



管理者ガイド

# NICE DCV



# NICE DCV: 管理者ガイド

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

Amazon の商標およびトレードドレスは、Amazon 以外の製品およびサービスに使用することはできません。また、お客様に誤解を与える可能性がある形式で、または Amazon の信用を損なう形式で使用することもできません。Amazon が所有していないその他のすべての商標は、Amazon との提携、関連、支援関係の有無にかかわらず、それら該当する所有者の資産です。

# Table of Contents

NICE DCV とは .....	1
NICE DCV の仕組み .....	1
機能 .....	1
料金 .....	2
NICE DCV サーバー .....	4
要件 .....	4
サポートされている機能 .....	6
設定 .....	9
インストール .....	9
Windows .....	10
Linux .....	16
ライセンス .....	55
NICE DCV ライセンス要件 .....	55
拡張評価ライセンスのインストール .....	58
実稼働ライセンスのインストール .....	60
実稼働ライセンスの更新 .....	72
アップグレード .....	72
互換性に関する考慮事項 .....	73
Windows での NICE DCV Server のアップグレード .....	73
Linux でのNICE DCVサーバーのアップグレード .....	74
アンインストール .....	74
Windows での NICE DCV Server のアンインストール .....	74
Linux での NICE DCV サーバーのアンインストール .....	75
イメージング .....	76
イメージの構築 .....	77
イメージパイプラインへの の追加 .....	77
Server の管理 .....	79
サーバーの起動 .....	79
サーバーの停止 .....	81
の有効化 QUIC UDP .....	82
TCP/UDP ポートとアドレスの変更 .....	84
サーバーTCP/UDPポートの変更 .....	85
特定のエンドポイントのリスニング .....	87
TLS 証明書の管理 .....	89

アイドル状態のクライアントの切断 .....	91
Linux でGPUの共有の有効化 .....	93
タッチスクリーンとスタイラスのサポートの有効化 .....	95
スタイラスの設定 .....	98
ゲームパッドサポートの有効化 .....	98
Xbox 360 コントローラーのサポート .....	99
USB リモート化の有効化 .....	100
スマートカードのキャッシュの設定 .....	102
WebAuthn リダイレクトの設定 .....	103
WebAuthn リダイレクトブラウザ拡張機能のセットアップ .....	104
セッションストレージの有効化 .....	106
Windows でのセッションストレージの有効化 .....	106
Linux でのセッションストレージの有効化 .....	107
Linux でのプリンターの設定 .....	108
プリンター問題のトラブルシューティング .....	110
Linux でのクリップボードの設定 .....	111
クライアントクリップボードコンテンツをプライマリ選択に貼り付ける .....	111
クライアントクリップボードにプライマリ選択コンテンツをコピーする .....	112
マルチチャンネルオーディオの設定 .....	113
Windows NICE DCVサーバーでのオーディオチャネルの設定 .....	115
Linux NICE DCVサーバーでのオーディオチャネルの設定 .....	115
HTTP ヘッダーの設定 .....	117
Windows NICE DCV Server でのHTTPヘッダーの設定 .....	117
Linux NICE DCVサーバーでのHTTPヘッダーの設定 .....	118
認証の設定 .....	118
Windows での認証の設定 .....	119
Linux での認証の設定 .....	121
外部認証システムによる認証の設定 .....	122
認可の設定 .....	122
デフォルトのアクセス許可ファイル .....	123
カスタムアクセス許可ファイル .....	123
アクセス許可ファイルでの作業 .....	123
セッションの管理をする .....	130
NICE DCV セッションの概要。 .....	130
コンソールセッション .....	131
仮想セッション .....	132

コマンドラインツールを使用したセッションの管理 .....	132
Windows NICE DCV サーバーでのコマンドラインツールの使用 .....	133
Linux NICE DCV サーバーでのコマンドラインツールの使用 .....	133
コマンドラインツールの使用 .....	133
セッションの開始 .....	135
コンソールセッションおよび仮想セッションの手動開始 .....	136
自動コンソールセッションの有効化 .....	140
セッションの停止 .....	142
Syntax .....	143
例 .....	143
実行中のセッションの管理 .....	143
セッションストレージの管理 .....	143
セッション認可の管理 .....	145
セッション表示レイアウトの管理 .....	146
セッション名の管理 .....	149
セッションのタイムゾーンの管理 .....	151
セッションの表示 .....	152
すべてのアクティブなセッションを一覧表示する .....	152
特定のセッションに関する情報を表示する .....	153
スクリーンショットの取得 .....	154
構文 .....	155
オプション .....	155
例 .....	156
方法 .....	158
外部認証を使用する .....	158
NICE DCV サーバー設定 .....	159
トークンの使用 .....	160
認証サービス要件 .....	160
アイドル状態のセッションの検索と停止 .....	162
X サーバーへのリモート X 接続を有効にする .....	163
X サーバーへのリモート X 接続を有効にする .....	164
NICE DCV ウェブブラウザクライアントを iFrame 内に埋め込む .....	165
トラブルシューティング .....	168
ログファイルの使用 .....	168
Windows でのログファイルの詳細レベル変更 .....	169
Linux でのログファイルの詳細レベル変更 .....	170

Linux での仮想セッション作成のトラブルシューティング .....	170
Linux での仮想セッション作成失敗の調査 .....	170
Linux でのフェイルセーフ仮想セッションの作成 .....	172
UID を変更すると Linux セッションが開始されない .....	173
Windows のカーソルの問題の解決 .....	173
IntelliJ IDEA へのコピーと貼り付けの修正 .....	174
自己署名証明書によるリダイレクトの説明 .....	175
Windows の NVIDIA GPU でのマルチモニター/全画面表示の失敗 .....	176
NICE DCV のパフォーマンスと統計のモニタリング .....	177
カウンターセット .....	177
パラメータリファレンス .....	187
audio パラメータ .....	188
clipboard パラメータ .....	189
connectivity パラメータ .....	193
display パラメータ .....	201
display/linux パラメータ .....	211
extensions パラメータ .....	211
input パラメータ .....	212
license パラメータ .....	213
log パラメータ .....	215
printer パラメータ .....	220
redirection パラメータ .....	223
security パラメータ .....	224
session-management パラメータ .....	235
session-management/automatic-console-session パラメータ .....	239
session-management/defaults パラメータ .....	241
smartcard パラメータ .....	242
webauthn パラメータ .....	243
webcam パラメータ .....	244
windows パラメータ .....	245
設定パラメータの変更 .....	246
Windows NICE DCV サーバー .....	247
Linux NICE DCV サーバー .....	248
NICE DCV のサポート終了 .....	249
EOSL タイムライン .....	249
お客様向けの EOSL パス .....	250

EOSL に関するよくある質問 .....	250
セキュリティ .....	252
データ保護 .....	252
データ暗号化 .....	253
コンプライアンス検証 .....	254
リリースノートとドキュメント履歴 .....	255
リリースノート .....	255
NICE DCV 2023.1-16388 .....	257
NICE DCV 2023.1-16388 .....	258
NICE DCV 2023.1-16388 .....	259
NICE DCV 2023.1-16220 .....	261
NICE DCV 2023.0-15487 .....	263
NICE DCV 2023.0-15065 .....	264
NICE DCV 2023.0-15022 .....	265
NICE DCV 2023.0-14852 .....	267
NICE DCV 2022.2-14521 .....	268
NICE DCV 2022.2-14357 .....	269
NICE DCV 2022.2-14175 .....	269
NICE DCV 2022.2-14126 .....	270
NICE DCV 2022.2-13907 .....	271
NICE DCV 2022.1-13300 .....	272
NICE DCV 2022.1-13216 .....	273
NICE DCV 2022.1-13067 .....	273
NICE DCV 2022.0-12760 .....	274
NICE DCV 2022.0-12627 .....	275
NICE DCV 2022.0-12123 .....	275
NICE DCV 2022.0-11954 .....	276
NICE DCV 2021.3-11591 .....	278
NICE DCV 2021.2-11445 .....	278
NICE DCV 2021.2-11190 .....	279
NICE DCV 2021.2-11135 .....	280
NICE DCV 2021.2-11048 .....	281
DCV 2021.1-10851 .....	283
DCV 2021.1-10598 .....	283
DCV 2021.1-10557 .....	284
DCV 2021.0-10242 .....	285

---

DCV 2020.2-9662 .....	286
DCV 2020.2-9508 .....	286
DCV 2020.1-9012 .....	287
DCV 2020.1-9012 .....	288
DCV 2020.1-8942 .....	288
DCV 2020.0-8428 .....	290
DCV 2019.1-7644 .....	291
DCV 2019.1-7423 .....	292
DCV 2019.0-7318 .....	292
DCV 2017.4-6898 .....	294
DCV 2017.3-6698 .....	295
DCV 2017.2-6182 .....	298
DCV 2017.1-5870 .....	300
DCV 2017.1-5777 .....	300
DCV 2017.0-5600 .....	301
DCV 2017.0-5121 .....	302
DCV 2017.0-4334 .....	302
DCV 2017.0-4100 .....	303
ドキュメント履歴 .....	303
.....	CCCX

# NICE DCV とは

NICE DCV は、高性能のリモートディスプレイプロトコルです。さまざまなネットワーク条件で、リモートデスクトップやアプリケーションストリーミングをクラウドやデータセンターからあらゆるデバイスへ安全に配信できます。NICE DCV と Amazon EC2 を使用すると、グラフィックスを多用するアプリケーションを Amazon EC2 インスタンス上でリモートで実行できます。結果をより控えめなクライアントマシンにストリーミングできるため、高価な専用ワークステーションが不要になります。

## トピック

- [NICE DCV の仕組み](#)
- [NICE DCV の機能](#)
- [NICE DCV の料金](#)

## NICE DCV の仕組み

NICE DCV を使用するには、サーバーに NICE DCV サーバーソフトウェアをインストールします。NICE DCV サーバーソフトウェアは、安全な[セッション](#)を作成するために使用されます。サーバー上にアプリケーションをインストールして実行します。サーバーはハードウェアを使用して、インストールされているアプリケーションに必要な高性能処理を実行します。ユーザーは、NICE DCV クライアントアプリケーションを使用してセッションにリモート接続してアプリケーションにアクセスします。接続が確立されると、NICE DCV サーバーソフトウェアはアプリケーションのビジュアル出力を圧縮し、暗号化されたピクセルストリームでクライアントアプリケーションにストリームを返します。クライアントアプリケーションは圧縮されたピクセルストリームを受信し、復号してローカルディスプレイに出力します。

## NICE DCV の機能

NICE DCV には以下の機能があります。

- デスクトップ全体を共有 — 高性能 NICE DCV プロトコルを使用してリモートデスクトップ全体を完全に制御します。
- イメージのみを転送 — ジオメトリやシーン情報ではなく、レンダリングされたイメージをピクセルとして転送します。顧客の固有情報がネットワーク経由で送信されないため、追加のセキュリティレイヤーが提供されます。

- H.264 ベースのエンコーディングをサポート — H.264 ベースの動画圧縮とエンコーディングを使用して帯域幅の消費を削減します。
- 可逆品質の動画圧縮をサポート - ネットワークとプロセッサの条件が許す限り、可逆品質のビデオ圧縮をサポートします。
- 表示レイアウトと一致 — クライアントウィンドウのサイズに合わせて、サーバーの画面解像度と表示レイアウトを自動的に調整します。
- マルチスクリーンをサポート — セッションデスクトップを最大 4 台のモニターに拡張できます。高ピクセル密度モニターは、Windows および macOS のネイティブクライアントでサポートされています。
- 圧縮レベルを適応させる — ネットワークの利用可能な帯域幅とレイテンシーに基づいて動画圧縮レベルを自動的に調整します。
- コラボレーションを可能にする — 複数のコラボレーションクライアントをサポートする動的セッションを提供します。クライアントは、セッション中いつでも接続および切断できます。
- サーバーごとに複数のセッションをサポート (Linux NICE DCV サーバーのみ) — Linux NICE DCV サーバーごとに複数の仮想セッションをサポートし、コスト削減を最大化します。
- GPU 共有のサポート (Linux NICE DCV サーバーのみ) — Linux NICE DCV サーバーで実行されている複数の仮想セッション間で、1 つ以上の物理 GPU を共有できます。
- タッチ入力、スタイラス入力、ゲームパッドをサポート — ローカルコンピューターに接続された入力デバイスを使用して、リモート NICE DCV セッションを操作できます。
- WebAuthn、スマートカード、スタイラス、USB のリモート化をサポート — NICE DCV セッションで周辺機器をローカルコンピューターと同じように使用できます。
- オーディオのインとアウト、印刷、コピーと貼り付けをサポート — セッションとローカルコンピューターの間でこれらのキー操作を実行できます。
- ファイル転送をサポート — セッションとローカルコンピューター間でファイルを転送できます。
- HTML5 クライアントを提供 - Windows および Linux 上の最新のウェブブラウザで利用できる HTML5 クライアントを提供します。
- 最新の Linux デスクトップ環境をサポート — RHEL 8 上の Gnome 3 など最新の Linux デスクトップをサポートします。

## NICE DCV の料金

Amazon EC2 インスタンスで NICE DCV サーバーを使用する場合、追加料金は発生しません。使用するインスタンスおよびその他の Amazon EC2 機能に対して標準料金がかかります。

それ以外の場合はライセンスが必要です。詳細については、「[NICE DCV サーバーのライセンス](#)」を参照してください。

# NICE DCV サーバー

NICE DCV サーバーは、Windows と Linux で使用できます。どちらのサーバーも同様の機能を提供しますが、いくつかの違いがあります。ニーズに最適な NICE DCV サーバーを選択します。次の表は、Windows および Linux の NICE DCV サーバーでサポートされている機能を比較したものです。

## トピック

- [要件](#)
- [サポートされている機能](#)

## 要件

NICE DCV の良好なユーザーエクスペリエンスを確保するために、サーバーが次の最小要件を満たしていることを確認してください。ユーザーエクスペリエンスは、NICE DCV サーバーから NICE DCV クライアントにストリーミングされるピクセル数に大きく左右されるため注意が必要です。

Amazon EC2 インスタンスに NICE DCV サーバーをインストールする場合は、Amazon EC2 のインスタンスタイプ G3、G4dn、G4ad、G5 を使用することをお勧めします。これらのインスタンスタイプは、ハードウェアベースの OpenGL および GPU 共有をサポートする GPU を提供します。詳細については、「[Amazon EC2 G3 インスタンス](#)」、「[Amazon EC2 G4 インスタンス](#)」、および「[Amazon EC2 G5 インスタンス](#)」を参照してください。

NICE DCV サーバーは、他のインスタンスタイプにもインストールできますが、画面解像度の制限が生じる可能性があります。Windows Server 2016 でこの制限を回避するには、[EC2 用 NICE DCV 仮想ディスプレイドライバ](#)をダウンロードしてインストールします。DCV 2023.1 以降を実行している Windows Server 2019 以降では、追加の操作は必要ありません。

サーバーは、次の表に示す最小要件を満たしている必要があります。

	Windows Server	[Linux サーバー]
オペレーティングシステム	<ul style="list-style-type: none"> <li>• Windows 10</li> <li>• Windows 11</li> <li>• Windows Server 2016</li> <li>• [Windows Server 2019]</li> <li>• Windows Server 2022</li> </ul>	<ul style="list-style-type: none"> <li>• Amazon Linux 2</li> <li>• CentOS 7.6 以降</li> <li>• CentOS Stream 8</li> <li>• CentOS Stream 9</li> <li>• RHEL 7.6 以降</li> </ul>

	Windows Server	[Linux サーバー]
	<p><b>Note</b></p> <p>サポートされているすべての Windows オペレーティングシステムでは、.NET Framework 4.5 が必要で、x86-64 アーキテクチャをサポートしている必要があります。</p>	<ul style="list-style-type: none"> <li>• RHEL 8.x</li> <li>• RHEL 9.x</li> <li>• SP5 以降の SUSE Linux Enterprise 12</li> <li>• SUSE Linux Enterprise 15 with SP5</li> <li>• Rocky Linux 8.5 以降</li> <li>• Rocky Linux 9</li> <li>• Ubuntu 20.04</li> <li>• Ubuntu 22.04</li> </ul>
サポートされているアーキテクチャ	64 ビット x86	<ul style="list-style-type: none"> <li>• 64 ビット x86</li> <li>• 64 ビット ARM (Amazon Linux 2、RHEL 7.x/8.x/9.x、CentOS 7/8/9、Rocky Linux 8/9、および Ubuntu 22.04 を実行する Amazon EC2 インスタンスでのみサポート)</li> </ul>
GPU	<p>(オプション) ハードウェアベースのビデオエンコーディングには、NVIDIA または AMD GPU が必要です。サーバーに GPU がない場合は、ソフトウェアベースのビデオエンコーディングが使用されます。</p> <p><b>Note</b></p> <ul style="list-style-type: none"> <li>• NVIDIA GPU では、ハードウェアベースのビデオエンコーディングに NVENC が必要です。コンピューティング機能を備えた NVIDIA GPU <math>\geq 3.5</math> が必要です。</li> <li>• AMD GPU のハードウェアベースのビデオエンコーディングには、Linux または Windows 用の Advanced Media Framework (AMF)、または Windows 専用の Rapidfire が必要です。Linux の場合、AMD ドライバーにより提供される追加のパッケージ <code>amf-amdgpu-pro</code> をインストールすれば、AMF エンコーダを Ubuntu インスタンスで使用できます。</li> </ul>	

	Windows Server	[Linux サーバー]
		仮想セッション間での GPU 共有には NVIDIA GPU が必要です。 <div style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p><b>Note</b></p> <p>AMD GPU を搭載した Linux サーバーでは、コンソールセッションのみがサポートされています。</p> </div>
ネットワーク	NICE DCV サーバーでは、デフォルトで、ポート 8443 経由で通信が行われます。ポートは設定可能ですが、1024 より大きい必要があります。サーバーに必要なポートを介した通信が許可されていることを確認します。	

**Note**

NICE DCV は、サポートが終了したオペレーティングシステムをサポートしていません。オペレーションシステムについてはベンダーにお問い合わせください。

NICE DCV クライアント要件の詳細については、「NICE DCV ユーザーガイド」の「[NICE DCV クライアント要件](#)」を参照してください。

## サポートされている機能

次の表は、Windows と Linux の NICE DCV サーバーでサポートされている機能を比較したものです。

機能	<a href="#">Windows NICE DCV サーバー</a>	<a href="#">Linux NICE DCV サーバー</a>
<a href="#">コンソールセッション</a>	✓	✓
<a href="#">仮想セッション</a>	✗	✓

機能	<u>Windows NICE DCV サーバー</u>	<u>Linux NICE DCV サーバー</u>
<u>QUIC (UDP) トランスポートプロトコル</u>	✓	✓
<u>設定可能な TCP/UDP ポートおよびアドレス</u>	✓	✓
<u>カスタム TLS 証明書</u>	✓	✓
<u>アイドル状態のクライアントの切断</u>	✓	✓
<u>GPU 共有</u>	✗	✓
<u>USB リモート化</u>	✓	✓
<u>スマートカードのサポート</u>	✓	✓
<u>ウェブカメラのサポート</u>	✓ (Windows 10 および Server 2016 以降)	✗
<u>セッションストレージとファイル転送</u>	✓	✓
<u>コピーと貼り付け</u>	✓	✓
<u>カスタム HTTP ヘッダー</u>	✓	✓
<u>セッションからの印刷</u>	✓	✓
<u>ステレオ 2.0 オーディオ再生</u>	✓	✓
<u>サラウンドサウンドオーディオ再生</u>	✓ (7.1 まで)	✓ (5.1 まで)
<u>ステレオ 2.0 オーディオ録音</u>	✓	✓
<u>タッチスクリーンのサポート</u>	✓ (Windows 10 および Server 2012 以降)	✓

機能	<a href="#">Windows NICE DCV サーバー</a>	<a href="#">Linux NICE DCV サーバー</a>
<a href="#">スタイラスのサポート</a>	✓ (Windows 10 および Server 2019)	✓
<a href="#">ゲームパッドのサポート</a>	✓ (Windows 10 および Server 2016 以降)	✗
全画面表示 (選択したモニター)	✓	✗
タイムゾーンのリダイレクト	✓	✓
WebAuthn リダイレクト	✓	✗

NICE DCV クライアント機能の詳細については、「[NICE DCV ユーザーガイド](#)」の「NICE DCV クライアント機能」を参照してください。

# NICE DCV サーバーのセットアップ

を使用するにはNICE DCV、NICE DCVセッションをホストするNICE DCVサーバーにサーバーソフトウェアをインストールします。ソフトウェアに正しくライセンスが付与されていることを確認します。

以下のトピックでは、NICE DCVサーバーをインストールしてライセンスする方法について説明します。[ライセンス](#)に関するトピックは、オンプレミスおよびその他のクラウドベースサーバーでのインストールにのみ適用されます。これは、Amazon EC2インスタンスでNICE DCVサーバーを使用するためのライセンスを必要としないためです。

## トピック

- [NICE DCV サーバーのインストール](#)
- [NICE DCV サーバーのライセンス](#)
- [NICE DCV サーバーのアップグレード](#)
- [NICE DCV サーバーのアンインストール](#)
- [イメージングNICE DCVサーバー](#)

## NICE DCV サーバーのインストール

以下のトピックでは、Windows および Linux に最新バージョンのNICE DCVサーバーをインストールする方法について説明します。Amazon EC2インスタンス、またはその他のオンプレミスまたはクラウドベースのサーバーNICE DCVに をインストールする場合は、次の手順に従います。

### Note

サーバーの以前のバージョンからNICE DCV最新バージョンにアップグレードする場合は、「」を参照してください[NICE DCV サーバーのアップグレード](#)。

## トピック

- [Windows への NICE DCV Server のインストール](#)
- [Linux への NICE DCV サーバーのインストール](#)

## Windows への NICE DCV Server のインストール

このセクションでは、Windows にNICE DCVサーバーをインストールする方法について説明します。

### トピック

- [Amazon EC2インスタンスの Windows NICE DCVサーバーの前提条件](#)
- [Windows への NICE DCV Server のインストール](#)

## Amazon EC2インスタンスの Windows NICE DCVサーバーの前提条件

このトピックでは、NICE DCVサーバーをインストールする前に Windows Amazon EC2インスタンスを設定する方法について説明します。Amazon EC2 Windows インスタンスにNICE DCVサーバーをインストールしない場合は、次の前提条件をスキップします。

### トピック

- [高速コンピューティングインスタンスの前提条件](#)
- [その他のインスタンスファミリーの前提条件](#)

### 高速コンピューティングインスタンスの前提条件

### GPU グラフィックインスタンスの前提条件

GPU グラフィックインスタンス (G2, G3, G4dn, G4ad) NVIDIAまたは AMD GPUドライバーをインストールして設定することをお勧めします。G5 GPU ドライバーは以下を許可します。

- アプリケーションの DirectX および OpenGL ハードウェアアクセラレーション
- H.264 ビデオストリーミングエンコードのハードウェアアクセラレーション
- カスタマイズ可能なサーバーモニターの解像度
- サーバーモニターの最大解像度の増加— 最大 4096x2160
- サーバーモニター数の増加

GPU グラフィックインスタンスにNVIDIA GPUドライバーをインストールする方法については、「Amazon EC2ユーザーガイド」の以下のトピックを参照してください。

- を使用するインスタンス NVIDIA GPU (G2, G3, G4dnインスタンスなど) については、「[Windows へのNVIDIAドライバーのインストール](#)」を参照してください。G5

- 使用するインスタンス AMD GPU (G4ad インスタンスなど) については、[「Windows インスタンスにAMDドライバーをインストールする」](#)を参照してください。

Amazon EC2 G4ad インスタンスの詳細については、[新しい Amazon EC2 G4ad インスタンスのブログ記事「ディープダイブ」](#)を参照してください。

#### その他の高速コンピューティングインスタンスの前提条件

GPU グラフィックインスタンスではない高速コンピューティングインスタンス (P2, P3、P3dn インスタンスなど) を使用している場合は、適切なNVIDIAGPUドライバーをインストールして設定することをお勧めします。NVIDIA GPU ドライバーは、H.264 ビデオストリーミングエンコーディングのハードウェアアクセラレーションを有効にします。

高速コンピューティングインスタンスにNVIDIAGPUドライバーをインストールする方法については、「Amazon EC2ユーザーガイド」の[「パブリックNVIDIAドライバー」](#)を参照してください。

高速コンピューティングインスタンスにNVIDIAGPUドライバーをインストールしても、サーバーモニターの制限や解像度は向上しません。サーバーモニター解像度のサポートを追加するには、NVIDIAGRIDSドライバーをインストールします。詳細については、NVIDIAウェブサイト[NVIDIAの「ソフトウェアGPU」](#)を参照してください。

#### その他のインスタンスファミリーの前提条件

高速コンピューティングインスタンス以外のインスタンスでは、Windows 2016 を使用している場合、または 2023.1 より前のNICE DCVサーバーバージョンを実行している場合は、NICE DCV仮想ディスプレイドライバーをインストールすることをお勧めします。このようなインスタンスには、汎用、コンピューティング最適化、メモリ最適化、ストレージ最適化などのインスタンスファミリーがあります。

NICE DCV 仮想ディスプレイドライバーをインストールすると、次のことが可能になります。

- 最大 4 つのモニターに対するサポート
- カスタム解像度に対するサポート
- 4K UHD解像度のサポート

Windows コントロールパネルを使用して、サーバーによってアタッチされたNICE DCVサーバーモニターを管理することはできません。

**Note**

NICE DCV 仮想ディスプレイドライバーは、Windows Server 2016 以降でサポートされています。間接ディスプレイドライバー (IDD) はサーバーにパッケージ化されているため、Windows Server 2019 以降でDCVサーバー 2023.1 以降を使用している場合は、ドライバーは必要ありませんDCV。IDD が推奨されますが、[GetConsoleScreenshot](#)機能は期待どおりに機能しません。

**Important**

ドライバーなど、他のGPUドライバーと共にNICE DCV仮想ディスプレイNVIDIA GPUドライバーをインストールすると、競合が発生する可能性があります。競合を避けるため、NICE DCV仮想ディスプレイドライバーを他のGPUドライバーと組み合わせてインストールしないことをお勧めします。

インスタンスにNICE DCV仮想ディスプレイドライバーをインストールするには

1. [NICE DCV ウェブサイト](#) からNICE DCV仮想ディスプレイドライバーインストーラをダウンロードします。
2. ウィザードを実行してドライバーをインストールするには、インストールファイルを開くか、またはダブルクリックします。または、次のコマンドを使用して自動インストールを実行します。

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

3. インスタンスを再起動して、そのインスタンスに再接続します。

## Windows への NICE DCV Server のインストール

インストールウィザードを使用して、Windows ホストNICE DCVサーバーにサーバーをインストールできます。ウィザードは、NICE DCVサーバーのインストールをカスタマイズする方法を示す一連のステップをガイドします。また、コマンドラインを使用して自動インストールを実行することもできます。その場合はデフォルト設定によりインストール手順が自動化されます。

### 目次

- [ウィザードの使用](#)

