2022 年 6 月 29 日</a> (p. 125)。	2022 年 6 月 29 日
NICE DCV 版本 2022.0	NICE DCV 2022.0 现已上市。 有关更多信息，请参阅 <a href="#">DCV 2022.0-11954—2022 年 2 月 23 日</a> (p. 127)。	2022 年 2 月 23 日
NICE DCV 版本 2021.3	NICE DCV 2021.3 现已上市。 有关更多信息，请参阅 <a href="#">DCV 2021.3-11591—2021 年 12 月 20 日</a> (p. 128)。	2021 年 12 月 20 日
NICE DCV 版本 2021.2	NICE DCV 2021.2 现已上市。 有关更多信息，请参阅 <a href="#">DCV 2021.2-11048—2021 年 9 月 1 日</a> (p. 129)。	2021 年 9 月 1 日
NICE DCV 版本 2021.1	NICE DCV 2021.1 现已上市。 有关更多信息，请参阅 <a href="#">DCV 2021.1-10557—2021 年 5 月 31 日</a> (p. 131)。	2021 年 331 年 331 年
NICE DCV 版本 2021.0	NICE DCV 2021.0 现已推出。 有关更多信息，请参阅 <a href="#">DCV 2021.0-10242—2021 年 4 月 12 日</a> (p. 131)。	2021 年 4 月 12 日
NICE DCV Web 客户端软件开发工具包	NICE DCV Web 客户端 SDK 现已推出。NICE DCV Web 客户端 SDK 是一个 JavaScript 库，可用于开发自己的 NICE DCV Web 浏览器客户端应用程序，最终用户可以使用这些应用程序连接到正在运行的 NICE DCV 会话并与其交互。有关更多信息，请参阅 <a href="#">NICE DCV Web 客户端 SDK 开发人员指南</a> 。	2021 年 3 月 24 日
NICE DCV 版本 2020.2	NICE DCV 2020.2 现已上市。 有关更多信息，请参阅 <a href="#">DCV 2020.2-9508—2020 年 11 月 11 日</a> (p. 132)。	2020 年 11 月 11 日

更改	说明	日期
NICE DCV 版本 2020.1	NICE DCV 2020.1 现已推出。有关更多信息，请参阅 <a href="#">DCV 2020.1-8942—2020 年 8 月 3 日</a> (p. 133)。	2020 年 8 月 3 日
NICE DCV 版本 2020.0	NICE DCV 2020.0 包括对环绕立体声 7.1、触摸和触控笔以及使用新 Microsoft Edge 浏览器的多显示器的多显示器的支持。有关更多信息，请参阅 <a href="#">安装 NICE DCV 服务器</a> 中的 NICE DCV 管理员指南。	2020 年 4 月 16 日
HTTP 响应标头	可将 NICE DCV 服务器配置为发送其他 HTTP 响应标头。	2019 年 8 月 26 日
macOS 客户端	NICE DCV 现在提供了一个 macOS 客户端。有关更多信息，请参阅 <a href="#">macOS 客户端</a> 中的 NICE DCV 用户指南。	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收到的智能卡数据，以帮助提高性能。有关更多信息，请参阅 <a href="#">配置智能卡缓存</a> 中的 NICE DCV 管理员指南。	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 .04. 04、SLES 12 和 Ubuntu 16.04。有关更多信息，请参阅 <a href="#">Linux 客户端</a> 中的 NICE DCV 用户指南。	2018 年 8 月 29 日
更新了“参数参考”	“参数参考”已更新。有关更多信息，请参阅 <a href="#">NICE DCV 服务器参数参考</a> 中的 NICE DCV 管理员指南。	2018 年 8 月 7 日
USB 远程控制	NICE DCV 使客户端可以使用专门的 USB 设备，如 3D 指点设备或绘图板。有关更多信息，请参阅 <a href="#">启用 USB 远程处理</a> 中的 NICE DCV 管理员指南。	2018 年 8 月 7 日
NICE DCV 的初始版本。	此内容的第一版。	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。