## • [自動インストール](#)

### ウィザードの使用

ガイド付きインストールにはNICE DCVサーバーインストールウィザードを使用します。

ウィザードを使用してNICE DCVサーバーを Windows にインストールするには

1. NICE DCV サーバーをインストールするサーバーを起動して接続します。
2. [NICE DCV](#) ウェブサイトからNICE DCVサーバーインストーラをダウンロードします。

#### Note

NICE DCV サーバーは 64 ビットバージョンでのみ使用でき、64 ビット Windows オペレーティングシステムでサポートされています。

#### Tip

ダウンロード用ウェブサイトの[最新のパッケージ](#)には、利用可能な最新バージョンに常に接続されているリンクが含まれます。これらのリンクを使用して、最新のNICE DCVパッケージを自動的に取得できます。

3. `nice-dcv-server-x64-Release-2023.1-version_number.msi` を実行します。
4. ようこそ画面で、[Next] を選択します。
5. [End-User License Agreement] (エンドユーザーライセンス契約) 画面で、使用許諾書を読みます。同意する場合は、[I accept the terms in the License Agreement] (使用許諾契約書の条項に同意します) チェックボックスを選択して、[Next] (次へ) を選択します。
6. (オプション) [コンポーネントの選択] 画面で項目を選択して、インストールするコンポーネントを設定します。インストールするコンポーネントをマークするには、その項目を選択し、「ローカルハードドライブにインストールされる」を選択します。インストール対象からコンポーネントを除外するには、その項目を選択し、[すべての機能が使用できなくなる] を選択します。
7. DCV サービス設定画面で、次の操作を行います。
  - a. (オプション) サーバーのファイアウォールを手動で構成して必要なポートで通信できるようにするには、[No, I will manually configure my firewall later (いいえ、後でファイアウォールを手動で構成します)] を選択します。

- b. (オプション) インストール後にNICE DCVサーバーを手動で起動するには、いいえ、DCVサービスを手動で起動します。このオプションを選択すると、インストールが完了した後に自動的にコンソールセッションを開始することはできません。このオプションを選択すると、ステップ9はスキップされます。
8. [Next (次へ)] を選択します。
  9. DCV セッション管理設定画面で、自動コンソールセッションの所有者を指定します。または、インストールの完了後に自動コンソールセッションが開始されないようにするには、[いいえ、セッションを手動で作成します] を選択します。

 Note

以前にサーバーを自動的に開始することを選択した場合にのみ、このステップを完了します。

10. [Install] (インストール) を選択します。

## 自動インストール

デフォルト設定では、自動インストールにより次の処理が実行されます。

- ポート 8443 を介した通信を許可するファイアウォールルールを追加します。
- NICE DCV サーバーの自動起動を有効にします。
- 自動コンソールセッションを作成します。
- コンソールセッションの所有者を、インストールを実行するユーザーに設定します。

インストールコマンドに次のオプションを追加することで、デフォルトのアクションを上書きすることができます。

- `DISABLE_FIREWALL=1` — インストーラによりファイアウォールルールが追加されないようにします。
- `DISABLE_SERVER_AUTOSTART=1` — インストール後にNICE DCVサーバーが自動的に起動するのを防ぎます。
- `DISABLE_AUTOMATIC_SESSION_CREATION=1` — インストーラにより自動コンソールセッションが開始されないようにします。
- `AUTOMATIC_SESSION_OWNER=owner_name` — 自動コンソールセッションの別の所有者を指定します。

- ADDLOCAL=*component\_list* — インストールする要素のセットに要素を追加します。
- REMOVE=*component\_list* — インストールする要素のセットから要素を削除します。

**Note**

REMOVE オプションは ADDLOCAL オプションの後で評価されます。両方のリストに入っている要素はインストールされません。

component\_list は、次の値が含まれている可能性があるコンマ区切りリストです。

- audioMicDriver: マイクドライバ
- audioSpkDriver: スピーカードライバ
- printerDriver: プリンタードライバ
- usbDriver: USB デバイスリモート化ドライバ (デフォルトでは無効)
- webcamDriver: ウェブカメラのドライバ
- gamepadDriver: ゲームパッドドライバ
- webClient: ウェブクライアント
- webauthn: Webauthn リダイレクト
- ALL: すべてのコンポーネント

無人インストールを使用して Windows に NICE DCV サーバをインストールするには

1. サーバをインストールしようとしている NICE DCV サーバを起動して接続します。
2. [NICE](#) ウェブサイトから NICE DCV サーバインストーラをダウンロードします。

**Note**

NICE DCV サーバは 64 ビットバージョンでのみ使用でき、64 ビット Windows オペレーティングシステムでサポートされています。

3. コマンドプロンプトウィンドウを開いて、インストーラをダウンロードしたフォルダに移動します。
4. 次のいずれかの例に示されているように、自動インストーラを実行します。
  - デフォルトのコンポーネントをインストールします。

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

- すべてのコンポーネントをインストールします。

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

- コンポーネントのサブセットをインストールします。

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

## Linux への NICE DCV サーバーのインストール

このセクションでは、Linux にNICE DCVサーバーをインストールする方法について説明します。

### トピック

- [Linux NICE DCVサーバーの前提条件](#)
- [Linux にNICE DCVサーバーをインストールする](#)
- [インストール後のチェック](#)

### Linux NICE DCVサーバーの前提条件

NICE DCV では、クライアントは Linux サーバー上のリモートグラフィカル X セッションにアクセスできます。これにより、対応する Linux デスクトップにアクセスできます。NICE DCV は、コンソールセッションと仮想セッションの 2 種類の Linux デスクトップストリーミングをサポートしています。コンソールおよび仮想セッションの詳細については、「[NICE DCV セッションの管理](#)」を参照してください。

このトピックでは、Linux サーバーNICE DCVで を使用するために必要な前提条件をインストールする方法について説明します。

### 内容

- [デスクトップ環境とデスクトップマネージャーをインストールする](#)
- [Wayland プロトコルを無効にする \(GDM3 のみ \)](#)

- [X サーバーを設定する](#)
- [glxinfo ユーティリティをインストールする](#)
- [OpenGL ソフトウェアレンダリングの検証](#)
- [グラフィックインスタンス用のGPUドライバーをインストールする](#)
- [以外のGPUインスタンス用のXDummyドライバーをインストールする](#)

デスクトップ環境とデスクトップマネージャーをインストールする

デスクトップ環境とデスクトップマネージャーをインストールして、Linux サーバーNICE DCVでの操作性を向上させます。

デスクトップ環境は、Linux オペレーティングシステムの操作に役立つグラフィカルユーザーインターフェイス (GUI) です。デスクトップ環境は複数あり、その多くでNICE DCV動作します。デスクトップマネージャーは、ユーザーのログイン画面を管理し、デスクトップ環境セッションと X サーバーを起動および停止するプログラムです。

次のタブ付きのコンテンツは、サポートされているオペレーティングシステムにデフォルトのデスクトップ環境とデスクトップマネージャーをインストールするステップを示しています。また、サポートされているオペレーティングシステムで X サーバーを設定して開始する方法も示しています。

RHEL, CentOS, and Rocky Linux

RHEL、CentOS、Rocky Linux のデフォルトのデスクトップ環境は Gnome3 で、デフォルトのデスクトップマネージャーは GDM です。

CentOS RHEL、Rocky Linux にデスクトップ環境とデスクトップマネージャーをインストールして設定するには

1. デスクトップ環境とデスクトップマネージャーパッケージをインストールします。

- RHEL および Rocky Linux

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

- CentOS

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

2. ソフトウェアパッケージを更新して、Linux サーバーが最新のものになります。

```
$ sudo yum upgrade
```

3. Linux サーバーを再起動します。

```
$ sudo reboot
```

## Amazon Linux 2

### Note

現在、NICE DCVは Amazon Linux 2023 と互換性がありません。AL2023 には、NICE DCVの実行に必要なグラフィカルデスクトップ環境は含まれていません。

Amazon Linux 2 のデフォルトのデスクトップ環境は Gnome3 で、デフォルトのデスクトップマネージャーは GDM です。

Amazon Linux 2 にデスクトップ環境とデスクトップマネージャーをインストールして設定するには

1. デスクトップ環境とデスクトップマネージャーパッケージをインストールします。

```
$ 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. ソフトウェアパッケージを更新して、Linux サーバーが最新のものになるようにします。

```
$ sudo yum upgrade
```

3. Linux サーバーを再起動します。

```
$ sudo reboot
```

## Ubuntu 20.x and 22.x

Ubuntu 20.x/22.x の場合、デフォルトのデスクトップ環境は Gnome3 で、デフォルトのデスクトップマネージャーは GDM3 です。Ubuntu 20.x 以降、LightDM は NICE でサポートされなくなりましたDCV。

20.x/22.x にデスクトップ環境とデスクトップマネージャーをインストールして設定するには

1. デスクトップ環境とデスクトップマネージャーパッケージをインストールします。

```
$ sudo apt update
```

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

のインストール GDM3

```
$ sudo apt install gdm3
```

2. GDM3 がデフォルトのデスクトップマネージャーとして設定されていることを確認します。

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

出力は次のとおりです。

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

がデフォルトのデスクトップマネージャーとして設定されていない場合GDM3は、次のコマンドを使用してデフォルトとして設定します。

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

3. ソフトウェアパッケージを更新して、Linux サーバーが最新のものになりますようにします。

```
$ sudo apt upgrade
```

4. Linux サーバーを再起動します。

```
$ sudo reboot
```

**Note**

仮想セッションで 2022.2 より NICE DCV 古いバージョンのを使用する場合、[既知の GDM 問題が発生する](#)可能性があります。仮想セッションを正しく機能させるために、次のいずれかのソリューションを導入できます。

- がないサーバーGPUでは、仮想セッションを実行する必要がないため、デスクトップマネージャーを無効にすることができます。仮想セッションを作成する前に、次のコマンドを実行して、システムがマルチユーザーモードで実行されるように設定します。

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

- があるサーバーではGPU、デスクトップマネージャーを無効にするだけでなく、仮想セッションを作成する前にシステム上で X サーバーを起動する必要があります。これを行うには、以下のコマンドを実行します。

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

```
sudo dcvstartx &
```

NICE DCV 2022.2 以降は、この問題の影響を受けません。

## SUSE Linux Enterprise 12.x

SUSE Linux Enterprise 12.x のデフォルトのデスクトップ環境は SLE Classic で、デフォルトのデスクトップマネージャーは GDM です。

SUSE Linux Enterprise 12.x でデスクトップ環境とデスクトップマネージャーをインストールして設定するには

1. デスクトップ環境とデスクトップマネージャーパッケージをインストールします。

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

2. GDM がデフォルトのデスクトップマネージャーとして設定されていることを確認します。

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

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

3. ソフトウェアパッケージを更新して、Linux サーバーが最新のものになるようにします。

```
$ sudo zypper update
```

4. Linux サーバーを再起動します。

```
$ sudo reboot
```

## SUSE Linux Enterprise 15.x

SUSE Linux Enterprise 15.x のデフォルトのデスクトップ環境は SLE Classic で、デフォルトのデスクトップマネージャーは GDM3 です。

SUSE Linux Enterprise 15.x でデスクトップ環境とデスクトップマネージャーをインストールして設定するには

1. デスクトップ環境とデスクトップマネージャーパッケージをインストールします。

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

2. GDM がデフォルトのデスクトップマネージャーとして設定されていることを確認します。

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

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

3. ソフトウェアパッケージを更新して、Linux サーバーが最新のものになるようにします。

```
$ sudo zypper update
```

#### 4. Linux サーバーを再起動します。

```
$ sudo reboot
```

##### Note

仮想セッションで 2022.2 より NICE DCV 古いバージョンのを使用する場合、[既知の GDM 問題が発生する](#)可能性があります。仮想セッションを正しく機能させるために、次のいずれかのソリューションを導入できます。

- がないサーバーGPUでは、仮想セッションを実行する必要がないため、デスクトップマネージャーを無効にすることができます。仮想セッションを作成する前に、次のコマンドを実行して、システムがマルチユーザーモードで実行されるように設定します。

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

- があるサーバーではGPU、デスクトップマネージャーを無効にするだけでなく、仮想セッションを作成する前にシステム上で X サーバーを起動する必要があります。これを行うには、以下のコマンドを実行します。

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

```
sudo dcvstartx &
```

NICE DCV 2022.2 以降は、この問題の影響を受けません。

#### Wayland プロトコルを無効にする (GDM3 のみ)

NICE DCV は Wayland プロトコルをサポートしていません。GDM3 デスクトップマネージャーを使用している場合は、Wayland プロトコルを無効にする必要があります。を使用していない場合は GDM3、このステップをスキップします。

#### Wayland プロトコルを無効にする方法

- 適切なテキストエディタを使用して以下のファイルを開きます。

- RHEL、CentOS、および SUSE Linux Enterprise 15.x

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

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

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

2. [daemon] セクションで WaylandEnable を false に設定します。

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

3. GDM サービスを再起動します。

- RHEL および CentOS

```
$ sudo systemctl restart gdm
```

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

```
$ sudo systemctl restart gdm3
```

- SUSE Linux Enterprise 15.x

```
$ sudo systemctl restart xdm
```

## X サーバーを設定する

コンソールセッションまたはGPU共有を使用する場合は、Linux サーバーが適切に設定され、X サーバーが実行されていることを確認する必要があります。

### Note

GPU 共有せずに仮想セッションを使用する場合は、X サーバーは必要ありません。

X サーバーパッケージは通常、デスクトップ環境とデスクトップマネージャーの依存関係としてインストールされます。Linux サーバーの起動時に自動的に起動するように X サーバーを設定することをお勧めします。

RHEL、CentOS、Amazon Linux 2、Ubuntu 18.x、20.x、22.x、SUSELinux Enterprise 12.x、15.x で X サーバーを設定して起動するには：

1. Linux サーバーの起動時に自動的に起動するように X サーバーを設定します。

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

コマンドが `graphical.target` を返した場合、X サーバーは自動的に起動するようにすでに設定されています。次のステップに進みます。

コマンドが `multi-user.target` を返した場合、X サーバーは自動的に起動するように設定されていません。次のコマンドを実行します。

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

2. X サーバーを起動します。

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

3. X サーバーが実行中であることを確認する

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

以下に示しているのは、X サーバーが動作している場合の出力例です。

```
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
```

### glxinfo ユーティリティをインストールする

glxinfo ユーティリティは、Linux サーバーの OpenGL 設定に関する情報を提供します。このユーティリティを使用すれば、Linux サーバーでサポートされるように設定されているのが OpenGL のハードウェアレンダリングかソフトウェアレンダリングかを判断できます。ドライバーとサポートされている拡張機能に関する情報が得られます。

glxinfo ユーティリティは、GL DCV のパッケージ依存関係としてインストールされます。したがって、GL DCV をインストールした場合、glxinfo ユーティリティはすでに Linux サーバーにインストールされています。

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

glxinfo ユーティリティをインストールするには

次のコマンドを実行します。

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

## Ubuntu

glxinfo ユーティリティをインストールするには

次のコマンドを実行します。

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

## SUSE Linux Enterprise

glxinfo ユーティリティをインストールするには

次のコマンドを実行します。

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

## OpenGL ソフトウェアレンダリングの検証

Linux 以外のGPUサーバーでは、OpenGL は Mesa ドライバーを使用するソフトウェアレンダリングモードでのみサポートされます。Linux 以外のGPUサーバーを使用していて、OpenGL を使用する場合は、Linux サーバーに Mesa ドライバーがインストールされ、適切に設定されていることを確認します。

### Note

これは Linux GPU以外のサーバーにのみ適用されます。

OpenGL ソフトウェアレンダリングが利用可能であることを確認するには

X サーバーが実行されていることを確認し、次のコマンドを使用します。

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

以下に示しているのは、OpenGL ソフトウェアレンダリングが利用可能な場合の出力例です。

```
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
```

グラフィックインスタンス用のGPUドライバーをインストールする

トピック

- [NVIDIA ドライバーのインストールと設定](#)
- [AMD ドライバーをインストールして設定する](#)

NVIDIA ドライバーのインストールと設定

専用の を備えた Linux NVIDIA サーバーではGPU、適切なNVIDIAドライバーがインストールされ、適切に設定されていることを確認します。Amazon EC2 Linux インスタンスにNVIDIAドライバーをインストールする方法については、「Amazon [ユーザーガイド](#)」の「[Linux サーバーへのNVIDIAドライバーのインストール](#)」を参照してください。 EC2

#### Note

- これは、 を使用する Linux サーバーNVIDIAGPUsにのみ適用されます。
- GRID ドライバーは、GPUインストールされた ごとに最大 4 つの 4K ディスプレイをサポートします。ゲームドライバーは、GPUインストールされた 1 台につき 4K ディスプレイを 1 つだけサポートします。

Linux サーバーにNVIDIAドライバーをインストールしたら、 を更新しますxorg.conf。

更新された xorg.conf を生成するには

1. 以下のコマンドを実行します。

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

G3, G4、または G5 Amazon EC2インスタンスを使用していて、マルチモニターコンソールセッションを使用する場合は、`--connected-monitor=DFP-0,DFP-1,DFP-2,DFP-3`パラメータを含めます。これは次のように行います。

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

#### Note

サーバーにレガシー `/etc/X11/XF86Config` ファイルがないことを確認します。存在する場合、`nvidia-xconfig` は必要な `/etc/X11/xorg.conf` ファイルを生成するのではなく、その設定ファイルを更新します。次のコマンドを実行して、レガシー `XF86Config` ファイルを削除します。

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

2. 変更を有効にするには、X サーバーを再起動します。

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

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

がハードウェアベースのビデオエンコーディング `NVIDIAGPU` をサポートしていることを確認するには

Ubuntu 20 では、`NVENC`エンコーディングがサポートされ、コンピューティング機能が 3.0 以上、または 3.5 以上であることを確認してください。

`NVENC` サポートを確認するには、[NVIDIA「ビデオエンコードとデコードGPUのサポートマトリックス」](#)を参照してください。コンピューティング機能を確認するには、[NVIDIA「コンピューティング容量表」](#)を参照してください。

NVIDIA GPU がNVENCエンコーディングをサポートしていない場合、または必要なコンピューティング機能がない場合は、ソフトウェアベースのビデオエンコーディングが使用されます。

OpenGL ハードウェアレンダリングが利用可能であることを確認するには

次のコマンドを使用して、X サーバーが実行されていることを確認します。

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

以下に示しているのは、OpenGL ハードウェアレンダリングが利用可能な場合の出力例です。

```
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
```

### AMD ドライバーをインストールして設定する

G4ad インスタンスGPUなどの AMD がアタッチされたインスタンスには、適切なAMDドライバーがインストールされている必要があります。互換性のある Amazon EC2インスタンスにAMDGPUドライバーをインストールする方法については、[「Linux インスタンスにAMDドライバーをインストールする」](#)を参照してください。

Amazon EC2 G4ad インスタンスの詳細については、[新しい Amazon EC2 G4ad インスタンスのブログ記事「ディープダイブ」](#)を参照してください。

### 以外のGPUインスタンス用のXDummyドライバーをインストールする

#### トピック

- [XDummy ドライバーのインストールと設定](#)

### XDummy ドライバーのインストールと設定

専用の `glxinfo` が存在しない Linux サーバーでコンソールセッションを使用するにはGPU、Xdummy ドライバーがインストールされ、正しく設定されていることを確認します。XDummy ドライバーを使用すると、実数が存在しない場合に、X サーバーGPUを仮想フレームバッファで実行できます。

**Note**

- これは、仮想セッションを使用する場合は必須ではありません。
- XDummy ドライバーは、設定で定義された解像度のみをサポートできます。

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

XDummy ドライバーをインストールするには

次のコマンドを実行します。

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

## Ubuntu

XDummy ドライバーをインストールするには

次のコマンドを実行します。

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

## SUSE Linux Enterprise

XDummy ドライバーをインストールするには

次のコマンドを実行します。

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

Linux サーバーにXDummyドライバーをインストールしたら、 を更新しますxorg.conf。

xorg.conf XDummyで を設定するには

1. 任意のテキストエディタで /etc/X11/xorg.conf ファイルを開きます。
2. 設定に次のセクションを追加します。

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

```
Driver "dummy"
Option "UseEDID" "false"
VideoRam 512000
EndSection

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

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

#### Note

示されている設定は一例です。さらにモードを追加すれば、異なる virtual 解像度を設定できます。複数のダミーモニターを設定することもできます。

### 3. 変更を有効にするには、X サーバーを再起動します。

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

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

## Linux にNICE DCVサーバーをインストールする

NICE DCV サーバーは、ホストサーバーのオペレーティングシステムに応じて、一連の RPM または .deb パッケージを使用してインストールされます。このパッケージは、必要なパッケージとその依存関係をすべてインストールし、必要なサーバー設定を実行します。

**Note**

NICE DCV サーバーをインストールするには、ルートユーザーとしてサインインする必要があります。

## NICE DCV サーバーのインストール

## Amazon Linux 2 and RHEL/CentOS

NICE DCV サーバーは、64 ビット x86 および 64 ビット ARM アーキテクチャに基づく Amazon Linux 2 RHEL、および CentOS サーバーで使用できます。

**Important**

nice-dcv-gl および nice-dcv-gltest パッケージは、64 ビット ARM アーキテクチャに基づくサーバーでは使用できません。

Amazon Linux 2、RHEL および CentOS に NICE DCV サーバーをインストールするには

1. サーバーを起動して、サーバーをインストールする予定の NICE DCV サーバーに接続します。
2. NICE DCV サーバーパッケージは、安全な署名でデジタル GPG 署名されています。パッケージマネージャーがパッケージ署名を検証できるようにするには、NICE GPG キーをインポートする必要があります。そのためには、ターミナルウィンドウを開き、NICE GPG キーをインポートします。

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

3. ダウンロード [NICE ウェブサイト](#) から [パッケージ DCV](#) をダウンロードします。RPM および deb パッケージは .tgz アーカイブにパッケージ化されます。必ずオペレーティングシステムに適したアーカイブをダウンロードしてください。

- 64 ビット x86

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

- 64 ビット ARM

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

**i** Tip

ダウンロード用ウェブサイトの[最新のパッケージ](#)には、利用可能な最新バージョンに常に接続されているリンクが含まれます。これらのリンクを使用して、最新のNICE DCVパッケージを自動的に取得できます。

- 64 ビット x86

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

- 64 ビット ARM

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

4. .tgz アーカイブのコンテンツを抽出し、抽出されたディレクトリに移動します。

- 64 ビット x86

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

- 64 ビット ARM

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

5. NICE DCV サーバーをインストールします。

- 64 ビット x86

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

- 64 ビット ARM

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

- (オプション) NICE DCVバージョン 2021.2 以降でウェブクライアントを使用するには、`nice-dcv-web-viewer`パッケージをインストールします。

- 64 ビット x86

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

- 64 ビット ARM

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

- (オプション) 仮想セッションを使用する場合は、`nice-xdcv` パッケージをインストールします。

- 64 ビット x86

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

- 64 ビット ARM

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

- (オプション) GPU共有を使用する場合は、`nice-dcv-gl`パッケージをインストールします。

- 64 ビット x86

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

#### Note

オプションとして `nice-dcv-gltest` パッケージをインストールすることもできます。このパッケージには、仮想セッションがハードウェアベースの OpenGL を使用するように適切に設定されているかどうかを判断するために使用できるシンプルな OpenGL アプリケーションが含まれています。

- (オプション) NICE DCVで を使用する場合はNICE EnginFrame、`nice-dcv-simple-external-authenticator`パッケージをインストールします。

- 64 ビット x86

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

- 64 ビット ARM

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

10. (オプション) USB リモート化を使用して特殊なUSBデバイスをサポートするには、DCVUSBドライバーをインストールします。

DCV USB ドライバーをインストールするには、サーバーに動的カーネルモジュールサポート (DKMS) がインストールされている必要があります。をインストールするには、次のコマンドを使用しますDKMS。

DKMS は、Enterprise Linux 用追加パッケージ (EPEL) リポジトリからインストールできます。リポジトリを有効にするには、次のコマンドを実行しますEPEL。

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

EPEL リポジトリを有効にしたら、次のコマンドを実行して をインストールしますDKMS。

```
$ sudo yum install dkms
```

をインストールしたらDKMS、次のコマンドを実行してDCVUSBドライバーをインストールします。

```
$ sudo dcvusbdriverinstaller
```

11. (オプション) マイクのリダイレクトをサポートする予定がある場合は、pulseaudio-utils パッケージがシステムにインストールされていることを確認します。そのインストールには以下のコマンドを使用します。

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

## RHEL, CentOS, and Rocky Linux 8.5

NICE DCV サーバーは、RHEL、64 ビット x86 および 64 ビット ARM アーキテクチャに基づく CentOS サーバー、および Rocky Linux 8.5 以降で使用できます。

### Important

nice-dcv-gl および nice-dcv-gltest パッケージは、64 ビット ARM アーキテクチャに基づくサーバーでは使用できません。

NICE DCV サーバーを RHEL、CentOS、または Rocky Linux 8.5 にインストールするには

1. サーバーをインストールしようとしている NICE DCV サーバーを起動して接続します。
2. NICE DCV サーバーパッケージは、安全な署名でデジタル GPG 署名されています。パッケージ マネージャーがパッケージ署名を検証できるようにするには、NICE GPG キーをインポートする必要があります。そのためには、ターミナルウィンドウを開き、NICE GPG キーをインポートします。

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

3. ダウンロード [NICE ウェブサイト からパッケージ DCV](#) をダウンロードします。RPM および deb パッケージは .tgz アーカイブにパッケージ化されます。必ずオペレーティングシステムに適したアーカイブをダウンロードしてください。

- 64 ビット x86

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

- 64 ビット ARM

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

**i** Tip

ダウンロード用ウェブサイトの[最新のパッケージ](#)には、利用可能な最新バージョンに常に接続されているリンクが含まれます。これらのリンクを使用して、最新のNICE DCVパッケージを自動的に取得できます。

- 64 ビット x86

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

- 64 ビット ARM

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

4. .tgz アーカイブのコンテンツを抽出し、抽出されたディレクトリに移動します。

- 64 ビット x86

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

- 64 ビット ARM

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

5. NICE DCV サーバーをインストールします。

- 64 ビット x86

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

- 64 ビット ARM

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

6. (オプション) NICE DCVバージョン 2021.2 以降でウェブクライアントを使用する場合は、nice-dcv-web-viewerパッケージをインストールします。

- 64 ビット x86

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

- 64 ビット ARM

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

7. (オプション) 仮想セッションを使用する場合は、`nice-xdcv` パッケージをインストールします。

- 64 ビット x86

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

- 64 ビット ARM

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

8. (オプション) GPU共有を使用する場合は、`nice-dcv-gl`パッケージをインストールします。

- 64 ビット x86

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

#### Note

オプションとして `nice-dcv-gltest` パッケージをインストールすることもできます。このパッケージには、仮想セッションがハードウェアベースの OpenGL を使用するように適切に設定されているかどうかを判断するために使用できるシンプルな OpenGL アプリケーションが含まれています。

9. (オプション) NICE DCV を使用する場合は NICE EnginFrame、`nice-dcv-simple-external-authenticator` パッケージをインストールします。

- 64 ビット x86

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

- 64 ビット ARM

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

10. (オプション) USB リモート化を使用して特殊なUSBデバイスをサポートする場合は、DCVUSBドライバーをインストールします。

DCV USB ドライバーをインストールするには、サーバーに動的カーネルモジュールサポート (DKMS) がインストールされている必要があります。をインストールするには、次のコマンドを使用しますDKMS。

DKMS は、Enterprise Linux 用追加パッケージ (EPEL) リポジトリからインストールできます。リポジトリを有効にするには、次のコマンドを実行しますEPEL。

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

EPEL リポジトリを有効にしたら、次のコマンドを実行して をインストールしますDKMS。

```
$ sudo yum install dkms
```

をインストールしたらDKMS、次のコマンドを実行してDCVUSBドライバーをインストールします。

```
$ sudo dcvusbdriverinstaller
```

11. (オプション) マイクのリダイレクトをサポートする予定がある場合は、pulseaudio-utils パッケージがシステムにインストールされていることを確認します。そのインストールには以下のコマンドを使用します。

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

## RHEL, CentOS, and Rocky Linux 9

NICE DCV サーバーは、RHEL、64 ビット x86 および 64 ビット ARM アーキテクチャに基づく CentOS サーバー、および Rocky Linux 9 以降で使用できます。

### Important

`nice-dcv-gl` および `nice-dcv-gltest` パッケージは、64 ビット ARM アーキテクチャに基づくサーバーでは使用できません。

NICE DCV サーバーを RHEL、CentOS、または Rocky Linux 9 にインストールするには

1. サーバーをインストールしようとしている NICE DCV サーバーを起動して接続します。
2. NICE DCV サーバーパッケージは、安全な署名でデジタル GPG 署名されています。パッケージ マネージャーがパッケージ署名を検証できるようにするには、NICE GPG キーをインポートする必要があります。そのためには、ターミナルウィンドウを開き、NICE GPG キーをインポートします。

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

3. ダウンロード [NICE ウェブサイト からパッケージ DCV](#) をダウンロードします。RPM および deb パッケージは .tgz アーカイブにパッケージ化されます。必ずオペレーティングシステムに適したアーカイブをダウンロードしてください。

- 64 ビット x86

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

- 64 ビット ARM

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2023.1/Servers/nice-dcv-2023.1-16388-e19-aarch64.tgz
```

**i** Tip

ダウンロード用ウェブサイトの[最新のパッケージ](#)には、利用可能な最新バージョンに常に接続されているリンクが含まれます。これらのリンクを使用して、最新のNICE DCVパッケージを自動的に取得できます。

- 64 ビット x86

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

- 64 ビット ARM

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

4. .tgz アーカイブのコンテンツを抽出し、抽出されたディレクトリに移動します。

- 64 ビット x86

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

- 64 ビット ARM

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

5. NICE DCV サーバーをインストールします。

- 64 ビット x86

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

- 64 ビット ARM

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

6. (オプション) NICE DCVバージョン 2021.2 以降でウェブクライアントを使用する場合は、`nice-dcv-web-viewer`パッケージをインストールします。

- 64 ビット x86

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

- 64 ビット ARM

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

7. (オプション) 仮想セッションを使用する場合は、`nice-xdcv` パッケージをインストールします。

- 64 ビット x86

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

- 64 ビット ARM

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

8. (オプション) GPU共有を使用する場合は、`nice-dcv-gl`パッケージをインストールします。

- 64 ビット x86

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

#### Note

オプションとして `nice-dcv-gltest` パッケージをインストールすることもできます。このパッケージには、仮想セッションがハードウェアベースの OpenGL を使用するように適切に設定されているかどうかを判断するために使用できるシンプルな OpenGL アプリケーションが含まれています。

9. (オプション) NICE DCV を使用する予定がある場合は `NICE EnginFrame`、`nice-dcv-simple-external-authenticator` パッケージをインストールします。

- 64 ビット x86

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

- 64 ビット ARM

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

10. (オプション) USB リモート化を使用して特殊なUSBデバイスをサポートする場合は、DCVUSBドライバをインストールします。

DCV USB ドライバをインストールするには、サーバーに動的カーネルモジュールサポート (DKMS) がインストールされている必要があります。をインストールするには、次のコマンドを使用しますDKMS。

DKMS は、Enterprise Linux 用追加パッケージ (EPEL) リポジトリからインストールできます。リポジトリを有効にするには、次のコマンドを実行しますEPEL。

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

EPEL リポジトリを有効にしたら、次のコマンドを実行して をインストールしますDKMS。

```
$ sudo yum install dkms
```

をインストールしたらDKMS、次のコマンドを実行してDCVUSBドライバをインストールします。

```
$ sudo dcvusbdriverinstaller
```

11. (オプション) マイクのリダイレクトをサポートする予定がある場合は、pulseaudio-utils パッケージがシステムにインストールされていることを確認します。そのインストールには以下のコマンドを使用します。

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

## SLES 12.x/15.x

NICE DCV サーバーは、64 ビット x86 アーキテクチャのみに基づく SUSE Linux Enterprise Server (SLES) 12.x/15.x サーバーで使用できます。

NICE DCV サーバーを 12SLES.x/15.x にインストールするには

1. サーバーをインストールしようとしているNICE DCVサーバーを起動して接続します。
2. NICE DCV サーバーパッケージは、安全な署名でデジタルGPG署名されています。パッケージマネージャーがパッケージ署名を検証できるようにするには、NICE GPG キーをインポートする必要があります。そのためには、ターミナルウィンドウを開き、NICE GPG キーをインポートします。

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

3. ダウンロード[NICEウェブサイトからパッケージDCV](#)をダウンロードします。RPM および deb パッケージは .tgz アーカイブにパッケージ化されます。必ずオペレーティングシステムに適したアーカイブをダウンロードしてください。

- SLES 12.x

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

- SLES 15.x

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

**i** Tip

ダウンロード用ウェブサイトの[最新のパッケージ](#)には、利用可能な最新バージョンに常に接続されているリンクが含まれます。これらのリンクを使用して、最新のNICE DCVパッケージを自動的に取得できます。

- 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. .tgz アーカイブのコンテンツを抽出し、抽出されたディレクトリに移動します。

- SLES 12.x

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

- SLES 15.x

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

5. NICE DCV サーバーをインストールします。

- SLES 12.x

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

- SLES 15.x

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

6. (オプション) NICE DCV バージョン 2021.2 以降でウェブクライアントを使用する場合は、`nice-dcv-web-viewer` パッケージをインストールします。

- SLES 12.x

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

- SLES 15.x

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

7. (オプション) 仮想セッションを使用する場合は、`nice-xdcv` パッケージをインストールします。

- SLES 12.x

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

- SLES 15.x

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

8. (オプション) GPU共有を使用する場合は、`nice-dcv-gl`パッケージをインストールします。

- SLES 12.x

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

- SLES 15.x

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

#### Note

オプションとして `nice-dcv-gltest` パッケージをインストールすることもできます。このパッケージには、仮想セッションがハードウェアベースの OpenGL を使用するように適切に設定されているかどうかを判断するために使用できるシンプルな OpenGL アプリケーションが含まれています。

9. (オプション) NICE DCV を使用する予定がある場合は NICE EnginFrame、`nice-dcv-simple-external-authenticator` パッケージをインストールします。

- SLES 12.x

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

- SLES 15.x

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

10. (オプション) USB リモート化を使用して特殊なUSBデバイスをサポートする場合は、DCVUSBドライバーをインストールします。

DCV USB ドライバーをインストールするには、サーバーに動的カーネルモジュールサポート (DKMS) がインストールされている必要があります。をインストールするには、次のコマンドを使用しますDKMS。

をインストールするには、次のコマンドを実行します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

PackageHub リポジトリを有効にします。

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

 Note

SLES 15 SP1または を使用している場合はSP2、**15** 上記のコマンドので、**15.1**または のいずれかを使用します**15.2**。

をインストールしますDKMS。

```
$ sudo zypper refresh
```

```
$ sudo zypper install dkms
```

カーネルソースをインストールします。

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

インスタンスを再起動します。

```
$ sudo reboot
```

をインストールしたらDKMS、次のコマンドを実行してDCVUSBドライバーをインストールします。

```
$ sudo dcvusbdriverinstaller
```

11. (オプション) マイクのリダイレクトをサポートする予定がある場合は、pulseaudio-utils パッケージがシステムにインストールされていることを確認します。そのインストールには以下のコマンドを使用します。

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

## Ubuntu 20.04/22.04

NICE DCV サーバーは、64 ビット x86 および 64 ビットARMアーキテクチャに基づく Ubuntu サーバーで使用できます。

### Important

nice-dcv-gl および nice-dcv-gltestパッケージは、64 ビットARMアーキテクチャに基づくサーバーでは使用できません。

Ubuntu 20.04/22.04 にNICE DCVサーバーをインストールするには

1. サーバーをインストールしようとしているNICE DCVサーバーを起動して接続します。
2. NICE DCV サーバーパッケージは、安全な署名でデジタルGPG署名されています。パッケージマネージャーがパッケージ署名を検証できるようにするには、NICE GPG キーをインポートする必要があります。そのためには、ターミナルウィンドウを開き、NICE GPGキーをインポートします。

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

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

- ダウンロード [NICEウェブサイト](#) から [パッケージDCV](#) をダウンロードします。RPM および deb パッケージは .tgz アーカイブにパッケージ化されます。必ずオペレーティングシステムに適したアーカイブをダウンロードしてください。

- Ubuntu 20.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット ARM )

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

#### Tip

ダウンロード用ウェブサイトの [最新のパッケージ](#) には、利用可能な最新バージョンに常に接続されているリンクが含まれます。これらのリンクを使用して、最新の NICE DCV パッケージを自動的に取得できます。

- Ubuntu 20.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット ARM )

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

4. .tgz アーカイブのコンテンツを抽出し、抽出されたディレクトリに移動します。

- Ubuntu 20.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット ARM )

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

5. NICE DCV サーバーをインストールします。

- Ubuntu 20.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット ARM )

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

6. ( オプション ) NICE DCV バージョン 2021.2 以降でウェブクライアントを使用する場合は、`nice-dcv-web-viewer` パッケージをインストールします。

- Ubuntu 20.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット ARM )

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

7. dcv ユーザーを video グループに追加します。

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

8. (オプション) 仮想セッションを使用する予定がある場合は、nice-xdcv パッケージをインストールします。

- Ubuntu 20.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット ARM )

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

9. (オプション) GPU共有を使用する場合は、 nice-dcv-glパッケージをインストールします。

- Ubuntu 22.04 (64 ビット x86)

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

#### Note

オプションとして nice-dcv-glttest パッケージをインストールすることもできます。このパッケージには、仮想セッションがハードウェアベースの OpenGL を使用

するように適切に設定されているかどうかを判断するために使用できるシンプルな OpenGL アプリケーションが含まれています。

10. ( オプション) NICE DCVで を使用する予定がある場合はNICE EnginFrame、 nice-dcv-simple-external-authenticatorパッケージをインストールします。

- Ubuntu 20.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット x86)

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

- Ubuntu 22.04 (64 ビット ARM )

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

11. ( オプション) USB リモート化を使用して特殊なUSBデバイスをサポートする場合は、DCVUSBドライバーをインストールします。

DCV USB ドライバーをインストールするには、サーバーに動的カーネルモジュールサポート (DKMS) がインストールされている必要があります。をインストールするには、次のコマンドを使用しますDKMS。

DKMS は、公式の Ubuntu リポジトリで利用できます。をインストールするには、次のコマンドを実行しますDKMS。

```
$ sudo apt install dkms
```

をインストールしたらDKMS、次のコマンドを実行してDCVUSBドライバーをインストールします。

```
$ sudo dcvusbdriverinstaller
```

12. (オプション) マイクのリダイレクトをサポートする予定がある場合は、pulseaudio-utils パッケージがシステムにインストールされていることを確認します。そのインストールには以下のコマンドを使用します。

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

13. (オプション) Ubuntu 22.04 でのみ、システム認証SSOで を使用する必要がある場合は、次のコマンドを使用してインストールします。

```
$ nice-dcv-gnome-shell-extension_version_all.ubuntu2204
```

## インストール後のチェック

このトピックでは、NICE DCVサーバーが正しく設定されていることを確認するNICE DCVのために、インストール後に実行する必要があるインストール後のチェックについて説明します。

### 内容

- [NICE DCV サーバーが到達可能であることを確認する](#)
- [Xサーバーがアクセス可能であることを確認する](#)
- [GL DCV が正しくインストールされていることを確認します。](#)
- [NICE DCV DEB パッケージの署名を検証する](#)

### NICE DCV サーバーが到達可能であることを確認する

デフォルトでは、NICE DCVサーバーはTCPポート 8443 経由で通信するように設定されています。サーバーがこのポートを介して到達可能であることを確認します。ポート 8443 経由のアクセスを禁止するファイアウォールがある場合は、NICE DCVサーバーが通信するポートを変更する必要があります。詳細については、「[NICE DCV サーバーTCP/UDPポートとリッスンアドレスの変更](#)」を参照してください。

また、EC2インスタンスNICE DCVで を設定する場合は、セキュリティグループを作成します。これは、NICE DCVサーバーが通信するポートへのアクセスを有効にするためです。詳細については、「[」でセキュリティグループを設定する方法EC2](#)を参照してください。

### Xサーバーがアクセス可能であることを確認する

NICE DCV コンソールセッションと仮想セッションが Xサーバーにアクセスできることを確認する必要があります。

## コンソールセッション

NICE DCV サーバーをインストールすると、dcvユーザーが作成されます。このユーザーが X サーバーにアクセスできることを確認します。

**dcv ユーザーが X サーバーにアクセスできることを確認するには**

次のコマンドを実行します。

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

コマンドが SI:localuser:dcv を返した場合、dcv ユーザーは X サーバーにアクセスできます。

コマンドが SI:localuser:dcv を返さない場合、dcv ユーザーは X サーバーにアクセスできません。以下のコマンドを実行して X サーバーを再起動します。

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

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

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

## 仮想セッション

DCV GL パッケージをインストールした場合は、ローカルユーザーが X サーバーにアクセスできることを確認する必要があります。これにより、OpenGL ハードウェアアクセラレーションが仮想セッションで正しく機能するようになります。

**ローカルユーザーが X サーバーにアクセスできることを確認するには**

次のコマンドを実行します。

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

コマンドが LOCAL: を返した場合、ローカルユーザーは X サーバーにアクセスできます。

コマンドが LOCAL: を返さない場合、ローカルユーザーは X サーバーにアクセスできません。次のコマンドを実行して X サーバーを再起動し、GL DCV を無効または再度有効にします。

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

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

```
$ sudo dcvgladmin disable
```

```
$ sudo dcvgladmin enable
```

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

GL DCV が正しくインストールされていることを確認します。

dcvgldiag ユーティリティは、GL DCV パッケージをインストールすると自動的にインストールされます。このユーティリティを使用して、Linux サーバー設定が GL DCV 要件を満たしていることを確認できます。

dcvgldiag ユーティリティを実行するには

以下のコマンドを使用します。

```
$ sudo dcvgldiag
```

このユーティリティは、警告とエラーのリストと共に、考えられる解決策を返します。

NICE DCV DEB パッケージの署名を検証する

NICE DCV がインストールされたら、Debian パッケージ () の署名を確認できますDEB。この検証プロセスでは、GPGバージョン 1 を使用する必要があります。

DEB パッケージの署名を確認するには

以下のコマンドを使用します。

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

このコマンドから、署名が検証されていることを確認するための用語 GOODSIG を含むメッセージが返されます。次の例は署名確認メッセージを示しています。の代わりに *Example Key*、キーが表示されます。

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

## NICE DCV サーバーのライセンス

NICE DCV ライセンス要件は、NICE DCVサーバーのインストール場所と使用場所によって異なります。

### Important

次のライセンス要件は、NICE DCVバージョン 2017.0 以降にのみ適用されます。

## NICE DCV ライセンス要件

### トピック

- [NICE DCV Amazon での EC2](#)
- [のその他のユースケース NICE DCV](#)
- [Windows Server へのリモートアクセスに関する Microsoft ライセンス要件](#)

## NICE DCV Amazon での EC2

AWS Outposts および AWS Local Zones で実行されているEC2インスタンスを含め、インスタンスにサーバーをインストールして使用するためにライセンスNICE DCVサーバーは必要ありません。NICE DCV サーバーは Amazon EC2インスタンスで実行されていることを自動的に検出し、定期的に S3 バケットに接続して、有効なライセンスが使用可能かどうかを判断します。

インスタンスに以下の特性があることを確認してください。

- Amazon S3 エンドポイントに到達できる。インターネットにアクセスできる場合は、Amazon S3 パブリックエンドポイントを使用して接続します。インスタンスがインターネットにアクセスできない場合は、経路で Amazon S3 に到達できるようにするVPCアウトバウンドセキュリティグループルールまたはアクセスコントロールリスト (ACL) ポリシーを使用して、のゲートウェイエンドポイントを設定しますHTTPS。Amazon S3 詳細については、「Amazon VPCユーザーガイド」の「[ゲートウェイVPCエンドポイント](#)」を参照してください。S3 バケットへの接続に問題がある場合は、AWS ナレッジセンターの「[ゲートウェイVPCエンドポイントを使用して S3 バケットに接続できないのはなぜですか？](#)」を参照してください。

- 必要な Amazon S3 オブジェクトにアクセスするためのアクセス許可がある。次の Amazon S3 アクセスポリシーをインスタンスのIAMロールに追加し、*region* AWS リージョンの プレースホルダー (例: us-east-1)。詳細については、[IAM「ロールの作成」](#)を参照してください。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::dcv-license.region/*"
    }
  ]
}
```

- Windows インスタンスを使用している場合は、インスタンスがインスタンスメタデータサービスにアクセスできることを確認します。NICE DCV サーバーが適切にライセンスされるようにするには、このサービスへのアクセスが必要です。インスタンスメタデータサービスの詳細については、「Amazon [ユーザーガイド](#)」の「[インスタンスメタデータとユーザーデータ](#)」を参照してください。 EC2

カスタム Windows を使用している場合はAMI、EC2Configサービス (Windows Server 2012 R2 以前) または EC2Launch (Windows Server 2016 以降) のいずれかをインストールする必要があります。これにより、インスタンスはインスタンスメタデータサービスにアクセスできるようになります。詳細については、「Amazon [ユーザーガイド](#)」の「[EC2Configサービスを使用した Windows インスタンスの設定](#)」または「[を使用した Windows インスタンスの設定EC2Launch](#)」を参照してください。 EC2

Amazon EC2インスタンスにNICE DCVサーバーをインストールして使用する場合は、この章の残りの部分はスキップできます。この章の残りの部分は、NICE DCVサーバーの他のすべてのユースケースにのみ適用されます。

## その他のユースケース NICE DCV

その他のすべてのユースケースでは、NICE DCVサーバーをインストールして使用するにはライセンスが必要です。以下のライセンスオプションが利用できます。

- 自動評価ライセンス — このタイプのライセンスは、NICE DCVサーバーのインストール時に自動的にインストールされます。このタイプのライセンスは、インストール後 30 日間有効です。ライセンスの有効期限が切れると、サーバーでNICE DCVセッションを作成してホストできなくなりま

す。これらのライセンスは短期のテストと評価に適しています。長期間テストするには、拡張評価ライセンスをリクエストしてください。

#### Note

他のライセンスが設定されていない場合、NICE DCVサーバーはデフォルトで自動評価ライセンスになります。

- 拡張評価ライセンス — 拡張評価ライセンスは、自動評価ライセンスによって提供される最初の 30 日間の評価期間を延長する評価ライセンスです。期間は NICEによって case-by-case 決定されます。拡張評価ライセンスは有効期限に達した後に無効になり、サーバーでNICE DCVセッションを作成してホストできなくなります。延長評価ライセンスは、NICEウェブサイトの「[購入方法](#)」ページに記載されているNICEディストリビューターまたはリセラーにリクエストする必要があります。ライセンスは、NICE DCVサーバーにインストールする必要があるライセンスファイルとして提供されます。
- 本番ライセンス — 本番ライセンスは、 から購入するフルライセンスですNICE。実稼働ライセンスは、ライセンスサーバーによって管理されるフローティングライセンスです。フローティングライセンスを使用すると、ネットワーク内で複数のNICE DCVサーバーを実行できます。同時に、すべてのサーバーで作成できる同時NICE DCVセッションの数を制限することもできます。同時NICE DCVセッションごとに 1 つのライセンスが必要です。本番ライセンスはライセンスファイルとして配布され、Reprise License Manager (RLM) サーバーにインストールする必要があります。実稼働ライセンスには、次の 2 種類があります。
- 永久ライセンス — 永久ライセンスは有効期限がなく、無期限に使用できます。
- サブスクリプション — サブスクリプションは限られた期間 (通常は 1 年) 有効です。ライセンスの有効期限は、ライセンスファイルに示されます。ライセンスの有効期限が切れると、NICE DCVサーバーでNICE DCVセッションを作成してホストできなくなります。

NICE DCV 永続的ライセンスまたはサブスクリプションを購入する方法については、NICEウェブサイトで[購入する方法](#)と、お住まいのリージョンのNICEディストリビューターまたはリセラーを見つける方法を参照してください。

#### ライセンス要件

- NICE DCV クライアントにはライセンスは必要ありません。
- NICE DCV サーバーライセンスファイルは、NICE DCVサーバーの以前のバージョンと下位互換性があります。例えば、NICE DCVサーバーバージョン 2021 ライセンスをNICE DCVサーバーバージョン 2019 で使用できます。

- NICE DCV サーババージョンには、少なくともサーバNICE DCVライセンスの同じバージョンが必要です。例えば、NICE DCVサーババージョン 2021 を使用する場合は、ライセンスバージョン 2021 以降が必要です。新しいNICE DCVサーババージョンにアップグレードする場合は、互換性のあるライセンスファイルをリクエストする必要があります。詳細については、NICE DCVディストリビューターまたはリセラーにお問い合わせください。

#### Note

NICE DCV サーバの互換性については、「」を参照してください [互換性に関する考慮事項](#)。

## Windows Server へのリモートアクセスに関する Microsoft ライセンス要件

Microsoft では、Windows Server クライアントアクセスライセンス (CAL) に加えて、サーバのグラフィカルユーザーインターフェイス (GUI) CALにリモートアクセスするユーザーごとに、Windows Server のリモートデスクトップサービス (RDS) を Windows Server のバージョン用に用意する必要があります。これは、使用するリモートディスプレイプロトコルとは無関係です。このライセンスは、NICE DCVを使用してリモート Windows Server ホストGUIの GUIにアクセスする場合にも必要です。

Amazon EC2インスタンスでNICE DCVサーバを実行し、[Windows Server を使用する場](#)  
[合、Amazon AMI](#)は Windows Server のライセンスコストを処理しCAL、管理のみRDS CALsを目的とした2つの Windows Server を提供します。その用途は、テスト、メンテナンス、管理に限定されます。

詳細については、[マイクロソフト製品条項サイト](#)を参照してください。お使いの Microsoft ソフトウェアのライセンスや権限に関してご質問がございましたら、御社内の法務部門、Microsoft、または Microsoft 再販業者にお問い合わせください。

## 拡張評価ライセンスのインストール

に拡張評価ライセンスをリクエストするとNICE、ライセンスを定義するlicense.licファイルを受け取ります。

拡張評価ライセンスをインストールするには

license.lic ファイルをサーバの次のフォルダに配置します。

- Windows Server

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

- [Linux サーバー]

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

または、license.lic をサーバーの別のフォルダに配置するには、ライセンスファイルのフルパスが指定されるように license-file 設定パラメータを更新する必要があります。

### トピック

- [Windows サーバーでのライセンスパスの変更](#)
- [Linux サーバーでのライセンスパスの変更](#)

## Windows サーバーでのライセンスパスの変更

Windows サーバーの **license-file** 設定パラメータを更新するには

1. Windows レジストリエディタを開きます。
2. HKEY\_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/license/ キーに移動し、license-file パラメータを選択します。

レジストリキーに license-file パラメータが存在しない場合は、1 つ作成します。

- a. 左のパネルで、[ライセンス] キーのコンテキスト (右クリック) メニューを開き、[新規]、[文字列値] の順に選択します。
  - b. [名前] に license-file と入力し、[Enter] を押します。
3. [license-file] パラメータを開きます。[値のデータ] に、license.lic ファイルへのフルパスを入力します。
  4. [OK] を選択して Windows レジストリエディタを閉じます。

## Linux サーバーでのライセンスパスの変更

Linux サーバーの **license-file** 設定パラメータを更新するには

1. /etc/dcv/ に移動し、任意のテキストエディタで `dcv.conf` を開きます。
2. `license-file` セクションで `[license]` パラメータを見つけ、既存のパスを `license.lic` ファイルへの新しいフルパスに置き換えます。

`license-file` セクションに `[license]` パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

3. ファイルを保存して閉じます。

## 実稼働ライセンスのインストール

このトピックの次のセクションでは、実稼働ライセンス (永久ライセンスまたはサブスクリプション) を購入して使用方法について説明します。

### トピック

- [ステップ 1: RLMサーバーをインストールする](#)
- [ステップ 2: RLMサーバーのホスト ID を取得する](#)
- [ステップ 3: 永久ライセンスまたはサブスクリプションを購入する](#)
- [ステップ 4: ライセンスファイルに変更を加える](#)
- [ステップ 5: RLMサーバーを設定する](#)
- [ステップ 6: NICE DCVサーバーを設定する](#)

### ステップ 1: RLMサーバーをインストールする

永久ライセンスまたはサブスクリプションを購入すると、ライセンスの条件を定義するライセンスファイルが取得されます。Reprise License Manager (RLM) サーバーにライセンスファイルをインストールする必要があります。

の詳細についてはRLM、[Reprise Software](#) のウェブサイトを参照してください。

### トピック

- [Windows にRLMサーバーをインストールする](#)
- [Linux にRLMサーバーをインストールする](#)

Windows にRLMサーバーをインストールする

Windows にRLMサーバーをインストールするには

1. [Reprise Software ウェブサイト](#) からRLMライセンス管理バンドルをダウンロードします。

**Note**

NICE DCV バージョン 2022.1 以降では、RLMサーバー  $\geq 14$  が必要です。以前のバージョンでは RLM  $\geq 12$  が必要です。  
RLM ライセンス管理バンドルの最新の安定バージョンをインストールすることをお勧めします。

2. RLM ライセンス管理バンドルを にインストールしますC:\RLM。

Linux にRLMサーバーをインストールする

Linux にRLMサーバーをインストールするには

1. [Reprise Software ウェブサイト](#) からRLMライセンス管理バンドルをダウンロードします。

**Note**

NICE DCV バージョン 2022.1 以降では、RLMサーバー  $\geq 14$  が必要です。以前のバージョンでは RLM  $\geq 12$  が必要です。  
RLM ライセンス管理バンドルの最新の安定バージョンをインストールすることをお勧めします。

2. ユーザーグループと rlm ユーザーを作成します。これは、任意の有効なユーザーまたはサービスアカウントです。この値にはルートアカウントを使用しないことを強くお勧めします。

```
$ groupadd -r rlm
```

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

3. RLM サーバーに必要な `/opt/nice/rlm` および `/opt/nice/rlm/license` ディレクトリを作成します。

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

4. RLM ライセンス管理バンドルの内容を に抽出し `/opt/nice/rlm/`、ファイルが `rlm` ユーザーによって所有されていることを確認します。

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

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

## ステップ 2: RLMサーバーのホスト ID を取得する

RLM サーバーをインストールしたら、RLMサーバーのホスト ID を取得する必要があります。永久ライセンスまたはサブスクリプションを購入するときは、このホスト ID を指定する必要があります。

### Windows で RLMサーバーホスト ID を取得する

サーバーのホスト ID を取得するには、コマンドプロンプトを開き、

`C:\RLM\` に移動して、以下のコマンドを実行します。

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

コマンドは、RLMサーバーのホスト ID を次のように返します。

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

ホスト ID を書き留めます。これは次のステップで必要になります。

### Linux で RLMサーバーホスト ID を取得する

サーバーのホスト ID を取得するには (単一の戻り値)

1. `/opt/nice/rlm/` に移動します。
2. 次のコマンドを実行します。

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

コマンドは、次のように各ネットワークインターフェイスのRLMサーバーのホスト ID を返します。

3. ホスト ID を記録します。これは次のステップで必要になります。

### Example

このプロシージャが実行され、次の単一の ID が返されました。

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

その後、この ID が記録され、 のライセンスの購入に使用されますDCV。

サーバーのホスト ID を取得するには (複数の戻り値)。

1. /opt/nice/rlm/ に移動します。
2. 次のコマンドを実行します。

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

のリストに複数の が返されIDsますIDs。

3. 以下のコマンドを実行します。

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

コマンドは、ゲートウェイネットワークインターフェイスのRLMサーバーのホスト ID を返します。

4. ホスト ID を記録します。これは次のステップで必要になります。

### Example

プロシージャが実行され、複数の ID のリストに複数の IDsが返されました。

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

インターフェイスコマンドが実行され、次の ID が返されます。

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

その後、この ID が記録され、のライセンスの購入に使用されますDCV。

### ステップ 3: 永久ライセンスまたはサブスクリプションを購入する

NICE DCV 永続的ライセンスまたはサブスクリプションを購入する方法については、NICEウェブサイトでの[購入する方法](#)と、お住まいのリージョンのNICEディストリビューターまたはリセラーを見つける方法を参照してください。

RLM サーバーのホスト ID を指定する必要があります。ホスト ID は、NICE が提供するライセンスファイルに埋め込まれます。

### ステップ 4: ライセンスファイルに変更を加える

NICE DCV 永続的ライセンスまたはサブスクリプションを購入すると、ライセンスを定義するlicense.licファイルを受け取ります。license.lic ファイルには、次の情報が含まれます。

- RLM サーバーのホスト名。
- ライセンスの購入時に指定したRLMサーバーのホスト ID。
- RLM サーバーのTCPポート番号。デフォルト: 5053。
- ISV ポート番号。これは、RLMサーバーがNICE DCVライセンスリクエストをリッスンするオプションのポートです。指定しない場合、起動RLM時に によってランダムポートが選択されます。
- ライセンスの対象となるNICE DCV製品と、各製品の以下の詳細。
  - ライセンスの対象となるメジャーバージョン (2017 2017 NICE DCV製品の場合など)。
  - 有効期限。Permanent はライセンスの有効期限が切れていないことを示します。
  - 同時セッションの最大数 (例: サーバー上の同時セッションが 10 の場合は 10)。
  - ライセンスチェックサム。
  - ライセンス署名。

次のコードブロックは license.lic ファイルの形式を示しています。

```
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"
```

次のコードブロックは、ISVポートを省略したlicense.licファイルの例を示しています。ライセンスファイルには、dcv-gl DCVという2つのNICE製品のライセンスが含まれています。

```
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"
```

### license.lic ファイルを編集するには

1. 任意のテキストエディタでファイルを開きます。
2. RLM サーバーのホスト名とTCPポート番号を、で始まるファイルの最初の行に追加しますHOST。

#### Warning

- *RLM\_server\_host\_id* は、ライセンスの購入時に指定したホスト ID です。を編集することはできません *RLM\_server\_host\_id*.

3. (オプション) を追加してISV、で始まるファイルの行にISVポート番号を追加しますport=*port\_number*。このポートは、DCVサーバーとの通信を有効にするために必要です。

ISV ポートを指定しない場合は、を省略しますport=*port\_number*。ISV ポートを指定しない場合、起動RLMのたびにによってランダムなポートが使用されます。

#### Warning

ランダムに選択されたポートの使用を妨げるファイアウォール設定がある場合は、このポートを指定し、HOST行で指定されたRLMポートに加えて、有効にするようにファイアウォールを設定する必要があります。

4. ファイルを保存して閉じます。

**⚠ Warning**

ライセンスの他の部分を編集すると、ファイルの署名が破損し、ライセンスは無効になります。

## ステップ 5: RLMサーバーを設定する

ライセンスファイルを変更したら、RLMサーバーに配置してからRLMサービスを開始する必要があります。

### トピック

- [Windows で RLM Server を設定する](#)
- [Linux でRLMサーバーを設定する](#)

### Windows で RLM Server を設定する

Windows でRLMサーバーを設定するには

1. RLM サーバーに接続します。
2. 編集した license.lic ファイルを C:\RLM\license\ にコピーします。
3. NICE DCV サーバーからC:\Program Files\NICE\DCV\Server\license\nice.set ファイルをコピーし、RLMサーバーの C:\RLM\フォルダに配置します。
4. RLM サーバーを Windows サービスとしてインストールします。

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

RLM 起動オプションの詳細については、[「ソフトウェアライセンスマネージャーのリプライズ \(RLM\)」](#)の製品ページを参照してください。

5. RLM サーバーを起動します。

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

6. RLM サーバーが実行中であることを確認します。

- a. 任意のテキストエディタで C:\RLM\nice.dlog を開き、次の行が表示されることを確認します。

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

 Note

rlm.log ファイルの内容は、RLMサーバーのバージョンによって若干異なる場合があります。

- b. 以下のコマンドを実行します。

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

コマンドは、RLMサーバーに関する情報を返します。

## Linux でRLMサーバーを設定する

Linux でRLMサーバーを設定するには

1. 編集した license.lic ファイルを /opt/nice/rlm/license/ にコピーします。
2. NICE DCV サーバーから /usr/share/dcv/license/nice.set ファイルをコピーし、RLM サーバー /opt/nice/rlm に配置します。
3. RLM サーバーサービスを作成し、起動時に自動的に起動することを確認します。
  - a. dcv-rlm という名前のファイルを /opt/nice/rlm/ フォルダに作成します。

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

- b. 任意のテキストエディタを使用してこのファイルを開き、以下のスクリプトを追加します。ファイルを保存して閉じます。

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

### BEGIN INIT INFO
```

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

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

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

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

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

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

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

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

```
        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. スクリプトを実行可能にして /etc/init.d/ にコピーし、それを chkconfig ユーティリティに追加します。

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

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

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

4. RLM サーバーを起動します。

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

5. RLM サーバーが期待どおりに動作していることを確認します。任意のテキストエディタで var/log/rlm.log を開き、次の行が表示されることを確認します。

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

#### Note

rlm.log ファイルの内容は、RLMサーバーのバージョンによって若干異なる場合があります。

## ステップ 6: NICE DCVサーバーを設定する

NICE DCV サーバーを使用するように RLMサーバーを設定します。これを行うには、NICE DCVサーバーで `license-file` 設定パラメータを設定する必要があります。

`license-file` パラメータは、接続先の RLMサーバーの仕様で、形式で設定する必要があります `RLM_server_port@RLM_server`。RLMサーバーは、ホスト名または IP アドレスとして指定できます。明示的に設定されていない場合、RLMサーバーポートはデフォルトで 5053 です。

複数の RLMサーバーが使用されている場合は、Linux では `;`、Windows では `:` で区切って、複数の RLMサーバー仕様のリストを指定できます。その後、対応するサーバーと 1 つの接続が確立されるまで、RLMサーバーは各サーバーに順番に接続しようとします。これ

は、フェイルオーバーサーバーを使用してプライマリRLMサーバーに到達できない場合に引き継ぐ場合などに特に役立ちます。この場合、ライセンスを次の形式で指定できます:

*RLM\_primary\_server\_port@RLM\_primary\_server:RLM\_failover\_server\_port@RLM\_failover\_server*

#### Note

NICE DCV Server が Windows にインストールされている場合は、仕様のエントリを で区切る必要があります;。

## トピック

- [Windows NICE DCV Server の設定](#)
- [Linux NICE DCV サーバーの設定](#)

## Windows NICE DCV Server の設定

Windows サーバーの **license-file** 設定パラメータを設定するには

1. Windows レジストリエディタを開きます。
2. HKEY\_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/license/ キーに移動し、license-file パラメータを選択します。

レジストリキーに license-file パラメータが存在しない場合には作成する必要があります。

- a. 左のパネルで、[ライセンス] キーのコンテキスト (右クリック) メニューを開き、[新規]、[文字列値] の順に選択します。
  - b. [名前] に license-file と入力し、[Enter] を押します。
3. [license-file] パラメータを開きます。値データ には、RLMサーバーのポート番号とホスト名を *RLM\_server\_port@RLM\_server*形式で入力します。複数のRLMサーバーへの接続を設定する必要がある場合は、上記の注意事項を確認してください。
  4. [OK] を選択して Windows レジストリエディタを閉じます。

## Linux NICE DCV サーバーの設定

Linux サーバーの **license-file** 設定パラメータを設定するには

1. /etc/dcv/ に移動し、任意のテキストエディタで dcv.conf を開きます。

2. `license-file` パラメータを `[license]` セクションで見つけます。次に、既存のパスを `RLM_server_port@RLM_server` 形式の RLM サーバーのポートとホスト名に置き換えます。

`license-file` セクションに `[license]` パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

複数の RLM サーバーへの接続を設定する必要がある場合は、上記の注意事項を確認してください。

3. ファイルを保存して閉じます。

## 実稼働ライセンスの更新

NICE DCV サーバーは数分ごとに RLM サーバー上のライセンスをチェックします。RLM サーバーでライセンスが更新された場合、NICE DCV サーバーは実行中のセッションで使用されたライセンスを自動的に更新します。次の手順では、DCV ライセンスを更新する方法について詳しく説明します RLM。

RLM サーバーの DCV ライセンスを更新するには

1. 以前に [インストール](#) したライセンスファイルを更新します。Linux では `/opt/dcv/rlm/license/license.lic` に、Windows では `C:\RLM\license\license.lic` に保存されているはずですが。
2. `rlmutil rlmreread` コマンドを実行してライセンスファイルを強制的にリロードします。

RLM サーバーでライセンスが更新されると、NICE DCV サーバーは数分 (通常は 5 分以内) で新しいライセンスの使用を確認する必要があります。

NICE DCV バージョン 2021.0 以降では、次のコマンドを管理者として使用して、ライセンスをすぐに強制更新できます。

```
$ dcv reload-licenses
```

## NICE DCV サーバーのアップグレード

次のトピックでは、NICE DCV サーバーをアップグレードする方法について説明します。

## 内容

- [互換性に関する考慮事項](#)
- [Windows での NICE DCV Server のアップグレード](#)
- [Linux での NICE DCV サーバーのアップグレード](#)

## 互換性に関する考慮事項

NICE DCV サーバーバージョン 2017 以降は、NICE DCV クライアントバージョン 2017 以降と互換性があります。

### Note

オンプレミス NICE DCV サーバーと ベース以外のサーバーのサーバーライセンスの互換性要件については、EC2「」を参照してください[ライセンス要件](#)。

## Windows での NICE DCV Server のアップグレード

Windows で NICE DCV サーバーをアップグレードするには

1. RDP クライアントを使用して、管理者として NICE DCV サーバーに接続します。
2. 実行中の NICE DCV セッションがないことを確認します。 `dcv list-sessions NICE DCV` コマンドを使用して、実行中のセッションがないか確認します。実行中のセッションがある場合は、`dcv close-session NICE DCV` コマンドを使用してセッションを停止します。
3. 実行中のセッションがないことを確認したら、NICE DCV サーバーを停止します。詳細については、「[NICE DCV サーバーの停止](#)」を参照してください。
4. NICE DCV サーバー設定をバックアップします。レジストリエディタを開き、`HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv` に移動し、`dcv` キーを右クリックして、エクスポートを選択します。
5. [NICE](#) 最新バージョンの NICE DCV サーバーをウェブサイトからダウンロードします。
6. ステップ 3 から始めて、[ウィザードの使用](#) で説明されているステップに従います。
7. インストールが完了したら、NICE DCV サーバー設定が正しいことを確認します。レジストリエディタを開き、`HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv` に移動し、パラメータをステップ 4 でエクスポートした設定と比較します。

8. 新しいNICE DCVセッションを開始してNICE DCVサーバーをテストします。詳細については、「[NICE DCV セッションの開始](#)」を参照してください。

## Linux でのNICE DCVサーバーのアップグレード

Linux でNICE DCVサーバーをアップグレードするには

1. を使用してSSH、root ユーザーを使用してサーバーにサインインします。
2. 実行中のNICE DCVセッションがないことを確認します。dcv list-sessions NICE DCV コマンドを使用して、実行中のセッションがないか確認します。実行中のセッションがある場合は、dcv close-session NICE DCV コマンドを使用してセッションを停止します。
3. 実行中のセッションがないことを確認したら、NICE DCVサーバーを停止します。詳細については、「[NICE DCV サーバーの停止](#)」を参照してください。
4. NICE DCV サーバー設定をバックアップします。/etc/dcv/dcv.conf ファイルを安全な場所にコピーします。
5. [NICE DCV サーバーのインストール](#) に説明されているステップに従います。
6. インストールが完了したら、NICE DCVサーバー設定が正しいことを確認します。ステップ 4 でコピーしたファイルを開き、/etc/dcv/dcv.conf ファイルと比較します。
7. 新しいNICE DCVセッションを開始してNICE DCVサーバーをテストします。詳細については、「[NICE DCV セッションの開始](#)」を参照してください。

## NICE DCV サーバーのアンインストール

次のトピックでは、NICE DCVサーバーをアンインストールする方法について説明します。

内容

- [Windows での NICE DCV Server のアンインストール](#)
- [Linux での NICE DCV サーバーのアンインストール](#)

## Windows での NICE DCV Server のアンインストール

Windows でNICE DCVサーバーをアンインストールするには

1. RDP クライアントを使用して、管理者としてNICE DCVサーバーに接続します。

2. 実行中のNICE DCVセッションがないことを確認します。 `dcv list-sessions NICE DCV` コマンドを使用して、実行中のセッションがないか確認します。実行中のセッションがある場合は、 `dcv close session NICE DCV` コマンドを使用してセッションを停止します。
3. 実行中のセッションがないことを確認したら、NICE DCVサーバーを停止します。詳細については、「[NICE DCV サーバーの停止](#)」を参照してください。
4. Windows の [Settings] (設定) アプリケーションを開き、[Apps & Features] (アプリと機能) パネルに移動します。
5. NICE DCV サーバーを選択し、アンインストール を押します。
6. ( オプション) NICE DCVサーバーによって生成されたログファイルを削除することもできます。アンインストールが完了したら、 `C:\ProgramData\NICE\dcv\` に移動し、ログフォルダを削除します。

## Linux での NICE DCV サーバーのアンインストール

NICE DCV サーバーは、ホストサーバーのオペレーティングシステムに応じて、一連の RPM または .deb パッケージを使用してインストールされます。

### Note

NICE DCV サーバーをアンインストールするには、ルートユーザーとしてサインインする必要があります。

Linux でNICE DCVサーバーをアンインストールするには

1. 実行中のNICE DCVセッションがないことを確認します。 `dcv list-sessions NICE DCV` コマンドを使用して、実行中のセッションがないか確認します。実行中のセッションがある場合は、 `dcv close session NICE DCV` コマンドを使用してセッションを停止します。
2. 実行中のセッションがないことを確認したら、NICE DCVサーバーを停止します。詳細については、「[NICE DCV サーバーの停止](#)」を参照してください。
3. NICE DCV サーバーパッケージをアンインストールします。インストールの実行方法によっては、一部のパッケージがシステムにインストールされない場合があります、その場合はコマンドから省略できます。オプションパッケージのリストについては、「[Linux への NICE DCV サーバーのインストール](#)」を参照してください。

## Amazon Linux 2 and RHEL, CentOS

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

## SLES 12.x/15.x

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

## Ubuntu 22.04

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

4. (オプション) NICE DCVサーバーによって生成されたログファイルを削除することもできます。アンインストールが完了したら、[/var/log] に移動して [dcv] フォルダを削除します。

## イメージングNICE DCVサーバー

[Amazon EC2](#) インスタンスをカスタマイズしたら、これらの変更を [Amazon マシンイメージ](#) () としてキャプチャできますAMI。この機能を使用すると、1つの から複数のインスタンスを起動できます。AMIすべて同じ設定で、必要に応じて起動できます。高性能のリモートディスプレイプロトコルで安全にストリーミングする必要がある場合は、Amazon EC2インスタンスのイメージを取得する前にオペレーティングシステムNICE DCVに を追加できます。NICE DCV 設定はイメージに含まれているため、イメージレベルでビジネスユニットを分離したり、デプロイされたインスタンスに特定のDCV設定を設定したりできます。

例えば、1つの から複数の Amazon EC2インスタンスをデプロイする場合AMI、ローカルユーザーアカウントの自動コンソール作成を使用して、エンドユーザーにアクセスNICE DCV許可を委任できます。または、[NICE DCV Session Manager](#) などのブローカーを使用して、NICE DCVセッションの作成を大規模に管理することもできます。

の作成は、次の2つの方法のいずれかで実行NICE DCVAMIできます。

## NICE DCV イメージの構築

まず、システムに NICE DCV がインストールされている必要があります。そうでない場合は、システムが [でサポートされていることを確認しNICE DCV](#)、[インストール](#)手順に従ってください。NICE DCV をインストールして[設定したら](#)、インスタンス[AMI](#)の を取得します。

または、[Windows](#) または [Linux](#) のNICE DCV前提条件が満たされている場合は、Amazon が管理する Image Builder NICE DCVコンポーネントを実行してNICE、 をインストールして設定できますDCV。コンポーネントは、以下を実行して取得できます。

1. [Amazon EC2 Image Builder コンソール](#)内のコンポーネントページに移動します。
2. フィルター所有者ドロップダウンメニューを選択し、クイックスタート (Amazon マネージド) を選択します。
3. フィルターテキストボックスを使用して、 dcv-server-windowsまたは を検索しますdcv-server-linux。
4. コンポーネントのハイパーリンクを選択します。
5. NICE DCV コンポーネントページで、コンテンツセクションからコンポーネントコンテンツを取得します。
6. [AWS タスクオーケストレーターとエグゼキュター \(\)](#) を使用して、コンポーネントをインスタンスでローカルに実行します。AWSTOE

### Note

詳細については、[「 の開始方法 AWSTOE」](#)を参照してください。

コンポーネント内のパラメータの使用については、以下のセクションを参照してください。

## イメージパイプラインNICE DCVへの の追加

[EC2 Image Builder レシピ](#)は、新しいイメージを作成するための開始点として使用するベースイメージと、イメージをカスタマイズしてすべてが期待どおりに動作することを確認するために追加するコンポーネントのセットを定義します。このレシピ内で、パイプラインNICE DCV内の のインストールを自動化する dcv-server-windowsまたは dcv-server-linuxコンポーネントを選択します。これらのコンポーネントのいずれかを選択すると、要件を満たすようにパラメータを微調整できます。

**Note**

Linux の場合、すべての[前提条件](#)を満たす必要があります。これは、ベースAMIまたは前の Image Builder コンポーネントで実行できます。

## パラメータ

### Windows

- sessionOwner— 自動的に作成されたセッションのデフォルトの所有者を設定します。指定しない場合、コンソールの自動作成は無効になります。詳細については、NICE DCV 管理ガイドの「[自動コンソールセッションの有効化](#)」を参照してください。
- dcvPermissions— セッションのNICE DCVアクセス許可を設定します。詳細については、「[管理ガイド](#)」の「[アクセス許可ファイルの使用](#)」を参照してください。DCV

### Linux

- SessionOwner— 自動的に作成されたセッションのデフォルトの所有者を設定します。指定しない場合、コンソールの自動作成は無効になります。詳細については、NICE DCV 管理ガイドの「[自動コンソールセッションの有効化](#)」を参照してください。
- Packages— インストールされるNICE DCVパッケージを定義します。空の場合、使用可能なすべてのNICE DCVパッケージがインストールされます。詳細については、「[管理ガイド](#)」の「[Linux にNICE DCVサーバーをインストールする](#)」を参照してください。NICE DCV

コンポーネントを変更する場合は、[新しいコンポーネントバージョンを作成](#)できます。

# NICE DCV サーバーの管理

NICE DCV サーバーを起動、停止、または設定するには、管理者 (Windows) またはルート (Linux) としてサインインしている必要があります。

## トピック

- [NICE DCV サーバーの起動](#)
- [NICE DCV サーバーの停止](#)
- [QUIC UDP トランスポートプロトコルの有効化](#)
- [NICE DCV サーバーTCP/UDPポートとリッスンアドレスの変更](#)
- [TLS 証明書の管理](#)
- [アイドル状態のクライアントの切断](#)
- [Linux NICE DCVサーバーでGPUの共有の有効化](#)
- [タッチスクリーンとスタイラスのサポートの有効化](#)
- [ゲームパッドサポートの有効化](#)
- [USB リモート化の有効化](#)
- [スマートカードのキャッシュの設定](#)
- [WebAuthn リダイレクトの設定](#)
- [セッションストレージの有効化](#)
- [Linux NICE DCVサーバーでのプリンターの設定](#)
- [Linux NICE DCVサーバーでのクリップボードの設定](#)
- [マルチチャンネルオーディオの設定](#)
- [HTTP ヘッダーの設定](#)
- [NICE DCV 認証の設定](#)
- [NICE DCV 認証の設定](#)

## NICE DCV サーバーの起動

セッションをホストするには、NICE DCVサーバーが実行されている必要があります。

デフォルトでは、NICE DCVサーバーはホストされているサーバーが起動するたびに起動します。NICE DCV サーバーのインストール時に自動起動を無効にする場合は、サーバーを手動で起動

するか、自動起動を再度設定する必要があります。どちらのオプションを実行する場合も、次のいずれかの手順に従います。

## Windows NICE DCV server

Microsoft マネジメントコンソールのサービススナップインを使用して、NICE DCVサーバーを手動で起動します。

Windows でNICE DCVサーバーを起動するには

1. Microsoft 管理コンソールのサービススナップインを開きます。
2. 右側のペインで、DCVサーバー を開きます。
3. [開始] を選択します。

### Note

サーバーがすでに起動している場合、[Start] (開始) ボタンは無効になっています。

Microsoft マネジメントコンソールのサービススナップインを使用する自動起動を設定します。

Windows で自動的に起動するようにNICE DCVサーバーを設定するには

1. Microsoft 管理コンソールのサービススナップインを開きます。
2. 右側のペインで、DCVサーバー を開きます。
3. [スタートアップ サービス] で、[自動] を選択します。

## Linux NICE DCV server

コマンドラインを使用してNICE DCVサーバーを手動で起動します。

Linux でNICE DCVサーバーを起動するには

次のコマンドを使用します。

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

```
$ sudo systemctl start dcvserver
```

コマンドラインを使用して自動的に起動するようにNICE DCVサーバーを設定します。

Linux で自動的に起動するようにNICE DCVサーバーを設定するには

次のコマンドを使用します。

- RHEL、CentOS、SUSELinux Enterprise 12、および Ubuntu 18.x

```
$ sudo systemctl enable dcvserver
```

## NICE DCV サーバーの停止

NICE DCV サーバーはいつでも停止できます。サーバーを停止すると、すべてのアクティブなNICE DCVセッションが終了します。サーバーが再起動するまで新しいセッションを開始することはできません。

### Windows NICE DCV server

Microsoft マネジメントコンソールのサービススナップインを使用して、NICE DCVサーバーを手動で停止します。

Windows でNICE DCVサーバーを停止するには

1. Microsoft 管理コンソールのサービススナップインを開きます。
2. 右側のペインで、DCVサーバー を開きます。
3. [Stop] (停止) を選択します。

#### Note

サーバーがすでに停止している場合、[停止] ボタンは無効になっています。

Microsoft マネジメントコンソールのサービススナップインを使用する自動スタートアップを無効にします。

Windows でNICE DCVサーバーが自動的に起動しないようにするには

1. Microsoft 管理コンソールのサービススナップインを開きます。

2. 右側のペインで、DCVサーバーを開きます。
3. [スタートアップ サービス] で、[手動] を選択します。

## Linux NICE DCV server

コマンドラインを使用してNICE DCVサーバーを停止します。

Linux でNICE DCVサーバーを停止するには

以下のコマンドを使用します。

```
$ sudo systemctl stop dcvserver
```

コマンドラインを使用してNICE DCVサーバーの自動起動を無効にします。

Linux でNICE DCVサーバーが自動的に起動しないようにするには

以下のコマンドを使用します。

```
$ sudo systemctl disable dcvserver
```

## QUIC UDP トランスポートプロトコルの有効化

デフォルトでは、`quic` に基づくNICE DCV WebSocket プロトコルをデータ転送TCPに使用します。

データ転送に QUICプロトコルを使用するNICE DCVように `quic` を設定できます。このトランスポートプロトコルは `quic` に基づいていますUDP。ネットワークでレイテンシーとパケット損失が大きい場合、`quic` を使用するとパフォーマンスが向上QUICする可能性があります。`quic` を有効にするとQUIC、NICE DCVサーバーはデータ転送に QUICプロトコルを使用します。ただし、認証トラフィック WebSocket には引き続き `quic` を使用します。

### Note

`quic` は、ネットワークおよびセキュリティ設定でUDPトラフィックが許可されているQUIC場合にのみ使用できます。

`quic` を有効にするとQUIC、クライアントはNICE DCVサーバーセッションに接続するときに QUICプロトコルを使用してデータを転送できます。クライアントが接続時にQUICプロトコルを使用しない場

合は、を使用します WebSocket。QUIC プロトコルの詳細については、「[NICE DCV ユーザーガイド](#)」の [NICE DCV 「セッションへの接続」](#) を参照してください。

## Windows NICE DCV server

データ転送 NICE DCV に QUIC (UDP) を使用するようにを設定するには

1. Windows レジストリエディタを開き、HKEY\_USERS/S-1-5-18/Software/comGSettings/nicesoftware/dcv/connectivity/ キーに移動します。
2. enable-quic-frontend パラメータを開きます。[Value data] (値のデータ) に 1 を入力します。

### Note

パラメータが見つからない場合は、新しい DWORD (32 ビット) パラメータを作成し、という名前を付けます enable-quic-frontend。

3. Windows レジストリエディタを閉じます。
4. NICE DCV サーバー [を停止](#)して [再起動](#)します。

## Linux NICE DCV server

データ転送 NICE DCV に QUIC (UDP) を使用するようにを設定するには

1. 任意のテキストエディタで /etc/dcv/dcv.conf を開きます。
2. [[connectivity]] セクションで、次の操作を行います。
  - enable-quic-frontend の場合、true を指定します。

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

3. ファイルを保存して閉じます。
4. NICE DCV サーバー [を停止](#)して [再起動](#)します。

## NICE DCV サーバーTCP/UDPポートとリッスンアドレスの変更

デフォルトでは、NICE DCVサーバーはTCPポートをリッスン8443し、実行されるホスト上の任意のネットワークインターフェイスで通信するように設定されています。

NICE DCV サーバーのインストール後にカスタムTCPポートを指定できます。[を有効にするQUIC](#)のようにNICE DCVサーバーを設定した場合は、QUICトラフィックのカスタムUDPポートを指定することもできます。ポート番号は 1024 より大きくする必要があります。

NICE DCV サーバーがリッスンするネットワークアドレスを指定できます。例えば、IPv4またはのみIPv6を使用するかどうかを指定できます。また、サーバーを特定のネットワークインターフェイスにバインドして、トラフィックが特定のネットワークを流れるようにすることもできます。

### Important

NICE DCV サーバーのネットワーク設定に変更を適用するときは必ず、変更内容をクライアントに伝えます。例えば、クライアントはセッションへの接続に使用されるポート番号を知る必要があります。

### Tip

クライアントに公開されるネットワークアドレスとポートを制御する別の方法は、[NICE DCV Connection Gateway](#) または別のウェブプロキシまたはロードバランサーをサーバーのフロントエンドとして使用することです。ゲートウェイ経由でNICE DCVサーバーホストにアクセスすると、サーバー用の単一のアドレスを持つことができます。また、の標準ポート番号である 443 を含む 1024 より前のポート番号を使用することもできます HTTPS。

ネットワークアドレスとポートの設定の詳細については、お使いのゲートウェイのドキュメントを参照してください。

### トピック

- [NICE DCV サーバーTCP/UDPポートの変更](#)
- [特定のエンドポイントのリスニング](#)

## NICE DCV サーバーTCP/UDPポートの変更

### Windows NICE DCV server

NICE DCV サーバーで使用されるポートを変更するには、Windows レジストリエディタを使用して web-port および quic-port パラメータを設定します。

#### Windows でサーバーのポートを変更する方法

1. Windows レジストリエディタを開きます。
2. HKEY\_USERS/S-1-5-18/Software/comGSettings/nicesoftware/dcv/connectivity/ キーに移動します。
3. TCP ポートを設定するには、ウェブポートパラメータを選択します。

レジストリキーに web-port パラメータが存在しない場合は作成します。

- a. ナビゲーションペインで、接続キーのコンテンツ (右クリック) メニューを開きます。次に、新しい、DWORD (32 ビット) 値 を選択します。
  - b. [名前] に web-port と入力し、[Enter] を押します。
4. [web-port] パラメータを開きます。値データ に、新しいTCPポート番号を入力します。このパラメータを設定しない場合、NICE DCVサーバーはデフォルトでTCPポート 8443 を使用します。

#### Note

TCP ポート番号は 1024 より大きい必要があります。

5. QUIC が有効な場合、UDPポートを設定するには、quic-port パラメータを選択します。

レジストリキーに quic-port パラメータが存在しない場合は作成します。

- a. ナビゲーションペインで、接続キーのコンテンツ (右クリック) メニューを開きます。次に、新しい、DWORD (32 ビット) 値 を選択します。
  - b. [名前] に quic-port と入力し、[Enter] を押します。
6. quic-port パラメータを開きます。値データ に、新しいUDPポート番号を入力します。このパラメータを設定せず、QUICサポートが有効になっている場合、NICE DCVサーバーはデフォルトでUDPポート 8443 を使用します。

**Note**

UDP ポート番号は 1024 より大きい必要があります。

7. [OK] を選択して Windows レジストリエディタを閉じます。
8. NICE DCV サーバーを 停止して 再起動します。

## Linux NICE DCV server

NICE DCV サーバーで使用されるポートを変更するには、`dcv.conf` ファイルで `web-port` と `quic-port` パラメータを設定します。

### Linux でサーバーのポートを変更する方法

1. `/etc/dcv/` に移動し、任意のテキストエディタで `dcv.conf` を開きます。
2. `web-port` パラメータを `[connectivity]` セクションで見つけます。次に、既存の TCP ポート番号を新しい TCP ポート番号に置き換えます。

`[connectivity]` セクションに `web-port` パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

**Note**

TCP ポート番号は 1024 以上である必要があります。

3. `quic-port` パラメータを `[connectivity]` セクションで見つけます。次に、既存の UDP ポート番号を新しい UDP ポート番号に置き換えます。

`[connectivity]` セクションに `quic-port` パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

**Note**

UDP ポート番号は 1024 以上である必要があります。

4. ファイルを保存して閉じます。
5. NICE DCV サーバーを 停止して 再起動します。

## 特定のエンドポイントのリスニング

特定のネットワークアドレスのみをリッスンするには、NICE DCV サーバーの設定で `web-listen-endpoints` および `quic-listen-endpoints` パラメータを設定できます。

各エンドポイントは IPv4 または IPv6 アドレスで表され、オプションで区切られたポート番号が続きます。エンドポイントで指定されたポート番号は、`web-port` および `quic-port` パラメータで指定されたポートよりも優先されます。

複数のエンドポイントを指定できるため、一連のエンドポイントは角括弧で囲まれたカンマで区切られたリストで表され、各エンドポイントは一重引用符で囲まれます。例えば、`['0.0.0.0:8443', ':::8443']` はポート 8443 IPv6 上の任意のローカル IPv4 アドレスと任意のローカル IPv6 アドレス `:::1` を表し、`%1` は Windows ホスト上のインデックス 1 を持つネットワークインターフェイスにバインドされるアドレスを表し、`%eth1` は Linux ホスト上の `eth1` ネットワークインターフェイスにバインドされる IPv6 アドレス `:::%eth1` を表します。

**Note**

これらの設定パラメータは、NICE DCV Server 2022.0 以降でのみ使用できます。

### Windows NICE DCV server

#### Windows でサーバーのエンドポイントを変更する方法

1. Windows レジストリエディタを開きます。
2. `HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/connectivity/` キーに移動します。
3. TCP エンドポイントを設定するには、`web-listen-endpoints` パラメータを選択します。

レジストリキーに `web-listen-endpoints` パラメータが存在しない場合は作成します。

- a. ナビゲーションペインで、接続キーのコンテンツ (右クリック) メニューを開きます。次に、[新規]、[文字列の値] の順に選択します。
  - b. [名前] に `web-listen-endpoints` と入力し、[Enter] を押します。
4. `web-listen-endpoints` パラメータを開きます。[値のデータ] には、エンドポイントのリストを入力します。
  5. QUIC が有効になっている場合、UDPエンドポイントを設定するには、`quic-listen-endpoints` パラメータを選択します。

レジストリキーに `quic-listen-endpoints` パラメータが存在しない場合は作成します。

- a. ナビゲーションペインで、接続キーのコンテンツ (右クリック) メニューを開きます。次に、[新規]、[文字列の値] の順に選択します。
  - b. [名前] に `quic-listen-endpoints` と入力し、[Enter] を押します。
6. `quic-listen-endpoints` パラメータを開きます。[値のデータ] には、エンドポイントのリストを入力します。
  7. [OK] を選択して Windows レジストリエディタを閉じます。
  8. NICE DCV サーバー [を停止](#)して [再起動](#)します。

## Linux NICE DCV server

### Linux でサーバーのエンドポイントを変更する方法

1. `/etc/dcv/` に移動し、任意のテキストエディタで `dcv.conf` を開きます。
2. `web-listen-endpoints` パラメータを [connectivity] セクションで見つけます。次に、既存のエンドポイントのリストを置き換えます。

[connectivity] セクションに `web-listen-endpoints` パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

3. `quic-listen-endpoints` パラメータを [connectivity] セクションで見つけます。次に、既存のエンドポイントのリストを置き換えます。

[connectivity] セクションに `quic-listen-endpoints` パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

4. ファイルを保存して閉じます。
5. NICE DCV サーバーを [停止](#) して [再起動](#) します。

## TLS 証明書の管理

NICE DCV は、NICE DCV クライアントと NICE DCV サーバー間のトラフィックを保護するために使用される自己署名証明書を自動的に生成します。デフォルトでは、他の証明書がインストールされていない場合にこの証明書が使用されます。デフォルトの証明書には 2 つのファイルが含まれています。これらは証明書そのものです (`dcv.pem`) とキー (`dcv.key`)。詳細については、「[the section called “自己署名証明書によるリダイレクトの説明”](#)」を参照してください。

DCV クライアントユーザーがサーバーに接続すると、接続が確立される前に、検証のために実行できるサーバー証明書の警告が表示されることがあります。

ユーザーがウェブブラウザを使用して接続する場合、サーバーの証明書を信頼するかどうかに関する警告がブラウザによって表示され、管理者に問い合わせるよう推奨される場合があります。

同様に、Windows、Linux、または macOS クライアントを使用している場合は、NICE DCV サーバー管理者に特定の証明書のフィンガープリントを確認することをお勧めします。

証明書のフィンガープリントの信頼性を検証するには、`dcv list-endpoints -j` を実行して出力を証明書のフィンガープリントと照合します。

デフォルトの NICE DCV 証明書とそのキーを独自の証明書とキーに置き換えることができます。

独自の証明書を生成するときは、特定のニーズを満たす証明書属性を選択します。CN (Common Name) 属性は、ほとんどの場合、ホストのパブリックホスト名と一致します。SAN (Subject Alternative Name) 属性を指定してホストの IP アドレスに設定することがあるかもしれません。

証明書の生成方法については、個別の認証機関のドキュメントを参照してください。

**⚠ Important**

独自の証明書とキーを使用する場合は、証明書の名前を `dcv.pem` に、キーの名前を `dcv.key` にする必要があります。

## Windows NICE DCV server

Windows でサーバーのTLS証明書を変更するには

- 証明書とそのキーを Windows NICE DCVサーバーの次の場所に置きます。

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

## Linux NICE DCV server

Linux でサーバーのTLS証明書を変更するには

1. 証明書とそのキーを Linux NICE DCVサーバーの次の場所に置きます。

```
/etc/dcv/
```

2. 両方のファイルの所有権を `dcv` ユーザーに付与し、そのアクセス許可を `600` (所有者のみ、それらのファイルの読み書きが可能) に変更します。

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

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

**i Note**

2022.0 NICE DCV 以降、NICE DCVサーバーの実行中に証明書ファイルを更新すると、新しい証明書は自動的に再ロードされます。の以前のバージョンNICE DCVでは、NICE DCVサーバーを手動で[停止](#)して[再起動](#)する必要があります。

## アイドル状態のクライアントの切断

アイドル状態のクライアントを切断NICE DCVするようにを設定できます。具体的には、特定の期間にキーボードやポインタの入力をNICE DCVサーバーに送信しなかったクライアントに対してこれを行うことができます。デフォルトでは、NICE DCVサーバーは 60 分 (1 時間) アイドル状態になった後にNICE DCVクライアントを切断します。

アイドル切断タイムアウト期間をリセットする特定のアクションがあります。次のいずれかのアクションが発生すると、アイドルタイムアウト期間は設定された時間枠にリセットされます。

- マウスの移動
- マウスのボタンを押したり、マウスホイールを移動したりする
- キーボードの任意のキーを押す
- タッチスクリーンをタッチする (有効な場合)
- スタイラスの使用 (有効な場合)
- ゲームパッドの使用 (有効になっている場合)
- ウェブカメラによるストリーミング (有効になっている場合)
- ファイルのアップロード、ディレクトリの作成、ファイルのダウンロード、項目の一覧表示などのファイルストレージオペレーション

### Note

オーディオデバイスを接続して使用しても、アイドルタイムアウト期間はリセットされません。

サーバーを設定NICE DCVして、アイドル状態のクライアントに通知を送信することもできます。この通知は、セッションが切断されることを通知するためのものです。タイムアウト通知は、NICE DCVサーバーおよびクライアントバージョン 2017.4 以降でのみサポートされています。

以下の手順を使用して、カスタムアイドルタイムアウト期間を指定できます。

### Windows NICE DCV server

NICE DCV サーバーのアイドルタイムアウト期間を変更するには、Windows レジストリエディタを使用して `idle-timeout` パラメータを設定する必要があります。

Windows でアイドルタイムアウト期間を変更するには

1. Windows レジストリエディタを開きます。
2. HKEY\_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/connectivity/ キーに移動し、アイドルタイムアウトパラメータを選択します。

パラメータが見つからない場合は、次のステップに従ってパラメータを作成します。

- a. ナビゲーションペインで、接続キーのコンテンツ (右クリック) メニューを開きます。次に、新しい、DWORD (32 ビット) 値 を選択します。
  - b. [名前] に idle-timeout と入力し、[Enter] を押します。
3. [idle-timeout] パラメータを開きます。[値のデータ] に、アイドルタイムアウト期間 (分単位、10 進法) の値を入力します。アイドル状態のクライアントを切断しないようにするには、0 を入力します。
  4. [OK] を選択して Windows レジストリエディタを閉じます。

(オプション) アイドル状態のクライアントにタイムアウト通知を送信するようにNICE DCVサーバーを設定するには

1. HKEY\_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/connectivity/ キーに移動し、idle-timeout-warningパラメータを選択します。

パラメータが見つからない場合は、次のステップに従ってパラメータを作成します。

- a. ナビゲーションペインで、接続キーのコンテンツ (右クリック) メニューを開きます。次に、新しい、DWORD (32 ビット) 値 を選択します。
  - b. [名前] に idle-timeout-warning と入力し、[Enter] を押します。
2. idle-timeout-warning パラメータを開きます。[値のデータ] に、関連する警告通知が送信されてから切断されるまでの秒数 (10 進法) を入力します。たとえば、アイドルタイムアウトに達する 2 分前に通知を送信する場合は、120 と入力します。
  3. [OK] を選択して Windows レジストリエディタを閉じます。

## Linux NICE DCV server

NICE DCV サーバーのアイドルタイムアウト期間を変更するには、dcv.conf ファイルで idle-timeoutパラメータを設定する必要があります。

Linux でアイドルタイムアウト期間を変更するには

1. 任意のテキストエディタで `/etc/dcv/dcv.conf` を開きます。
2. `idle-timeout` パラメータを `[connectivity]` セクションで見つけます。次に、既存のタイムアウト期間を新しいタイムアウト期間(分単位、10 進法) に置き換えます。

`[connectivity]` セクションに `idle-timeout` パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

アイドル状態のクライアントを切断しないようにするには、`0` を入力します。

3. ファイルを保存して閉じます。

(オプション) アイドル状態のクライアントにタイムアウト通知を送信するようにNICE DCVサーバーを設定するには

1. 任意のテキストエディタで `/etc/dcv/dcv.conf` を開きます。
2. `idle-timeout-warning` パラメータを `[connectivity]` セクションに追加し、関連する警告通知が送信されてから切断されるまでの秒数(10 進法) を入力します。

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

たとえば、アイドルタイムアウトに達する 2 分前に通知を送信する場合は、`120` と指定します。

3. ファイルを保存して閉じます。

## Linux NICE DCVサーバーでGPUの共有の有効化

GPU 共有を使用すると、複数のNICE DCV仮想セッションGPU間で 1 つ以上の物理 GPU を共有できます。セッションの詳細については、「[NICE DCV セッションの管理](#)」を参照してください。GPU 共有を使用すると、1 つのNICE DCVサーバーを使用し、サーバーの物理GPUリソースを共有する複数の仮想セッションをホストできます。

**Note**

GPU 共有は Linux NICE DCVサーバーでのみサポートされています。

## 前提条件

開始する前に、次の前提条件を完了します。

- Linux NICE DCVサーバーにサーバーをインストールします。
- サーバーに NICE DCV dcv-gl および nice-xdvcv パッケージをインストールします。
- サーバーにサポートされている が少なくとも 1 NVIDIA つあることを確認します GPU。
- サーバーに NVIDIA GPU ドライバーをインストールします。公式 NVIDIA ドライバーが必要です。オープンソース NVIDIA ドライバーはサポートされていません。
- NVIDIA GPU ドライバーがハードウェアアクセラレーション OpenGL をサポートしていることを確認します。
- X サーバーをインストールし、Device ファイルの Screen セクションと xorg.conf セクションを設定します。

**Note**

nvidia-xconfig NVIDIA ユーティリティを使用して xorg.conf、 ファイルを自動的に作成し、使用可能なすべての NVIDIA に対して設定できます GPUs。

- X サーバーが実行中であることを確認します。
- ( オプション ) dcvgltest ツールを実行して NICE DCVサーバー設定を確認します。詳細については、「[インストール後のチェック](#)」を参照してください。

nice-dcv-gltest パッケージをインストールし、dcvgltest テストアプリケーションを実行して、サーバーが GPU 共有用に適切に設定されているかどうかを確認することもできます。

GPU 共有を有効にするには、仮想セッション GPUs で使用する のリストを指定する必要があります。を指定しない場合 GPUs、表示名 の標準 X サーバー GPU で使用される のみ :0.0 が使用されません。

このトピック GPUs で前述した前提条件を満たしたら、 dcv.conf ファイルの gl-displays パラメータで を指定します。

Linux NICE DCVサーバーでGPU共有を有効にするには

1. /etc/dcv/ に移動し、任意のテキストエディタで dcv.conf ファイルを開きます。
2. [display/linux] パラメータを gl-displays セクションに追加します。次に、使用可能な GPU を次の形式で指定します。

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

各パラメータの意味は次のとおりです。 *xserver\_port* はサーバーであり、 *screen\_number* は、に関連する画面に関連付けられている番号ですGPU。 *screen\_number* は から開始します0。

次の例は、デフォルトの X Server セッションでGPU実行されている 2 つの の gl-displays パラメータを示しています。

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

3. ファイルを保存して閉じます。
4. NICE DCV サーバーを [停止](#)して [再起動](#)します。

## タッチスクリーンとスタイラスのサポートの有効化

### Note

USB タッチスクリーンおよびスタイラスデバイスのリダイレクトは必要ありません。また、NICE DCVサーバーにベンダードライバーをインストールする必要はありません。NICE DCV は、ネイティブオペレーティングシステムを使用してタッチスクリーンとスタイラスをサポートしますAPIs。

Windows は Windows Ink を使用します。

Linux は X11 入力インジェクションを使用します。

- Windows サーバーでのサポート

タッチスクリーンは、サポート対象のすべての Windows オペレーティングシステムでサポートされています。スタイラスは Windows 10 と Windows 2019 以降のすべてのサポート対象の Windows オペレーティングシステムでサポートされています。Windows 2016、Windows 8.1、およびそれ以前のバージョンではサポートされていません。デフォルトでは、機能は Windows NICE DCV サーバーで有効になっています。追加の設定は必要ありません。

- Linux サーバーでのサポート

タッチスクリーンとスタイラスは、サポートされているすべての Linux オペレーティングシステムでサポートされています。この機能は、Linux NICE DCV サーバーでホストされている仮想セッションでデフォルトで有効になっています。ただし、Linux NICE DCV サーバーでホストされているコンソールセッションで機能を有効にするには、いくつかの追加設定が必要です。

**⚠ Important**

クライアントとサーバーの両方で機能がサポートされ、サーバーで NICE DCV 有効になっている場合、でのタッチスクリーンとスタイラスの使用が有効になります。クライアントサポートの詳細については、「[ユーザーガイド](#)」の「[クライアント機能 NICE DCV](#)」を参照してください。

Linux NICE DCV サーバーでホストされているコンソールセッションのタッチスクリーンとスタイラスのサポートを有効にするには

1. お好みのテキストエディタを使用して `/etc/X11/xorg.conf` を開きます。
2. 次のセクションをファイルに追加します。

```
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. ServerLayout セクションの最後に以下を追加します。

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

例:

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

4. 変更内容を保存し、ファイルを閉じます。

5. X サーバーを再起動します。

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

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

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

6. 入力デバイスが適切に設定されていることを確認するには、次のコマンドを実行します。

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

DCV スタイルスペン、DCV スタイルス消しゴム、DCV タッチスクリーンがコマンド出力に表示されます。以下は出力例です。

```
| 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)]
```

	<b>DCV Touchscreen</b>	id=10	[slave pointer (2)]
	Virtual core keyboard	id=3	[master keyboard (2)]
	Virtual core XTEST keyboard	id=5	[slave keyboard (3)]

## スタイラスの圧力範囲の設定

アプリケーションの中には、スタイラスの圧力範囲を 0~2048 の範囲に減らす必要があるものもあります。/etc/X11/xorg.conf ファイルで Pressure2k オプションを true に設定することで圧力範囲を設定できます。

### スタイラスの圧力を設定する方法

1. お好みのテキストエディタを使用して/etc/X11/xorg.confを開きます。
2. 次のセクションをファイルに追加します。

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

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

3. 変更内容を保存し、ファイルを閉じます。
4. X サーバーを再起動します。

## ゲームパッドサポートの有効化

NICE DCV Server 2022.0 以降では、サポートされている Windows または Linux オペレーティングシステムに接続するときにゲームパッドデバイスを使用できます。

以下のゲームパッドデバイスがサポートされています。

- Xbox 360 のコントローラー
- DualShock 4 コントローラー

上記のデバイスと互換性のあるデバイス、またはサポート対象のデバイスのいずれかをエミュレートするように設定できる他のデバイスも動作する場合があります。

#### Note

ゲームパッドデバイスは、Windows ネイティブNICE DCVクライアントを使用する場合にのみサポートされます。NICE DCV Client 2022.0 以降を使用していることを確認します。

ゲームパッドのサポートを有効にするには、最新バージョンの NICE DCV サーバーがインストールされていること、および Gamepad ドライバーのインストールが選択されていることを確認します。詳細については、「[Windows への NICE DCV Server のインストール](#)」を参照してください。ドライバーをインストールすると、この機能は Windows NICE DCVサーバーでデフォルトで有効になります。

## Xbox 360 コントローラーのサポート

Xbox 360 コントローラーには Windows ドライバーをインストールする必要があります。このドライバーは Windows に自動的にインストールされないため、Windows 公式のアップデート Web サイトから入手する必要があります。

Xbox 360 コントローラードライバーをダウンロードしてインストールするには

1. Microsoft Update Catalog ページでドライバーを検索します (<https://www.catalog.update.microsoft.com/Search.aspx?q=game+devices+XBOX+360+Controller+For+Windows>)
2. オペレーティングシステムに対応する最新バージョンのドライバーをダウンロードします。
3. .cab ファイルを開き、その内容を抽出 (解凍) します。

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

4. 次のコマンドでドライバーの .inf ファイルをインストールします。

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

## USB リモート化の有効化

を使用するとNICE DCV、クライアントは 3D ポインティングUSBデバイスや認証デバイスなど、さまざまな特殊なデバイスを使用できます。デバイスはコンピュータに物理的に接続され、NICE DCV サーバーで実行されているアプリケーションとやり取りします。

### Important

NICE DCV は、USBデバイスをリダイレクトするための一般的なメカニズムを提供します。ネットワーク遅延の影響を受けやすいデバイスの場合は、問題が発生することがあります。さらに、ドライバーの互換性の問題により一部のデバイスが期待どおりに機能しない可能性があります。本番稼働へのデプロイを実行する前に、デバイスが期待どおりに動作することを確認してください。

### Note

USB リモート化は Windows クライアントでのみサポートされます。ポータブル Windows クライアントやウェブブラウザクライアントではサポートされていません。NICE DCV クライアントで追加の設定が必要になる場合があります。クライアントにUSBリモート化をインストールする方法については、「NICE DCVユーザーガイド」の「インストール可能な [Windows クライアント](#)」のオプションステップを参照してください。

NICE DCV サーバーは許可リストを使用して、クライアントが使用できるUSBデバイスを決定します。デフォルトでは、一般的に使用される一部のUSBデバイスが許可リストに追加されます。つまり、クライアントはこれらのUSBデバイスをコンピュータに接続し、追加の設定なしでサーバー上で使用できます。詳細については、「NICE DCVユーザーガイド [USB](#)」の「[リモート化の使用](#)」を参照してください。

ただし、一部の特殊なデバイスは、デフォルトで許可リストに追加されない場合があります。これらのデバイスは、クライアントが使用する前に、NICE DCVサーバー上の許可リストに手動で追加する必要があります。追加されると、Windows クライアントの [設定] メニューに表示されます。

### Windows NICE DCV server

許可リストにUSBデバイスを追加するには、クライアントからUSBデバイスのフィルター文字列を取得し、usb-devices.conf ファイルに追加する必要があります。

Windows NICE DCVサーバーの許可リストにUSBデバイスを追加するには

1. 最新バージョンのNICE DCVサーバーがインストールされており、USBリモート化ドライバーのインストールが選択されていることを確認します。詳細については、「[Windows への NICE DCV Server のインストール](#)」を参照してください。
2. USB デバイスのハードウェアドライバーをNICE DCVサーバーにインストールします。
3. Windows クライアントマシンのファイルマネージャーで C:\Program Files (x86)\NICE\DCV\Client\bin\ に移動します。
4. dcvusblist.exe を実行します。
5. リスト内のUSBデバイスを右クリックします。
6. ドロップダウンメニューから [フィルター文字列のコピー] を選択します。
7. サーバーで任意のテキストエディタを使用して C:\Program Files\NICE\DCV\Server\conf\usb-devices.conf を開き、ファイルの下部の新しい行にフィルター文字列を追加します。
8. ファイルを保存して閉じます。
9. NICE DCV サーバー [を停止](#)して [再起動](#)します。

## Linux NICE DCV server

許可リストにUSBデバイスを追加するには、USBデバイスのフィルター文字列を usb-devices.conf ファイルに追加します。

Linux NICE DCVサーバーの許可リストへのUSBデバイスの追加

1. NICE DCV サーバーとDCVUSBドライバーの最新バージョンがインストールされていることを確認します。詳細については、「[Linux への NICE DCV サーバーのインストール](#)」を参照してください。
2. USB デバイスのハードウェアドライバーをNICE DCVサーバーにインストールします。
3. Windows クライアントマシンのファイルマネージャーで C:\Program Files (x86)\NICE\DCV\Client\bin\ に移動します。
4. dcvusblist.exe を実行します。
5. リスト内のUSBデバイスを右クリックします。
6. ドロップダウンメニューから [フィルター文字列のコピー] を選択します。
7. サーバーで任意のテキストエディタを使用して /etc/dcv/usb-devices.conf を開き、ファイルの下部の新しい行にフィルター文字列を追加します。

8. ファイルを保存して閉じます。
9. NICE DCV サーバーを[停止](#)して[再起動](#)します。

## スマートカードのキャッシュの設定

スマートカードキャッシュ機能を使用すると、NICE DCVサーバーはスマートカード値をキャッシュできます。この機能を有効にすると、NICE DCVサーバーはクライアントのスマートカードへの最近の呼び出しの結果をキャッシュします。その後の呼び出しでは、クライアントからではなく、サーバーのキャッシュから直接取得されます。これにより、クライアントとサーバー間で転送されるトラフィックの量が削減され、パフォーマンスが向上します。クライアントのインターネット接続が遅い場合には特にこれが役立ちます。

スマートカードのキャッシュ機能はデフォルトでは無効になっています。クライアントは、実行するアプリケーションごとに `DCV_PCSC_ENABLE_CACHE` 環境変数を設定してスマートカードのキャッシュを手動で有効にすることができます。手順については、[「ユーザーガイド」の「スマートカードの使用NICE DCV」](#)を参照してください。または、`DCV_PCSC_ENABLE_CACHE`環境変数に指定された値に関係なく、スマートカードのキャッシュを永続的に有効または無効にするようにNICE DCVサーバーを設定できます。

### Linux NICE DCV server

Linux NICE DCVサーバーでスマートカードキャッシュを永続的に有効または無効にするには

1. `/etc/dcv/` に移動し、任意のテキストエディタで `dcv.conf` を開きます。
2. `enable-cache` パラメータを `[smartcard]` セクションで見つけます。スマートカードのキャッシュを永続的に有効化するには、「`'always-on'`」と入力します。スマートカードのキャッシュを永続的に無効化するには、「`'always-off'`」と入力します。

`[smartcard]` セクションに `enable-cache` パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

3. ファイルを保存して閉じます。
4. NICE DCV サーバーを[停止](#)して[再起動](#)します。

## Windows NICE DCV server

Windows NICE DCVサーバーでスマートカードキャッシュを永続的に有効または無効にするには

1. Windows レジストリエディタを開きます。
2. HKEY\_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/smartcard/ キーに移動し、enable-cache パラメータを選択します。

パラメータが存在しない場合は、次のステップに従ってパラメータを作成します。

- a. 左のパネルで、smartcard キーのコンテキスト (右クリック) メニューを開き、[New] (新規)、[String Value] (文字列値) の順に選択します。
  - b. [名前] に enable-cache と入力し、[Enter] を押します。
3. [enable-cache] パラメータを開きます。[Value data] (値のデータ) で、スマートカードのキャッシュを永続的に有効にする場合は always-on を、スマートカードのキャッシュを永続的に無効にする場合は always-off を入力します。
  4. [OK] を選択して Windows レジストリエディタを閉じます。

## WebAuthn リダイレクトの設定

NICE DCV Server 2023.1 以降、ユーザーはリモートセッション内でサポートされているブラウザでウェブ認証 (WebAuthn) 標準を使用するウェブアプリケーションを認証できます。これは、認証プロンプトを Windows Hello や などのローカルに接続されたFIDO2認証ツール YubiKey、またはその他のFIDO2準拠した認証ツールにリダイレクトすることによって行われます。

WebAuthn リダイレクトはUSBリダイレクトとは無関係に動作します。ベンダー固有のドライバーをNICE DCVサーバーにインストールする必要はありません。WebAuthn リクエストのリダイレクトは、ブラウザAPIのネイティブ を介して容易になります。

を使用する前に WebAuthn、[サポートされている機能表](#)を再度チェックして、すべての要件を満たしていることを確認してください。

サポートされるブラウザ:

- Google Chrome 116 以降
- Microsoft Edge 116 以降

WebAuthn リダイレクトは、アクセスwebauthn-redirectation許可を使用して有効または無効にできません。詳細については、「[アクセス許可ファイルの使用](#)」を参照してください。

WebAuthn リダイレクトするには、ブラウザ拡張機能をリモートサーバーにインストールする必要があります。機能が有効で、ブラウザ拡張機能がインストールされている場合、セッション内でブラウザで実行されているウェブアプリケーションによって開始された WebAuthn リクエストは、ローカルクライアントにシームレスにリダイレクトされます。その後、ユーザーは Windows Hello や などの利用デバイスを使用して認証を YubiKey 確定できます。

#### Note

この機能は、リモートセッション中にブラウザ WebAuthn 内で を許可しますが、WebAuthn 認証機能を使用したDCVセッション認証はサポートしていません。

## WebAuthn リダイレクトブラウザ拡張機能のセットアップ

### ブラウザの初回起動時の自動プロンプト

WebAuthn リダイレクトを有効にして NICE DCV Server 2023.1 をインストールすると、ユーザーはブラウザを初めて起動するときにブラウザ拡張機能を有効にするように求められます。拡張機能をインストールしないこと、または後でアンインストールすることを選択した場合、WebAuthn リダイレクトは機能しません。管理者は、グループポリシーを使用してインストールを強制できます。

### グループポリシーを使用した のインストール

拡張をより広範囲にデプロイしようとしている組織には、グループポリシーを利用できます。

#### Microsoft Edge を使用する場合

1. [Microsoft Edge 管理用テンプレート](#)をダウンロードおよびインストールします。
2. グループポリシー管理ツール (gpmmc.msc) を起動します。
3. 「フォレスト」 > ドメイン「FQDN」 (example.com など) > グループポリシーオブジェクトに移動します。
4. 目的のポリシーを選択するか、新しいポリシーを作成して右クリックし、[編集] を選択します。
5. 「コンピューター構成」 > 「管理用テンプレート」 > 「Microsoft Edge」 > 「拡張機能」の順に移動します。

6. [拡張機能管理設定の構成] にアクセスし、[有効] に設定します。
7. [拡張機能管理設定の構成] フィールドに以下を入力します。

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

8. 変更内容を保存し、サーバーを再起動します。

### Google Chrome を使用する場合

1. [Google Chrome 管理用テンプレート](#) を取得して実装します。
2. Microsoft Edge の手順と同様に、グループポリシー管理ツールに移動します。
3. [コンピュータの構成] > [管理用テンプレート] > [Google Chrome] > [拡張機能] に進みます。
4. [拡張機能管理設定の構成] にアクセスし、[有効] に設定します。
5. [拡張機能管理設定の構成] フィールドに以下を入力します。

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

6. 変更内容を保存し、サーバーを再起動します。

### 手動インストール

拡張機能はそれぞれのブラウザストアから入手できます。

- [Microsoft Edge アドオン](#)
- [Chrome ウェブストア](#)

### 手動でインストールする場合

1. NICE DCV セッションに接続します。
2. 任意のブラウザを開き、関連するブラウザストア (上記のリンク) に移動します。
3. [インストール] (Microsoft Edge) または [Chrome に追加] (Google Chrome) を選択して続行します。

4. 画面上の指示に従ってください。拡張機能が正常に追加されると、確認メッセージが表示されません。

## シークレットモードでの WebAuthn リダイレクトの使用 (Chrome のみ)

シークレットモードを使用する場合、Amazon DCV WebAuthn Redirection Extension 内で実行することを特別に許可する必要があります。許可しない場合、Redirection WebAuthn は発生しません。これを実行するには:

1. 拡張機能設定を開きます。
2. 詳細で Incognito で許可を見つけます。
3. スイッチをオンに切り替えます。

## セッションストレージの有効化

セッションストレージは、クライアントが特定のNICE DCVセッションに接続したときにアクセスできるNICE DCVサーバー上のフォルダです。セッションストレージをセッション用に有効にすると、クライアントは特定のフォルダからファイルをダウンロードしたり、アップロードしたりできるようになります。この機能により、クライアントはセッション接続中にファイルを共有できるようになります。

### トピック

- [Windows NICE DCV Server でのセッションストレージの有効化](#)
- [Linux NICE DCVサーバーでのセッションストレージの有効化](#)

## Windows NICE DCV Server でのセッションストレージの有効化

セッションストレージを有効にするには、まずセッションストレージに使用するフォルダを作成します。次に、Windows レジストリエディタを使用して storage-root パラメータを設定します。

Windows でセッションストレージを有効にするには

1. セッションストレージに使用するフォルダを作成します (例: c:\session-storage)。
2. storage-root パラメータを設定します。
  - a. Windows レジストリエディタを開きます。

- b. HKEY\_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/session-management/automatic-console-session キーに移動し、storage-root パラメータを選択します。

レジストリキーに storage-root パラメータが存在しない場合は、以下の手順に従ってパラメータを作成します。

- i. ナビゲーションペインで、セッション管理/automatic-console-sessionキーのコンテキスト (右クリック) メニューを開きます。次に、[New] (新規)、[String] (文字列) の順に選択します。
  - ii. [名前] に storage-root と入力し、[Enter] を押します。
- c. [storage-root] パラメータを開きます。[Value data] (値のデータ) に、ステップ 1 で作成したフォルダへのフルパスを入力します。

パスで %home% を使用して、現在サインインしているユーザーのホームディレクトリを指定することもできます。たとえば、次のパスでは、c:\Users\*username*\storage\ がセッションストレージディレクトリとして使用されます。

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

#### Note

指定したサブディレクトリが存在しない場合、セッションストレージは無効になります。

- d. [OK] を選択して Windows レジストリエディタを閉じます。
  - e. NICE DCV サーバーを[停止](#)して[再起動](#)します。
3. セッションを開始してから --storage-root オプションを指定します。詳細については、「[NICE DCV セッションの開始](#)」を参照してください。

## Linux NICE DCVサーバーでのセッションストレージの有効化

セッションストレージを有効にするには、セッションストレージに使用するフォルダを作成した上で、storage-root ファイルの dcv.conf パラメータを設定します。

Linux でセッションストレージを有効にするには

1. セッションストレージに使用するフォルダを作成します (例: /opt/session-storage/)。

2. storage-root パラメータを設定します。
  - a. /etc/dcv/ に移動し、任意のテキストエディタで dcv.conf を開きます。
  - b. storage-root パラメータを [session-management/automatic-console-session] セクションで見つけます。ステップ 1 で作成したフォルダへのフルパスで既存のパスを置き換えます。

[session-management/automatic-console-session] セクションに storage-root パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

パスで %home% を使用して、現在サインインしているユーザーのホームディレクトリを指定することもできます。たとえば、次のパラメータでは、セッションストレージに \$HOME/storage/ ディレクトリを使用します。

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

 Note

指定したサブディレクトリが存在しない場合、セッションストレージは無効になります。

3. ファイルを保存して閉じます。
4. NICE DCV サーバー [を停止](#)して [再起動](#)します。
5. セッションを開始してから --storage-root オプションを指定します。詳細については、[「NICE DCV セッションの開始」](#)を参照してください。

## Linux NICE DCVサーバーでのプリンターの設定

NICE DCV では、ローカルリダイレクトプリンターまたは仮想NICE DCVプリンターのいずれかに印刷できます。

サポートされている Linux ディストリビューションを使用している場合は、印刷をサポートするようにNICE DCVサーバーを設定する必要があります。

Windows NICE DCVサーバーを使用している場合は、追加の設定は必要ありません。

Linux NICE DCVサーバーでプリンターリダイレクトを有効にするには

1. サーバーに CUPS サービスをインストールします。

- Amazon Linux 2、RHEL、および CentOS

```
$ sudo yum install cups
```

- Ubuntu

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

- SUSE Linux Enterprise

```
$ sudo zypper install cups
```

2. dcv ユーザーをプリンター管理者グループに追加します。プリンター管理者グループの名前は、オペレーティングシステムごとに異なる場合があります。例えば、プリンター管理者グループに lpadmin という名前が付けられている場合、以下のコマンドを実行します。

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

3. このプリンター管理者グループが cups 設定ファイルの SystemGroup パラメータに引用されていることを確認してください。例えば、プリンター管理者グループに lpadmin という名前が付けられている場合、テキストエディタを使用して /etc/cups/cups-files.conf を開き、次の行を検索します。

```
SystemGroup lpadmin
```

設定ファイルにその行が表示されたら、インストールは完了です。次のステップに進みます。

設定ファイルに行が表示されない場合は、次の形式の行を手動で追加し、ファイルを保存して閉じます。

```
SystemGroup printer_admin_groupname
```

4. ( SUSE Linux Enterprise のみ) プリンター管理者グループに、Cups ローカル証明書を読み取るアクセス許可があることを確認します。この証明書は /var/run/cups/certs/ ディレクトリ

にあります。例えば、プリンター管理者グループに lpadmin という名前が付けられている場合、以下のコマンドを実行します。

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

5. cups サービスを再起動します。

```
$ sudo systemctl restart cups
```

6. NICE DCV サーバーを[停止](#)して[再起動](#)します。

## プリンター問題のトラブルシューティング

SUSE Linux Enterprise および 8 RHEL では、プリンターソケットへの接続が妨げられる場合があります。これらのオペレーティングシステムのいずれかを実行しているときに印刷に問題が生じた場合は、ログファイルを調べて、このことが原因かどうかを判断してください。

テキストエディタを使用して /var/log/audit/audit.log を開き、ログに次のような行があるかどうかを確認します。

```
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
```

ログファイルに同様の行が表示されている場合、オペレーティングシステムがプリンターソケットへのアクセスを妨げています。

この問題を解決するには、プリンターソケットへのアクセスを許可する CUPS ポリシーを作成する必要があります。これを作成するには、次のステップを実行します。

1. 必要なポリシーファイルを作成します。任意のテキストエディタを使用して、cupsd\_policy という名前の新規ファイルを作成し、以下のコンテンツを追加します。

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

2. ポリシーをインストールします。

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

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

## Linux NICE DCVサーバーでのクリップボードの設定

Linux オペレーティングシステムには 2 つのバッファがあり、コンテンツをコピーして貼り付けることができます。これらのバッファはプライマリ選択とクリップボードです。コンテンツをプライマリ選択にコピーするには、ポインタをドラッグしてコンテンツをハイライト表示します。それをプライマリ選択から貼り付けるには、ポインタまたは Shift+Insert キーボードショートカットのいずれかを使用します。コンテンツをクリップボードにコピーするには、コンテンツをハイライト表示し、コンテキスト (右クリック) メニューからコピーを選択します。クリップボードから貼り付けるには、コンテキスト (右クリック) メニューから貼り付けを選択します。

Linux NICE DCVサーバーでは、クライアントとサーバー間でコピーアンドペーストアクションを実行するときに、プライマリ選択またはクリップボードを使用するようにサーバーを設定できます。

### トピック

- [クライアントクリップボードコンテンツをプライマリ選択に貼り付ける](#)
- [クライアントクリップボードにプライマリ選択コンテンツをコピーする](#)

## クライアントクリップボードコンテンツをプライマリ選択に貼り付ける

デフォルトでは、クライアントにコピーされたコンテンツはクリップボードに配置されます。このコンテンツをサーバーに貼り付けるには、コンテキスト (右クリック) メニューを使用してクリップボードから貼り付ける必要があります。

クリップボードのコンテンツをプライマリ選択に配置するようにサーバーを設定できます。これにより、コンテキスト (右クリック) メニューを使用して、両方のクリップボードからコピーしたコンテンツを貼り付けることができます。または、マウスの中央ボタンまたは Shift+Insert キーボードショートカットのいずれかを使用して、プライマリ選択からコンテンツをコピーして貼り付けることもできます。

クリップボードのコンテンツをプライマリ選択に配置するようにサーバーを設定するには

1. /etc/dcv/ に移動し、任意のテキストエディタで dcv.conf を開きます。

2. [clipboard] セクションで primary-selection-paste パラメータを見つけ、値を true に設定します。

[clipboard] セクションに primary-selection-paste パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

3. ファイルを保存して閉じます。
4. NICE DCV セッションを[停止](#)して[再起動](#)します。

## クライアントクリップボードにプライマリ選択コンテンツをコピーする

デフォルトでは、ユーザーはクリップボードを使用してサーバーからクライアントにのみコンテンツをコピーできます。つまり、プライマリ選択にコピーしたコンテンツをクライアントに貼り付けることはできないということです。

プライマリ選択コンテンツをクリップボードに配置するようにサーバーを設定できます。つまり、ユーザーがサーバー上のプライマリ選択にコンテンツをコピーすると、そのコンテンツもクリップボードにコピーされます。これにより、ユーザーはクリップボードのコンテンツをクライアントに貼り付けることができます。

プライマリ選択コンテンツをクリップボードに配置するようにサーバーを設定するには

1. /etc/dcv/ に移動し、任意のテキストエディタで dcv.conf を開きます。
2. [clipboard] セクションで primary-selection-copy パラメータを見つけ、値を true に設定します。

[clipboard] セクションに primary-selection-copy パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

3. ファイルを保存して閉じます。
4. NICE DCV セッションを[停止](#)して[再起動](#)します。

## マルチチャンネルオーディオの設定

NICE DCV は、NICE DCV ネイティブクライアントを使用する場合、最大 7.1 のオーディオチャンネルをサポートします。ウェブブラウザクライアントでは、ステレオ 2.0 オーディオチャンネルのみがサポートされます。

NICE DCV では、次のマルチチャンネルオーディオ設定がサポートされています。

- ステレオ 2.0 (2 チャンネル)
- 4 チャンネルステレオ 4.0 (4 チャンネル)
- サラウンド 5.1 (6 チャンネル)
- サラウンド 7.1 (8 チャンネル) - Windows NICE DCV サーバーのみ



サーバーで提供されるチャンネル数よりも少ない数のオーディオチャンネルをクライアントから要求された場合、サーバーによりチャンネル数がダウンミックスされます。クライアントから要求されたチャンネル数に一致させるためです。例えば、クライアントからサラウンドサウンド 5.1 が要求され、サーバーではサラウンドサウンド 7.1 までサポートされているとします。その場合はサーバーによりオーディオが 5.1 にダウンミックスされます。

サーバーで、ソースアプリケーションのオーディオ出力に一致させるためにオーディオが自動的にダウンミックスされることはありません。例えば、ソースアプリケーションでサラウンドサウンド 7.1 が提供されているのに対し、クライアントではステレオ 2.0 しかサポートされていないとします。

左前と右前のオーディオチャンネルのみがクライアントに対してストリーミングされます。残りのチャンネルは失われます。これが true の場合、オーディオチャンネルが失われないように、オーディオチャンネルをダウンミックスするようにNICE DCVサーバーを設定します。

## トピック

- [Windows NICE DCVサーバーでのオーディオチャンネルの設定](#)
- [Linux NICE DCVサーバーでのオーディオチャンネルの設定](#)

## Windows NICE DCVサーバーでのオーディオチャンネルの設定

Windows サーバーではサラウンドサウンド 7.1 (オーディオチャンネル 8 つ) がサポートされています。デフォルトの設定はステレオです。ただし、異なる設定を使用するようにサーバーを設定することができます。

Windows サーバーでのオーディオチャンネルの設定:

1. [Sound Control Panel] (サウンドコントロールパネル) を開きます。デスクトップのタスクバーのスピーカーアイコンを右クリックし、[Sounds] (サウンド) を選択します。
2. 再生タブを開き、NICE DCVスピーカーを選択します。
3. [設定] を選択します。
4. 必要なチャンネル設定を選択します。
5. [OK] を選択します。

## Linux NICE DCVサーバーでのオーディオチャンネルの設定

Linux サーバーではデフォルトでステレオ 2.0 (オーディオチャンネル 2 つ) がサポートされており、マルチチャンネルオーディオがサポートされるようにするにはいくつかの追加設定が必要です。

以下の手順を実行する必要があります。

1. PulseAudio サウンドサーバーを設定します。
2. PulseAudio デバイスを使用するようにNICE DCVサーバーを設定します。
3. 使用するチャンネル数を設定します。

## PulseAudio サウンドサーバーを設定するには

1. 任意のテキストエディタで `/etc/pulse/default.pa` を開きます。
2. ファイルの末尾に次の行を追加します。

```
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. ファイルを保存して閉じます。

PulseAudio サウンドサーバーを設定したら、PulseAudio サウンドNICE DCVサーバーから音声をキャプチャするようにサーバーを設定する必要があります。

## PulseAudio デバイスを使用するようにNICE DCVサーバーを設定するには

1. 任意のテキストエディタで `/etc/dcv/dcv.conf` を開きます。
2. `grab-device` パラメータを `[audio]` セクションで見つけます。次に、既存の値を、前のステップで取得したデバイス名に置き換えます。

`[audio]` セクションに `grab-device` パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

3. ファイルを保存して閉じます。

PulseAudio サウンドNICE DCVサーバーから音声をキャプチャするようにサーバーを設定したら、使用するチャンネルの数を指定できます。

## 使用するチャンネル数を設定する方法

1. 任意のテキストエディタで `/etc/dcv/dcv.conf` を開きます。
2. `source-channels` パラメータを `[audio]` セクションで見つけます。次に、既存のチャンネル数を、2.0 の場合は 2 に、4.0 の場合は 4 に、5.1 の場合は 6 に置き換えます。

`[audio]` セクションに `source-channels` パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

3. ファイルを保存して閉じます。
4. NICE DCV サーバーを[停止](#)して[再起動](#)します。

## HTTP ヘッダーの設定

ユーザーがウェブブラウザNICE DCVクライアントを使用してセッションに接続するときに、クライアントに追加のHTTPレスポンスヘッダーを送信するようにNICE DCVサーバーを設定できます。レスポンスヘッダーは、ユーザーが接続しているNICE DCVサーバーに関する追加情報を提供することができます。

トピック

- [Windows NICE DCV Server でのHTTPヘッダーの設定](#)
- [Linux NICE DCVサーバーでのHTTPヘッダーの設定](#)

## Windows NICE DCV Server でのHTTPヘッダーの設定

Windows でHTTPヘッダーを設定するには、Windows レジストリエディタを使用して `web-extra-http-headers` パラメータを設定します。

Windows でHTTPヘッダーを設定するには

1. Windows レジストリエディタを開きます。
2. `HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/connectivity/` キーに移動します。
3. ナビゲーションペインで、接続キーのコンテンツ (右クリック) メニューを開きます。次に、[New] (新規)、[String] (文字列) の順に選択します。
4. [名前] に `web-extra-http-headers` と入力し、[Enter] を押します。
5. `web-extra-http-headers` パラメータを開きます。値データ には、HTTPヘッダー名と値を次の形式で入力します。

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

複数のヘッダーを指定するには、それらをカンマ区切りのリストに追加します。

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

6. [OK] を選択して Windows レジストリエディタを閉じます。
7. NICE DCV サーバーを[停止](#)して[再起動](#)します。

## Linux NICE DCVサーバーでのHTTPヘッダーの設定

Linux でHTTPヘッダーを設定するには、`dcv.conf` ファイルで `web-extra-http-headers` パラメータを設定します。

Linux でHTTPヘッダーを設定するには

1. 任意のテキストエディタで `/etc/dcv/dcv.conf` を開きます。
2. `[connectivity]` セクションの位置を確認します。HTTP ヘッダー名と値を次の形式で指定します。

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

複数のヘッダーを指定するには、それらをカンマ区切りのリストに追加します。

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

3. ファイルを保存して閉じます。
4. NICE DCV サーバーを[停止](#)して[再起動](#)します。

## NICE DCV 認証の設定

デフォルトでは、クライアントはNICE DCVセッションに接続する前に、NICE DCVがホストされているサーバーに対して認証する必要があります。クライアントによる認証が失敗した場合、クライアントをセッションに接続できません。クライアントがサーバーに対して認証を行わなくてもセッションに接続できるように、クライアント認証の要件を無効にすることができます。

NICE DCV は、次の認証方法をサポートしています。

- **system** — デフォルトの認証方法です。クライアント認証は基盤となるオペレーティングシステムに委任されます。Windows NICE DCVサーバーの場合、認証は `WinLogon` に委任されます。Linux NICE DCVサーバーの場合、認証は `PAM` に委任されます。クライアントは、NICE DCVセッションに接続するときにシステム認証情報を提供します。クライアントが NICE DCVサーバーに適切なサインイン認証情報を持っていることを確認します。
- **none** — NICE DCVセッションに接続するときにクライアント認証は必要ありません。NICE DCVサーバーは、セッションに接続しようとするすべてのクライアントへのアクセスを許可します。

クライアントがNICE DCVサーバーで使用される認証方法を認識していることを確認します。また、セッションに接続するために必要な情報がそろっていることを確認してください。

## トピック

- [Windows での認証の設定](#)
- [Linux での認証の設定](#)
- [外部認証システムによる認証の設定](#)

## Windows での認証の設定

NICE DCV サーバーの認証方法を変更するには、Windows レジストリエディタを使用して `authentication` パラメータを設定する必要があります。

Windows で認証方法を変更するには

1. Windows レジストリエディタを開きます。
2. `HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/security/` キーに移動し、認証パラメータを選択します。

レジストリキーに `authentication` パラメータが存在しない場合は、1 つ作成します。

- a. ナビゲーションペインで、`[authentication]` (認証) キーのコンテンツ (右クリック) メニューを開きます。次に、`[New]` (新規)、`[string value]` (文字列値) の順に選択します。
  - b. `[名前]` に `authentication` と入力し、`[Enter]` を押します。
3. `[authentication]` パラメータを開きます。`[値のデータ]` に、「`system`」または「`none`」と入力します。

4. [OK] を選択して Windows レジストリエディタを閉じます。

## Windows 認証情報プロバイダー

Windows 認証情報プロバイダーを使用すると、DCVサーバーに対して認証できる場合、ユーザーは Windows ログインをバイパスできます。

Windows 認証情報プロバイダーは、DCVauthenticationパラメータが に設定されている場合にのみサポートされますsystem。DCV authentication パラメータが に設定されている場合none、ユーザーはDCVサーバーに対して自動的に認証された後に Windows に手動でサインインする必要があります。

デフォルトでは、NICEDCVサーバーのインストール時に Windows 認証情報プロバイダーが有効になります。

Windows 認証情報プロバイダーを無効にするには

1. Windows レジストリエディタを開きます。
2. HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Authentication\Credential Providers\{8A2C93D0-D55F-4045-99D7-B27F5E263407} キーに移動します。
3. 編集、新規、DWORD値 を選択します。
4. 名前には、**Disabled** と入力します。
5. 値を開きます。[値のデータ] に 1 と入力し、[OK] を選択します。
6. Windows レジストリエディタを閉じます。

Windows 認証情報プロバイダーを再度有効にするには

1. Windows レジストリエディタを開きます。
2. HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Authentication\Credential Providers\{8A2C93D0-D55F-4045-99D7-B27F5E263407} キーに移動します。
3. [無効] の値を開きます。[値のデータ] に 0 と入力し、[OK] を選択します。
4. Windows レジストリエディタを閉じます。

## Linux での認証の設定

NICE DCV サーバーの認証方法を変更するには、`dcv.conf` ファイルで `authentication` パラメータを設定する必要があります。

Linux で認証方法を変更するには

1. `/etc/dcv/` に移動し、任意のテキストエディタで `dcv.conf` を開きます。
2. `authentication` パラメータを `[security]` セクションで見つけます。次に、既存の値を `system` または `none` のいずれかに置き換えます。

`[security]` セクションに `authentication` パラメータが存在しない場合は、次の形式を使用して追加します。

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

3. ファイルを保存して閉じます。

## PAM サービス

Linux では、`NICEDCVauthentication` パラメータがに設定されている場合 `system`、認証は PAM サービスを実行して実行されます。

デフォルトでは、NICE DCV サーバーによって実行される特権アクセス管理 (PAM) サービスは `dcv` です/`etc/pam.d/dcv`。

を使用してユーザーを認証 PAM するときによって実行されるステップを変更する場合は DCV、NICE の `authentication` セクションで `pam-service` パラメータを設定できます `dcv.conf`。

PAM サービスを変更するには

1. `root` として `/etc/pam.d` ディレクトリに進み、新しいファイル (`dcv-custom` など) を作成します。
2. 任意のテキストエディタを使用して `dcv-custom` ファイルを編集します。PAM サービスファイルの構文については、システムドキュメントを参照してください。
3. `/etc/dcv/` に移動し、任意のテキストエディタで `dcv.conf` を開きます。
4. `pam-service` パラメータを `[authentication]` セクションで見つけます。次に、既存のサービス名を新しい PAM サービス名に置き換えます。

[authentication] セクションに pam-service パラメータが存在しない場合は、次の形式を使用して手動で追加します。

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

#### Note

PAM サービス名は、で作成したファイルの名前と一致する必要があります/etc/pam.d。

5. ファイルを保存して閉じます。

## 外部認証システムによる認証の設定

DCV は、外部認証を使用するように設定できます。このプロセスと要件の詳細は「[外部認証を使用する](#)」を参照してください。

## NICE DCV 認証の設定

認可は、特定のNICE DCV機能に対するアクセス許可をNICE DCVクライアントに付与または拒否するために使用されます。NICE ではDCV、認証はアクセス許可ファイルを使用して設定されます。アクセス許可ファイルは、特定のユーザーがセッションに接続するときに使用できる特定のNICE DCV機能を定義します。

NICE DCV は、次の 2 種類のアクセス許可ファイルをサポートしています。

### トピック

- [デフォルトのアクセス許可ファイル](#)
- [カスタムアクセス許可ファイル](#)
- [アクセス許可ファイルでの作業](#)

## デフォルトのアクセス許可ファイル

セッションの作成時にカスタムアクセス許可ファイルを指定しない場合、デフォルトのアクセス許可ファイルはすべてのセッションに使用されます。デフォルトのアクセス許可ファイルは、セッション所有者のみに、すべての機能へのフルアクセスを許可します。

デフォルトのアクセス許可ファイルをカスタマイズして、カスタム承認を含めることができます。デフォルトのアクセス許可ファイルは、Windows NICE DCV サーバー C:\Program Files\NICE\DCV\Server\conf\default.perm 上の と Linux NICE DCV サーバー /etc/dcv/default.perm 上の にあります。

デフォルトのアクセス許可ファイルのカスタマイズについては、「[アクセス許可ファイルでの作業](#)」を参照してください。

## カスタムアクセス許可ファイル

カスタムアクセス許可ファイルを使用して、特定のユーザーまたはグループが NICE DCV セッションに接続するときにアクセスできる機能を定義できます。カスタムアクセス許可ファイルを使用する場合、デフォルトのアクセス許可ファイルを上書きします。

カスタムアクセス許可ファイルを使用するには、まずアクセス許可ファイルを作成する必要があります。次に、`dcv create-session` コマンドで `--permissions-file` オプションを使用してセッションを開始するときに指定します。セッション開始の詳細については、「[NICE DCV セッションの開始](#)」を参照してください。

カスタムアクセス許可ファイルの作成については、「[アクセス許可ファイルでの作業](#)」を参照してください。

## アクセス許可ファイルでの作業

カスタムアクセス許可ファイルを作成したり、任意のテキストエディタを使用して既存のアクセス許可ファイルを更新することができます。アクセス許可ファイルは、通常、次の形式になります。

```
#import file_to_import

[groups]
group_definitions

[aliases]
```

```
alias_definitions
```

```
[permissions]
```

```
user_permissions
```

次のセクションでは、アクセス許可ファイルを更新または作成するときにセクションを設定する方法を説明します。

## 目次

- [アクセス許可ファイルをインポートする](#)
- [グループを作成する](#)
- [エイリアスを作成する](#)
- [アクセス許可を追加する](#)

## アクセス許可ファイルをインポートする

imports セクションは一般的に、アクセス許可ファイルの最初のセクションです。このセクションでは、既存のアクセス許可ファイルを参照したり含めたりすることができます。また、これを使用して、以前に定義したNICE DCVアクセス許可をアクセス許可ファイルに組み込むこともできます。

アクセス許可ファイルには複数のインポートを含めることができます。インポートされたアクセス許可ファイルは、他のアクセス許可ファイルをインポートする場合があります。

アクセス許可ファイルをユーザーのアクセス許可ファイルにインポートするには

- #import ステートメントを使用して、絶対パスまたは相対パスでファイルの場所を指定します。
  - Windows NICE DCVサーバー :

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

- Linux NICE DCVサーバー :

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

## 例

次のステートメントでは、絶対パスを使用して `dcv-permissions.file` という名前のアクセス許可ファイルをインポートします。Windows NICE DCV サーバー NICE DCV のインストールフォルダにあります。

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

## グループを作成する

アクセス許可ファイルの `[groups]` セクションを使えば、類似するユースケースやアクセス許可要件を持つユーザーのためにユーザーグループを定義することができます。グループには特定のアクセス許可が割り当てられます。グループに割り当てられたアクセス許可は、グループに含まれるすべてのユーザーに適用されます。

アクセス許可ファイルでグループを作成するには、まずファイルにグループのセクション見出しを追加します。

```
[groups]
```

次に、セクション見出しの下にグループを作成することができます。グループを作成するには、グループ名を指定してから、カンマ区切りリストでグループメンバーを指定します。グループメンバーは、個々のユーザー、他のグループ、およびオペレーティングシステムのユーザーグループです。

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

ユーザーをグループに追加するには

ユーザー名を指定します。

### Note

ユーザー名の先頭を `user:` で始めます。Windows ドメインユーザー名にドメイン名を含めることができます。

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

既存のグループをグループに追加するには

先頭が `group:` で始まるグループ名を指定します。

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

オペレーティングシステムのユーザーグループをグループに追加するには (Linux NICE DCVサーバーのみ)

先頭が `osgroup:` で始まるグループの名前を指定します。

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

## 例

次の例では、`groups` セクション見出しを追加して、`my-group` という名前のグループを作成します。このグループには個々のユーザーが含まれます。これらには `john` と `jane` という名前が付けられています。1 つは `observers` という名前の既存のグループです。もう 1 つは、`guests` という名前のオペレーティングシステムユーザーグループです。

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

## エイリアスを作成する

アクセス許可ファイルの `[aliases]` セクションを使用して、一連の NICE DCV 機能を作成できます。エイリアスが定義されると、エイリアスを使用するアクセス許可をグループまたは個々のユーザーに付与もしくは拒否することができます。アクセス許可の付与または拒否により、エイリアスに含まれているすべての機能に対するアクセス許可が付与、または拒否されます。

アクセス許可ファイルでエイリアスを作成するには、まずファイルにエイリアスのセクション見出しを追加します。

```
[aliases]
```

その後、セクション見出しの下にエイリアスを作成することができます。新しいエイリアスを作成するには、エイリアス名を指定してから、カンマ区切りリストでエイリアスメンバーを指定します。エイリアスメンバーは、個々の NICE DCV 機能または他のエイリアスにすることができます。

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

## 例

次の例では、aliases セクション見出しを追加して、file-management という名前のエイリアスを作成します。これには、file-upload 機能、file-download 機能、および、clipboard-management という名前の既存のエイリアスが含まれます。

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

## アクセス許可を追加する

アクセス許可ファイルの [permissions] セクションでは、特定の機能やエイリアスへの、ユーザーおよびグループアクセスを制御することができます。

ユーザーのアクセス許可ファイルにアクセス許可を追加するには、まずファイルにアクセス許可のセクション見出しを追加します。

```
[permissions]
```

その後、セクション見出しの下にアクセス許可を追加することができます。アクセス許可を追加するには、制御するアクター、適用されるルール、および許可を適用する機能を指定します。

```
actor rule features
```

アクターは、ユーザー、グループ、またはオペレーティングシステムグループのいずれかになります。グループにはプレフィックス group: を付ける必要があります。オペレーティングシステムグループには osgroup: をプレフィックスとして付ける必要があります。NICE DCV には、セッション所有者を参照するために使用できる組み込み%owner%リファレンスが含まれています。これは、任意のユーザーの参照に使用できる組み込み %any% リファレンスを参照する場合にも使用できます。

以下のルールは、アクセス許可ステートメントで使用できます。

- allow — 機能へのアクセスを許可します。
- disallow — 機能へのアクセスを拒否しますが、後からアクセス許可で上書きできます。
- deny — 機能へのアクセスを拒否します。後からアクセス許可で上書きすることはできません。

機能には、個々のNICE DCV機能、エイリアス、またはその両方の組み合わせを含めることができます。機能のリストはスペースで区切る必要があります。NICE DCV には、すべてのNICE DCV機能を含む組み込みエイリアス builtin が含まれています。

アクセス許可ファイルでは、以下の機能を参照できます。

- audio-in — クライアントからNICE DCVサーバーにオーディオを挿入します。
- audio-out — クライアントでNICE DCVサーバーオーディオを再生します。
- builtin — すべての機能です。
- clipboard-copy — NICE DCVサーバーからクライアントクリップボードにデータをコピーします。
- clipboard-paste — クライアントクリップボードからNICE DCVサーバーにデータを貼り付けます。
- display — NICE DCVサーバーからビジュアルデータを受信します。
- extensions-client — NICE DCVクライアントにインストールされた拡張機能を起動することを許可します。
- extensions-server — NICE DCVサーバーにインストールされている拡張機能を起動することを許可します。
- file-download — セッションストレージからファイルをダウンロードします。
- file-upload — セッションストレージにファイルをアップロードします。
- gamepad — セッションでクライアントコンピュータに接続されているゲームパッドを使用します。バージョン 2022.0 NICE DCV 以降でサポートされています。
- keyboard — クライアントキーボードからNICE DCVサーバーに入力します。
- keyboard-sas — 安全なアテンションシーケンス (CTRL+Alt+Del) を使用します。keyboard 機能が必要です。バージョン 2017.3 NICE DCV 以降でサポートされています。
- mouse — クライアントポインタからNICE DCVサーバーへの入力。
- pointer — NICE DCVサーバーマウスの位置イベントとポインタの形状を表示します。バージョン 2017.3 NICE DCV 以降でサポートされています。
- printer — NICE DCVサーバーからクライアントに PDFs または XPS ファイルを作成します。
- screenshot — リモートデスクトップのスクリーンショットを保存します。バージョン 2021.2 NICE DCV 以降でサポートされています。

screenshot 認可を外す際に clipboard-copy アクセス許可を無効にすることをお勧めしました。これにより、ユーザーがサーバーのクリップボードでスクリーンショットをキャプチャしてクライアントに貼り付けることができなくなります。screenshot 認可が拒否された場合、Windows と macOS は、外部ツールを使ったクライアントのスクリーンショットのキャプチャも実行できなくなります。例えば、NICE DCV クライアントウィンドウで Windows スニッピングツールを使用すると、黒いイメージになります。

- smartcard — クライアントからスマートカードを読み込みます。
- stylus — 3D ポインティングUSBデバイスやグラフィックタブレットなどの特殊なデバイスからの入力。
- touch — ネイティブタッチイベントを使用します。バージョン 2017.3 DCV 以降でサポートされています。
- unsupervised-access— コラボレーションセッションで、ユーザーによる所有者なしのアクセスを設定するために使用します。
- usb — クライアントからUSBデバイスを使用します。
- webcam — セッションでクライアントコンピュータに接続されているウェブカメラを使用します。バージョン 2021.0 NICE DCV 以降でサポートされています。
- webauthn-redirect— Webauthn リクエストをリモートブラウザからローカルクライアントにリダイレクトします。バージョン 2023.1 NICE DCV 以降でサポートされています。

## 例

次の例では、アクセス許可のセクション見出しを追加して、4 つのアクセス許可を追加します。最初のアクセス許可は、ユーザーに john、display、および file-upload 機能への file-download という名前のアクセスを付与します。2 番目のアクセス許可は、audio-in 機能および audio-out 機能、clipboard-management 機能エイリアスへの observers グループアクセスを拒否します。3 番目のアクセス許可は、guests オペレーティングシステムグループに clipboard-management および file-management エイリアスへのアクセスを付与します。4 番目のアクセス許可は、セッション所有者にすべての機能へのアクセスを付与します。

```
[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
```

# NICE DCV セッションの管理

クライアントを NICE DCV セッションに接続できるようにするには、NICE DCV サーバーでセッションを作成する必要があります。アクティブなセッションがある場合にのみ、クライアントを NICE DCV サーバーに接続できます。

各 NICE DCV セッション には次の属性があります。

- セッション ID — NICE DCV サーバーで特定のセッションを識別するために使用します。
- 所有者 — セッションを作成した NICE DCV ユーザー。デフォルトでは、所有者のみがセッションに接続できます。

セッションに接続するのに、NICE DCV クライアントはこの情報を必要とします。

## トピック

- [NICE DCV セッションの概要。](#)
- [コマンドラインツールを使用した NICE DCV セッションの管理](#)
- [NICE DCV セッションの開始](#)
- [NICE DCV セッションの停止](#)
- [実行中の NICE DCV セッションの管理](#)
- [セッションのタイムゾーンの管理](#)
- [NICE DCV セッションの表示](#)
- [NICE DCV セッションのスクリーンショットの取得](#)

## NICE DCV セッションの概要。

NICE DCV には、コンソールセッションと仮想セッションの 2 種類のセッションがあります。次の表はこれら 2 種類のセッションの違いをまとめたものです。

セッションタイプ	サポート	複数のセッション	必要なアクセス許可	直接画面キャプチャー	GPU加速 OpenGLサポート
コンソール	Linux と Windows の NICE DCV サーバー	不可。各サーバーで許容されるコンソールセッションは 1 つのみ	セッションの開始と終了を実行できるのは管理者のみ	可	可。追加ソフトウェアなし
仮想	Linux NICE DCV サーバーのみ	可。1 つのサーバーで複数の仮想セッションが許容される	すべてのユーザーがセッションの開始と終了を実行できる	不可。各仮想セッションに対して専用の X サーバ (Xdcv) が実行される。X サーバーから画面がキャプチャーされる。	可。ただし DCV-GL パッケージが必要

### Note

同一の NICE DCV サーバーにおいて、コンソールセッションと仮想セッションを同時に実行することはできません。

## コンソールセッション

コンソールセッションは、Windows および Linux NICE DCV サーバーでサポートされています。Windows NICE DCV サーバーを使用している場合は、コンソールセッションしか使用できません。

NICE DCV サーバーで一度にホストできるコンソールセッションは 1 つのみです。コンソールセッションは、Windows NICE DCV サーバーの管理者および Linux NICE DCV サーバーのルートユーザーによって作成および管理されます。

コンソールセッションでは、NICE DCV によりデスクトップ画面のコンテンツが直接キャプチャーされます。サーバーが GPU で設定されている場合、NICE DCV コンソールセッションは GPU に直接アクセスします。

## 仮想セッション

仮想セッションは、Linux NICE DCV サーバーでのみサポートされます。

同一の NICE DCV サーバーにおいて、複数の仮想セッションを同時にホストすることができます。仮想セッションの作成と管理は NICE DCV ユーザーが行います。NICE DCV ユーザーは、自分が作成したセッションしか管理できません。ルートユーザーは NICE DCV サーバーで現在実行されているすべての仮想セッションを管理できます。

仮想セッションの場合、NICE DCV では、X サーバーインスタンス `Xdcb` が起動され、X サーバ内でデスクトップ環境が実行されます。NICE DCV により、仮想セッションごとに新しい専用 X サーバーインスタンスが起動されます。各仮想セッションは、X サーバーインスタンスによって提供される表示を使用します。

### Note

NICE DCV では各仮想セッションで独立した `Xdcb` 表示が使用されますが、ユーザーのホームフォルダ内のファイル、D-Bus サービス、デバイスなど、他の多くのシステムリソースはユーザーごとに表示されます。そのため、同じユーザーの複数の仮想セッションで共有され、アクセスできます。

共有リソースに関する懸念を軽減するようにオペレーティングシステムを設定していない限り、同じ NICE DCV サーバー上で同じユーザーの複数の仮想セッションを同時に実行しないでください。

`dcb-g1` パッケージがインストールされており、ライセンスを取得している場合、NICE DCV 仮想セッションにおいてサーバーの GPU へのアクセス権が共有されます。複数の仮想セッション間でハードウェアベースの OpenGL を共有するには、`dcb-g1.conf` ファイルを設定して、仮想 X サーバーインスタンスを GPU に接続する必要があります。

## コマンドラインツールを使用した NICE DCV セッションの管理

NICE DCV サーバーには、NICE DCV セッションの開始、停止、表示に使用できるコマンドラインツールが含まれています。

## Windows NICE DCV サーバーでのコマンドラインツールの使用

Windows NICE DCV サーバーでコマンドラインツールを使用するには、NICE DCV インストールディレクトリからコマンドを実行するか、または NICE DCV ディレクトリを PATH 環境変数に追加する必要があります。NICE DCV ディレクトリを PATH 環境変数に追加すると、任意のディレクトリからコマンドを使用できます。

NICE DCV インストールディレクトリからコマンドラインツールを使用するには

dcv.exe ファイルがあるフォルダ (デフォルトで C:\Program Files\NICE\DCV\Server\bin\) に移動し、コマンドプロンプトウィンドウを開きます。

または、別のディレクトリからコマンドを実行するときにフルパスを指定することもできます。

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

NICE DCV ディレクトリを PATH 環境変数に追加するには

1. ファイルエクスプローラーで、[この PC] を右クリックし、[プロパティ] を選択します。
2. [システムの詳細設定] を選択します。
3. [Advanced] (詳細設定) タブで、[Environment Variables] (環境変数) を選択します。
4. [システム環境変数] セクションで、[Path] 変数を選択し、[編集] を選択します。
5. [New] (新規) を選択し、NICE DCV インストールディレクトリ内の bin フォルダへのフルパス (例: C:\Program Files\NICE\DCV\Server\bin\) を指定します。
6. [OK] を選択し、環境変数ウィンドウを閉じます。

## Linux NICE DCV サーバーでのコマンドラインツールの使用

Linux NICE DCV サーバーでは、コマンドラインツールは、\$PATH 環境変数で自動的に設定されます。このツールは任意のフォルダから使用できます。ターミナルウィンドウを開き、実行するコマンドを入力します。

### コマンドラインツールの使用

次の表に、使用可能なコマンドラインツールのオプションを示します。このリストは、 を呼び出す --help ときに を使用して取得できます dcv。各コマンドの使用の詳細については、使用情報が必要なコマンドの --help 後に を渡します。例: dcv create-session --help。

コマンド	説明
<code>create-session</code>	新しい DCV セッションを作成する
<code>close-session</code>	アクティブな DCV セッションを閉じる
<code>describe-session</code>	DCV セッションの説明
<code>list-sessions</code>	アクティブな DCV セッションを一覧表示する
<code>list-connections</code>	DCV セッションのクライアント接続を一覧表示する
<code>close-connection</code>	アクティブなクライアント接続を閉じる
<code>get-screenshot</code>	DCV コンソールのスクリーンショットを取得する
<code>set-display-layout</code>	アクティブな DCV セッションの表示レイアウトを設定する
<code>set-name</code>	DCV セッションの名前を設定する
<code>set-permissions</code>	アクティブな DCV セッションのアクセス許可を設定する
<code>set-storage-root</code>	アクティブな DCV セッションのストレージルートを設定する
<code>reload-licenses</code>	実行中のすべてのセッションのライセンスを強制的に再ロードする

コマンド	説明
get-config	サーバー設定の取得
list-endpoints	DCV エンドポイントを一覧表示する
set-config	サーバー設定の設定
version	DCV のバージョンを表示する
help	ヘルプを表示する

## NICE DCV セッションの開始

デフォルトを使用して [Windows NICE DCV サーバーをインストール](#)すると、サーバーのインストール後に[コンソールセッション](#)が自動的に作成され、アクティブになります。デフォルトのコンソールセッションは Administrator によって所有されており、デフォルトのセッション ID は console です。このセッションを使用することも、[終了して新しいセッションを作成](#)することもできます。

NICE DCV サーバーのインストール時に自動的にコンソールセッションを作成しなかった場合は、手動でセッションを作成する必要があります。NICE DCV サーバーのインストール後はいつでも、[コンソールセッションの自動作成](#)を有効または無効にすることができます。

### Note

Linux NICE DCV サーバーでは、インストール後にデフォルトのコンソールセッションが取得されません。

オンプレミスまたは代替のクラウドベースサーバーでフローティングライセンスを使用していて、ライセンスでサポートされている同時セッションの最大数を越えたとします。その場合は no licenses エラーが発生する可能性があります。このエラーが発生した場合は、未使用のセッションを停止してライセンスをリリースしてから再試行してください。

セッションを開始するには、NICE DCV サーバー を実行している必要があります。詳細については、「[NICE DCV サーバーの起動](#)」を参照してください。

## トピック

- [コンソールセッションおよび仮想セッションの手動開始](#)
- [自動コンソールセッションの有効化](#)

## コンソールセッションおよび仮想セッションの手動開始

NICE DCV セッションはいつでも開始することができます。一度に 1 つのコンソールセッションしか実行できません。Linux NICE DCV サーバーを使用している場合は、複数の仮想セッションを同時に実行できます。

特に Windows NICE DCV サーバーを使用している場合は、セッションを作成する前に `dcv list-sessions` を実行することをお勧めします。

Windows または Linux の NICE DCV サーバーでコンソールまたは仮想セッションを作成するには、`dcv create-session` コマンドを使用します。

## トピック

- [Syntax](#)
- [オプション](#)
- [例](#)

## Syntax

セッションを開始するコマンドの最小限の構文は次のとおりです。

```
dcv create-session session_ID
```

すべてのオプションを含む完全な構文は以下のとおりです。

```
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**

\ 記号は、コマンドを複数行に分割する構文を表しています。

dcv create-session --help を使用して構文のクイックリファレンスを表示することもできます。

## オプション

以下のオプションは dcv create-session コマンドで使用できます。

### --type

このオプションは、Linux NICE DCV サーバーでのみサポートされます。作成されるセッションのタイプを指定します。console または virtual のいずれかになります。

型: 文字列

許可される値: console | virtual

必須: いいえ

### --name

セッションの名前を指定します。セッション名は、最大 256 文字までの任意の文字列とします。文字列が 256 文字を超えると、コマンドは失敗します。セッション名は、実行中のセッション間で一意である必要はありません。

セッション名は dcv set-name コマンドを使用していつでも変更できます。詳細については、「[セッション名の管理](#)」を参照してください。

型: 文字列

必須: はい

### --user

このオプションは Linux NICE DCV セッションの仮想セッションでのみサポートされています。この値は、セッションの作成に使用されるユーザーです。ルートユーザーのみが他のユーザーを偽装できます。

タイプ: 文字列

必須: いいえ

### --owner

セッションの所有者を指定します。省略した場合、デフォルトは現在サインインしているユーザーです。

タイプ: 文字列

必須: いいえ

### --permissions-file

カスタムアクセス権限ファイルへのパスを指定します。省略した場合、デフォルトはサーバーのデフォルトです。

タイプ: 文字列

必須: いいえ

### --storage-root

セッションストレージに使用されるフォルダのパスを指定します。

%home% を使用して、現在サインインしているユーザーのホームディレクトリを指定できます。たとえば、以下では、セッションストレージのディレクトリを Windows サーバーで c:\Users\*username*\storage\、または Linux サーバーで \$HOME/storage/ に設定します。

```
--storage-root %home%/storage/
```

#### Note

指定したサブディレクトリが存在しない場合、セッションストレージは無効になります。

タイプ: 文字列

必須: いいえ

### --gl

このオプションは Linux NICE DCV セッションの仮想セッションでのみサポートされています。デフォルトの dcv-gl 状態をオーバーライドします。on または off のいずれかになります。

型: 文字列

許可される値: on | off

必須: いいえ

### **--max-concurrent-clients**

セッションへの接続が許可されている NICE DCV クライアントの最大数を指定します。省略した場合、デフォルトは無制限の接続です。

タイプ: 整数

必須: いいえ

### **--init**

このオプションは Linux NICE DCV サーバーの仮想セッションでのみサポートされています。カスタム init スクリプトへのパスを指定します。このスクリプトを使用して特定のデスクトップ環境を起動し、セッションの開始時に特定のアプリケーションを自動的に起動することができます。スクリプトは実行可能である必要があります。省略した場合、デフォルトはデフォルトのデスクトップ環境を起動するスクリプトです。

タイプ: 文字列

必須: いいえ

### ***session ID***

コマンドの最後にセッションの ID を指定します。

型: 文字列

必須: はい

## 例

### 例 1 - コンソールセッション

以下のコマンドにより、`dcv-user` が所有するコンソールセッションが作成されます。セッション ID は一意の `my-session` で、セッション名は `my graphics session` です。 `perm-file.txt` という名前のアクセス許可ファイルも指定されます。

- Windows NICE DCV サーバー

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

- Linux NICE DCV サーバー

```
$ sudo dcv create-session \
--type=console \
--owner dcv-user \
--name "my graphics session" \
--permissions-file perm-file.txt \
my-session
```

## 例 2 - 仮想セッション (Linux NICE DCV サーバーのみ)

以下のコマンドでは、意図されたセッション所有者 `dcv-user` になりすますために、ルートユーザーを使用して仮想セッションを作成します。ルートユーザーによって作成されていても、セッションの所有者は `dcv-user` です。

```
$ sudo dcv create-session \
--owner dcv-user \
--user dcv-user \
my-session
```

## 例 3 - 仮想セッション (Linux NICE DCV サーバーのみ)

以下のコマンドでは、作成したユーザーが所有する仮想セッションが作成されます。

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

## 自動コンソールセッションの有効化

自動コンソールセッションを有効にすると、NICE DCV サーバーが起動するたびに、コンソールセッションが自動的に作成されます。自動コンソールセッションは、`owner` 設定パラメータで指定された NICE DCV ユーザーによって所有されています。そのセッション ID は常に `console` です。

自動コンソールセッションに影響するパラメータには他にも、max-concurrent-clients、permissions-file、および storage-root があります。これらのパラメータの詳細については、「[session-management/automatic-console-session パラメータ](#)」を参照してください。

 Note

NICE DCV では自動仮想セッションがサポートされません。

## Windows NICE DCV server

Windows NICE DCV サーバーで自動コンソールセッションを有効にするには

1. Windows レジストリエディタを開きます。
2. HKEY\_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/session-management キーに移動します。
3. create-session パラメータの作成
  - a. ナビゲーションペインで、[session-management] キーのコンテキスト (右クリック) メニューを開き、[New] (新規)、[DWORD (32-bit) Value] (DWORD (32 ビット) 値) の順に選択します。
  - b. [名前] に create-session と入力し、[Enter] を押します。
  - c. [create-session] パラメータを開きます。[値のデータ] に 1 と入力し、[OK] を選択します。
4. HKEY\_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/session-management/automatic-console-session キーに移動します。
5. owner パラメータの作成
  - a. ナビゲーションペインで、automatic-console-session キーのコンテキスト (右クリック) メニューを開き、新規、文字列値 を選択します。
  - b. [名前] に owner と入力し、[Enter] を押します。
  - c. [owner] パラメータを開きます。[値のデータ] にセッションの所有者名を入力し、[OK] を選択します。
6. [OK] を選択して Windows レジストリエディタを閉じます。
7. NICE DCV サーバーを [停止](#) して [再起動](#) します。

## Linux NICE DCV server

Linux NICE DCV サーバー で自動コンソールセッションを有効にするには

1. /etc/dcv/ に移動し、任意のテキストエディタで dcv.conf を開きます。
2. 次の形式を使用して、create-session および owner パラメータを [session-management/automatic-console-session] セクションに追加します。

```
[session-management]
create-session = true

[session-management/automatic-console-session]
owner="session-owner"
```

3. ファイルを保存して閉じます。
4. NICE DCV サーバーを[停止](#)して[再起動](#)します。

## NICE DCV セッションの停止

コンソールセッションを停止できるのは、Windows NICE DCVサーバーでは管理者、Linux NICE DCV サーバーではルートユーザーだけです。Linux NICE DCV サーバーの仮想セッションは、ルートユーザー、またはこのセッションを作成した NICE DCV ユーザーのみが停止できます。

### Note

セッションを停止すると、そのセッションで実行されているすべてのアプリケーションが閉じます。

Windows または Linux の NICE DCV サーバーでコンソールセッションや仮想セッションを停止するには、dcv close-session コマンドを使用して一意のセッション ID を指定します。

### トピック

- [Syntax](#)
- [例](#)

## Syntax

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

### 例

例えば、次のコマンドでは一意の ID `my-session` を使用してセッションが停止されます。

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

## 実行中の NICE DCV セッションの管理

次のセクションでは、実行中の NICE DCV セッションの管理について説明します。

### トピック

- [NICE DCV セッションストレージの管理](#)
- [NICE DCV セッション認可の管理](#)
- [NICE DCV セッション表示レイアウトの管理](#)
- [セッション名の管理](#)

## NICE DCV セッションストレージの管理

セッションストレージは NICE DCV サーバーにあるディレクトリで、NICE DCV セッションに接続されているクライアントによるアクセスが可能です。

NICE DCV サーバーでセッションストレージが有効になっている場合、`dcv set-storage-root` コマンドを使用して、セッションストレージに使用するディレクトリを指定できます。NICE DCV サーバーでのセッションストレージの有効化の詳細については、「[セッションストレージの有効化](#)」を参照してください。

セッションストレージパスを設定するには、`dcv set-storage-root` コマンドを使用して、セッション ID と、使用するディレクトリへのパスを指定します。

### トピック

- [Syntax](#)
- [オプション](#)

- [例](#)

## Syntax

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

ディレクトリパスについては、%home% を使用して、現在サインインしているユーザーのホームディレクトリを指定できます。たとえば、%home%/storage/ パスは Windows サーバーで c:\Users\*username*\storage\ に解決されます。Linux サーバでは \$HOME/storage/ に解決されます。

## オプション

以下のオプションは dcv set-storage-root コマンドで使用できます。

### --session

ストレージディレクトリを指定するセッション ID。

型: 文字列

必須: はい

## 例

### Windows NICE DCV サーバーの例

次の例では、セッション ID が my-session であるセッションのストレージパスを c:\session-storage に設定します。

```
C:\> dcv set-storage-root --session my-session c:\session-storage
```

### Linux NICE DCV サーバーの例

次の例では、セッション ID が my-session であるセッションの場合、現在のユーザーのホームディレクトリにある session-storage という名前のディレクトリへのストレージパスをに設定します。

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

## NICE DCV セッション認可の管理

認可は、特定の NICE DCV 機能に対して、NICE DCV クライアントのアクセス許可を付与する場合や拒否する場合に使用されます。通常、認可は NICE DCV セッションの起動時に設定されます。ただし、実行中のセッションのアクセス許可を編集することはできません。NICE DCV 認可の詳細については、「[NICE DCV 認証の設定](#)」を参照してください。

実行中のセッションのアクセス許可に変更を加えるには、`dcv set-permissions` コマンドを使用します。

トピック

- [Syntax](#)
- [オプション](#)
- [例](#)

### Syntax

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

`--none`、`--reset-builtin`、`--file` のいずれかを指定する必要があります。

### オプション

以下のオプションは `dcv set-permissions` コマンドで使用できます。

`--session`

アクセス許可を設定するセッションの ID を指定します。

`--reset-builtin`

セッションのアクセス許可をデフォルトのセッションアクセス許可にリセットします。デフォルトのアクセス許可では、セッション所有者のみにすべての機能へのフルアクセス権を付与します。

`--none`

セッションのすべてのアクセス許可を取り消します。

## --file

カスタムアクセス許可ファイルへのパスを指定します。指定したファイルが空の場合、すべてのアクセス許可が取り消されます。カスタムアクセス許可ファイルの作成の詳細については、「[アクセス許可ファイルでの作業](#)」を参照してください。

## 例

### 例 1 — すべてのアクセス許可の取り消し

次の例では、ID が `my-session` であるセッションのすべてのクライアントアクセス許可を取り消します。

```
C:\> dcv set-permissions --session my-session --none
```

### 例 2 — カスタムアクセス許可の指定

次の例では、ID が `my-session` であるセッションに対して、`perm-file.txt` という名前のカスタムアクセス許可ファイルを指定します。このファイルは `c:\dvc\` ディレクトリにあります。

```
C:\> dcv set-permissions --session my-session --file c:\dvc\perm-file.txt
```

### 例 3 - アクセス許可のリセット

次の例では、ID が `my-session` であるセッションのアクセス許可をデフォルトにリセットします。

```
C:\> dcv set-permissions --session my-session --reset-builtin
```

## NICE DCV セッション表示レイアウトの管理

実行中の NICE DCV セッションの表示レイアウトを設定できます。この表示レイアウトは、クライアントがセッションに接続するとき使用されるデフォルト設定を指定します。ただし、クライアントは、NICE DCV クライアント設定またはオペレーティングシステムのネイティブ表示設定を使用して、手動でレイアウトを上書きできます。

ホストサーバーのハードウェアとソフトウェアの設定において、指定された解像度または画面数がサポートされていない場合、NICE DCV サーバーでは指定された表示レイアウトが適用されません。

NICE DCV では、設定とサーバーシステム設定に従って解像度を設定できます。

- ウェブクライアントの解像度は、デフォルトで 1920 x 1080 ( web-client-max-head解像度サーバー設定から) に制限されています。
- ネイティブクライアントは、デフォルトで 4096 x 2160 ( から) に制限されています max-head-resolution。

使用可能な解像度とモニターの数にはサーバーの設定によって異なることに注意してください。最適なパフォーマンスを得るには、[前提条件ガイド](#)に従ってシステム環境とドライバーを適切に設定してください。

#### Note

ネイティブクライアントの場合、最大 4 台のモニターを使用できます。  
ウェブクライアントの場合、最大 2 台のモニターを使用できます。  
どの設定でも、高解像度または最大モニター数を超えるモニターはサポートされていません。

## トピック

- [表示レイアウトの制限](#)
- [表示レイアウトを指定する](#)
- [表示レイアウトの表示](#)

## 表示レイアウトの制限

指定した範囲外の表示レイアウトをクライアントがリクエストしないように NICE DCV サーバーを設定できます。表示レイアウトの変更を制限するには、次の NICE DCV サーバーパラメータを設定します。

- [enable-client-resize](#) — クライアントが表示レイアウトを変更できないようにするには、このパラメータを false に設定します。
- [min-head-resolution](#) および [max-head-resolution](#) — 許容される最小解像度と最大解像度をそれぞれ指定します。
- [web-client-max-head-resolution](#) — ウェブブラウザクライアントの最大許容解像度を指定します。max-head-resolution 制限は、web-client-max-head-resolution 制限の上に適用されます。デフォルトでは、ウェブブラウザクライアントの最大解像度は 1920x1080 です。よ

り高い解像度を指定すると、ウェブブラウザとクライアントコンピュータの仕様によっては、パフォーマンスの問題が発生する可能性があります。

- [max-num-heads](#) - 最大表示数を指定します。
- [max-layout-area](#) - 画面領域に許可される最大ピクセル数を指定します。総画面面積 (ピクセル) が指定された値を超えているリクエストは無視されます。

これらのパラメータの詳細については、パラメータリファレンスの「[display パラメータ](#)」を参照してください。

## 表示レイアウトを指定する

実行中の NICE DCV セッションの表示レイアウトを設定するには

`dcv set-display-layout` コマンドを使用して、表示レイアウトと表示レイアウト記述子を設定するセッションを指定します。

```
dcv set-display-layout --session session-id display-layout-descriptor
```

表示レイアウト記述子は、表示の数と、各表示の解像度と位置オフセットを指定します。説明は、次の形式で指定する必要があります。

```
widthxheight+|-x-position-offset+|-y-position-offset
```

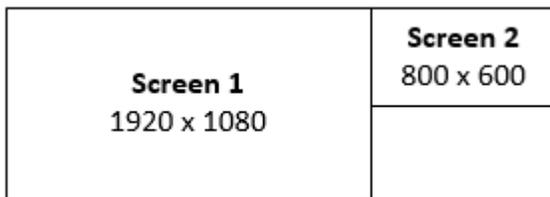
複数の画面を指定する場合は、画面記述子をカンマで区切ります。画面位置のオフセットは、画面 1 に対する画面の左上隅の位置を指定します。画面の位置オフセットを指定しない場合、デフォルトは  $x=0$  および  $y=0$  です。

### Important

複数の画面を指定する場合は、画面の重なりを避けるために、各画面の位置オフセットを適切に設定してください。

たとえば、次の表示レイアウト記述子は 2 つの画面を指定します。

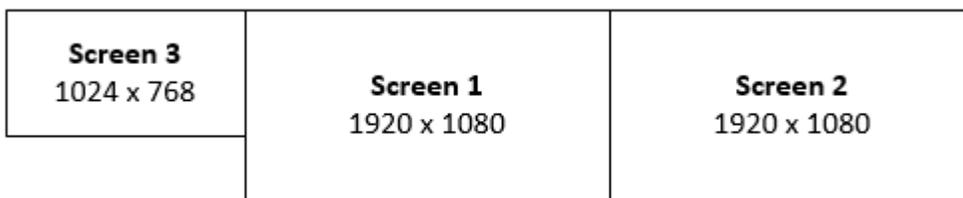
- 画面 1: 解像度 1920x1080、オフセット  $x=0$ 、 $y=0$
- 画面 2: 解像度 800x600、オフセット  $x=1920$ 、 $y=0$ 、画面 1 の右側に表示されます。



```
1920x1080+0+0,800x600+1920+0
```

次の表示レイアウト記述子は、3つの画面を指定します。

- 画面 1: 解像度 1920x1080、オフセット x=0、y=0
- 画面 2: 解像度 1920x1080、オフセット x=1920、y=0、画面 1 の右側に表示されます。
- 画面 3: 解像度 1024x768、オフセット x=-1024、y=0、画面 1 の左側に表示されます。



```
1920x1080+0+0,1920x1080+1920+0,1024x768-1024+0
```

## 表示レイアウトの表示

セッションの表示レイアウトを表示するには

dcv describe-session コマンドを使用して、出力の display layout 要素を確認します。詳細については、「[NICE DCV セッションの表示](#)」を参照してください。

## セッション名の管理

実行中のセッションの名前は、いつでも変更できます。特定のセッション名を使用すれば、その名前に基づいてセッションをすばやく識別できます。セッション名は、実行中のセッション間で一意である必要はありません。

実行中のセッションの名前を変更するには、dcv set-name コマンドを使用します。

### トピック

- [Syntax](#)

- [オプション](#)
- [例](#)

## Syntax

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

--name または --none のどちらかを指定する必要があります。

## オプション

以下のオプションは dset-name コマンドで使用できます。

### --session

名前を設定するセッションの ID。

型: 文字列

必須: はい

### --name

セッションに割り当てる名前。セッションに名前を割り当てる場合にのみ、このオプションを指定します。名前を削除する場合は、このパラメータを省略します。セッション名の長さは 256 文字以下にしてください。パスワードには、文字、数字、特殊文字を使用できます。指定した文字列が 256 文字を超えると、コマンドは失敗します。

タイプ: 文字列

必須: いいえ

### --none

セッションから既存の名前を削除するには、このパラメータを指定します。セッション名を削除しない場合は、このオプションを省略します。

必須: いいえ

## 例

### 例 1 - セッション名の変更

次の例では、ID が `my-session` であるセッションの名前を `my graphics session` に設定します。

```
$ dcv set-name --session my-session --name "my graphics sessions"
```

## 例 2 - セッション名の削除

次の例では、ID が `my-session` であるセッションの名前を削除します。

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

## セッションのタイムゾーンの管理

DCV を使用すると、セッションのオーナーとユーザーは DCV サーバーの場所またはユーザーの現在の場所を反映するようにセッションのタイムゾーンを設定できます。

### タイムゾーンリダイレクトの有効化

特定のセッションのすべてのユーザーに対してこの機能を有効または無効にできます。

1. [enable-timezone-redirect](#) パラメータを次のいずれかの値に設定します。

- `always-on`: タイムゾーンのリダイレクトは常に有効になります。

機能がオンになり、セッションにクライアントのタイムゾーン情報が表示されます。ユーザーは機能をオフにすることはできません。

- `always-off`: タイムゾーンのリダイレクトは常に無効になります。

機能はオフになり、セッションには独自のタイムゾーン情報が表示されます。ユーザーは機能をオンにすることはできません。

- `client-decides`: タイムゾーンのリダイレクトはデフォルトで有効になります。

セッションではこの機能が有効になり、クライアントのタイムゾーンが表示されます。ユーザーはこれを無効にして、サーバーのタイムゾーンを表示することもできます。

### Note

この設定は標準のデフォルト設定です。

**Note**

セッション内の個々のユーザーのみにこの機能が必要である場合は、まず全ユーザーに対して中央パラメーターを設定します。次に [アクセス許可を追加する](#) でカスタム権限ファイルを作成し、各ユーザーの権限を個別に調整する必要があります。

2. 変更内容を反映するには該当するセッションを再起動します。

## NICE DCV セッションの表示

Windows NICE DCV サーバーの管理者または Linux NICE DCV サーバーのルートユーザーは、サーバーで実行されているすべてのアクティブなセッションを表示できます。NICE DCV ユーザーは、自分が作成したセッションのみ表示できます。

### トピック

- [すべてのアクティブなセッションを一覧表示する](#)
- [特定のセッションに関する情報を表示する](#)

### すべてのアクティブなセッションを一覧表示する

Windows または Linux の NICE DCV サーバーでアクティブなコンソールまたは仮想セッションを一覧表示するには、`dcv list-sessions` コマンドを使用します。

### トピック

- [Syntax](#)
- [出力](#)

### Syntax

```
dcv list-sessions
```

### 出力

このコマンドから、次の形式でアクティブなセッションのリストが返されます。

```
Session: session-id (owner:session-owner type:virtual/console name:'my session')
```

## 特定のセッションに関する情報を表示する

セッションに関する情報を表示するには、`dcv describe-session` コマンドを使用して、一意のセッション ID を指定します。

トピック

- [Syntax](#)
- [出力](#)

### Syntax

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

### 出力

次の出力例では、`display-layout` 要素は、セッションの表示レイアウトが 800x600 画面を 2 つ 使用するように設定されていることを示しています。このうち、2 番目の画面は 1 番目の画面の `x=800` (右側) にオフセットされます。

```
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
```

`--json` (または `-j`) オプションを指定して、コマンドが出力を JSON 形式で返すようにすることもできます。JSON 出力には、セッションに関する追加の詳細が表示されます。

```
$ dcv describe-session session-id --json
```

JSON 出力の例を次に示します。

```
{
  "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
  }
]
}
```

## NICE DCV セッションのスクリーンショットの取得

dcv get-screenshot コマンドを使用すれば、実行中のセッションのデスクトップのスクリーンショットを取得できます。

## 構文

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

## オプション

### --max-width

スクリーンショットの最大幅 (ピクセル) を指定します。幅と高さを指定しない場合は、セッションの表示解像度がスクリーンショットに使用されます。高さのみを指定すると、縦横比が維持されるように幅が自動的に調整されます。

タイプ: 整数

必須: いいえ

### --max-height

スクリーンショットの最大高さ (ピクセル) を指定します。幅と高さを指定しない場合は、セッションの表示解像度がスクリーンショットに使用されます。幅のみを指定すると、縦横比が維持されるように高さが自動的に調整されます。

タイプ: 整数

必須: いいえ

### --format

スクリーンショットのファイル形式。現在サポートされている形式は JPEG と PNG のみです。--format と --output のオプションに競合するファイルタイプを指定した場合は、--format に対して指定した値が優先されます。例えば、--format JPEG と --output myfile.png を指定した場合、NICE DCV で JPEG イメージファイルが作成されます。

型: 文字列

許可される値: JPEG | PNG

必須: いいえ

### --primary

プライマリディスプレイのスクリーンショットのみを取得するかどうかを示します。プライマリディスプレイのスクリーンショットのみを取得するには、--primary を指定します。すべての

ディスプレイのスクリーンショットを取得するには、このオプションを省略します。すべてのディスプレイのスクリーンショットを取得することを選択した場合、すべてのディスプレイが単一のスクリーンショットにまとめられます。

必須: いいえ

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

base64 でエンコードされた JSON 形式で出力を配信するかどうかを示します。JSON 出力を取得するには、`--json` を指定します。それ以外の場合はこれを省略します。

必須: いいえ

## **--output, -o**

スクリーンショットの保存先のパス、ファイル名、およびファイルタイプを指定します。例えば、Windows の場合は `c:\directory\filename.format` を指定し、Linux の場合は `/directory/filename.format` を指定します。形式は `.png` または `.jpeg` でなければなりません。`--format` と `--output` のオプションに競合するファイルタイプを指定した場合は、`--format` に対して指定した値が優先されます。例えば、`--format JPEG` と `--output myfile.png` を指定した場合、NICE DCV で JPEG イメージファイルが作成されます。

タイプ: 文字列

必須: いいえ

## 例

### 例 1

次のコマンド例では、`my-session` という名前のセッションのスクリーンショットを取得します。スクリーンショットにはサーバーの解像度が使用されます。

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

### 例 2

次のコマンド例では、幅 200 ピクセル、高さ 100 ピクセルのスクリーンショットを取得します。`my-session` という名前のセッションのスクリーンショットが取得されます。スクリーンショットが `myscreenshot.png` というファイル名で現在のディレクトリに保存されます。

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

### 例 3

次のコマンド例では、`my-session` という名前のセッションのスクリーンショットを撮影します。スクリーンショットはプライマリディスプレイのもののみです。ファイルが現在のディレクトリに保存され、スクリーンショットに `myscreenshot.png` という名前が付けられます。

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

### 例 4

次のコマンド例では、`my-session` という名前のセッションのスクリーンショットを取得します。このコマンドにより、base64 および JSON 形式でエンコードされたファイルが出力されます。

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

# 方法

## トピック

- [外部認証を使用する](#)
- [アイドル状態のセッションの検索と停止](#)
- [Xサーバーへのリモート X 接続を有効にする](#)
- [NICE DCV ウェブブラウザクライアントを iFrame 内に埋め込む](#)

## 外部認証を使用する

デフォルトでは、NICE DCV クライアント認証は基盤となるオペレーティングシステムに委譲されます。Windows NICE DCV サーバーでは、認証はに委任されます WinLogon。Linux NICE DCV サーバーの場合、認証は PAM に委譲されます。

外部認証サーバーを使用してクライアントを認証するように NICE DCV を設定することができます。その場合は既存の認証システムを使用できます。外部認証の場合は、NICE DCV で既存のログインメカニズムを活用し、認証を外部認証サーバーに委譲します。

外部認証は、DCV サーバーへのアクセス権を持つユーザーを検証して、セッションを作成できるようにします。システム認証とは異なり、独自の外部認証システムをそのように設定しない限り、基盤となる OS に対してユーザーを認証することはありません。

[DCV セッションマネージャー](#)には外部認証システムが組み込まれています。この機能を使用するには、DCV サーバーで [auth-token-verifier](#) パラメータにセッションマネージャーのアドレスを設定する必要があります。

外部認証サーバーを使用するには、以下が必要です。

- ログインメカニズム — これは、ユーザーがログインに使用するフロントエンドメカニズムです。既存の認証情報検証システムを使用してユーザーを検証する機能と、トークンを生成して NICE DCV サーバーに提供できる機能を備えています。詳細については、「[トークンの使用](#)」を参照してください。
- 認証サーバー — これは、ログインメカニズムによって生成されたトークンを認証するサーバーです。このサーバーでは、トークンを含む HTTP (S) POST リクエストを NICE DCV サーバーから受信し、必要な認証を実行して、応答を NICE DCV サーバーに送信できます。認証サーバーの実装の詳細については、「[認証サービス要件](#)」を参照してください。

- NICE DCV サーバー設定 — NICE DCV サーバーは、外部認証サーバーが使用されるように設定する必要があります。詳細については、「[NICE DCV サーバー設定](#)」を参照してください。

## トピック

- [NICE DCV サーバー設定](#)
- [トークンの使用](#)
- [認証サービス要件](#)

## NICE DCV サーバー設定

外部認証サービスが使用されるように NICE DCV サーバーを設定する必要があります。

### Linux NICE DCV server

#### Linux で外部認証サーバーを指定する方法

1. `/etc/dcv/` に移動し、任意のテキストエディタで `dcv.conf` を開きます。
2. `[security]` セクションにある `auth-token-verifier` パラメータを見つけて、既存の値を、外部認証サーバーの URL と通信用ポート (`url:port` 形式) に置き換えます。例えば、を使用している場合は `DcvSimpleExternalAuthenticator`、を指定します `http://127.0.0.1:8444`。

`auth-token-verifier` セクションに `[security]` パラメータが存在しない場合は、次の形式を使用して手動で追加します。

```
[security] auth-token-verifier=url:port
```

3. ファイルを保存して閉じます。

### Windows NICE DCV server

#### Windows で外部認証サーバーを指定する方法

1. Windows レジストリエディタを開きます。
2. `HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/` キーに移動します。
3. セキュリティ `auth-token-verifier` パラメータ で パラメータを見つけます。 [???](#)
4. 次のいずれかを行います。

- 値のデータを使用する場合、外部認証サーバーの URL と通信用ポートを `url:port` 形式で入力します。

#### Example

例えば、を使用している場合は `DcvSimpleExternalAuthenticator`、を指定します `http://127.0.0.1:8444`。

- セキュリティセクションに `auth-token-verifier` パラメータがない場合は、に追加します PowerShell。「[設定パラメーターの変更](#)」を参照してください。

5. Windows レジストリエディタを閉じます。
6. NICE DCV サーバーを [停止](#)して [再起動](#)します。

## トークンの使用

トークンを生成したら、それを NICE DCV サーバーに送信できます。ウェブブラウザクライアントで、次のようにトークンを接続 URL に追加します。

```
https://server_hostname_or_IP:port/?authToken=token#session_id
```

例:

```
https://my-dcv-server.com:8443/?authToken=1234567890abcdef#my-session
```

## 認証サービス要件

カスタム認証サービスは、NICE DCV サーバーの同一ホストで実行することも、別のホストで実行することもできます。認証サービスにより、NICE DCV サーバーからの HTTP (S) POST リクエストがリッスンされます。

NICE DCV サーバーで使用される POST リクエスト形式を以下に示します。

```
POST / HTTP/1.1
Content-Type: application/x-www-form-urlencoded
sessionId=session_id&authenticationToken=token&clientAddress=client_address
```

認証サービスにより、指定されたトークンが有効かどうか判断されます。

トークンの検証後、認証サーバーから NICE DCV サーバーにレスポンスが返されます。レスポンス本文には、認証プロセスの結果に応じて、次のいずれかを含める必要があります。

- 認証が成功した場合、認証サービスから yes の結果とユーザー識別子が返されます。例:

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

- 認証が成功しなかった場合、認証サービスから no の結果が返されます。例:

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

## DcvSimpleExternalAuthenticator

NICE DCV には、DcvSimpleExternalAuthenticator という基準外部認証サーバーが付属しています。DcvSimpleExternalAuthenticator は、独自のカスタム認証サーバーを作成するための出発点を使用できる単一の Python スクリプトです。

DcvSimpleExternalAuthenticator サーバーは HTTP と HTTPS をサポートしており、NICE DCV サーバーがインストールされているのと同じサーバーで実行する必要があります。デフォルトでは、`8444` ポートでリクエストを DcvSimpleExternalAuthenticator リッスンします。必要に応じて、ポートを変更できます。これを行うには、任意のテキストエディタで `/etc/dcv/simpleextauth.conf` を開き、`EXTAUTH_PORT` パラメータを見つけて、既存の値を必要なポート番号に置き換えます。

を使用するには DcvSimpleExternalAuthenticator、`nice-dcv-simple-external-authenticator` パッケージをインストールする必要があります。詳細については、「[NICE DCV サーバーのインストール](#)」を参照してください。

## 簡易外部認証システムの使用

1. 認証ディレクトリに移動します。

```
sudo mkdir -p /var/run/dcvsimpleextauth
```

2. 認証トークンを生成します。

### Example

この例では、`123456` はサンプル認証トークン、`session-123` はサンプルセッション ID、`username` はユーザーです。



```
{
  "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"
}
]
```

コマンド出力で、`num-of-connections` パラメータは、アクティブなクライアント接続の数を示します。値が 0 の場合は、アクティブなクライアント接続がなく、セッションが現在アイドル状態であることを示します。また、`last-disconnection-time` パラメータを使用して、セッションが最後にアクティブなクライアント接続を確立した日時を判別することもできます。

この情報を使用してアイドル状態のセッションを識別するスクリプトまたは cron ジョブを作成できます。その後、[dcv close-session](#) コマンドを使用して、それらの使用を停止できます。

#### Note

セッションを停止すると、そのセッションで実行されているすべてのアプリケーションが閉じます。

## X サーバーへのリモート X 接続を有効にする

デフォルトでは、`Xdcv` では固有のセキュリティリスクにより X 転送の使用が禁止されています。この動作は、新しいバージョンの `Xorg` サーバーから NICE DCV に継承されます。NICE DCV サーバーには、セキュリティリスクを最小限に抑えるために、以下のようなデフォルトの緩和策が実装されています。

- X サーバーではネットワークからの X 接続が禁止されています。X サーバーは、`-nolisten tcp` コマンドラインオプションを使用して起動するように設定されています。ただし、X サーバーへのリモート X 接続を有効にするようにデフォルトの動作を変更できます。この回避策の詳細については、「[X サーバーへのリモート X 接続を有効にする](#)」を参照してください。

- X サーバーは GLX 間接コンテキストを無効にします。DCV-GL との競合のため、現在、GLX 間接コンテキストを有効にするための回避策はありません。

セキュリティ上のリスクとその緩和策の詳細については、「[X.Org セキュリティアドバイザリ](#)」を参照してください。

## X サーバーへのリモート X 接続を有効にする

デフォルトでは、Xdcv は、セキュリティ上のリスクにさらされる可能性を低くするために、`-nolisten tcp` コマンドラインオプションを使用して起動するように設定されています。ただし、X 転送を有効にするようにデフォルトの動作を変更できます。

X 転送を有効にするには

お好みのテキストエディタを使用して `/etc/dcv/dcv.conf` を開きます。ファイルの末尾に以下を追加します。

- IPv4 および IPv6 を介した X 転送を有効にするには

```
[session-management]
virtual-session-xdcv-args="-listen tcp"
```

- IPv4 のみを介した X 転送を有効にするには

```
[session-management]
virtual-session-xdcv-args="-listen tcp -nolisten inet6"
```

### Note

X 転送を有効にしても既存のセッションには影響しません。有効化後に開始された新しいセッションのみに影響します。

X 転送をテストするには

1. NICE DCV セッションを接続します。
2. NICE DCV サーバーによるリッスンが、ポートで、6000 ~ 6063 の範囲で行われていることを確認します。

```
$ netstat -punta | grep 600
```

3. リモートサーバーを NICE DCV サーバーのホストアクセスリストに追加します。

```
$ xhost +remote_server
```

4. NICE DCV セッションの表示番号を取得します。

```
$ dcv describe-session session_name | grep display
```

5. アプリケーションがホストされているリモートサーバーに SSH で接続します。

```
$ ssh user@remote_server
```

6. リモートサーバーから、NICE DCV セッションの X サーバーを参照する表示環境変数をエクスポートします。

```
$ export DISPLAY=dcv_server_ip:display_number
```

7. リモートサーバーから、X 転送機能をテストするアプリケーションを実行します。例:

```
xterm
```

テストアプリケーション (この場合は xterm) が NICE DCV サーバーのデスクトップ環境に表示されます。

## NICE DCV ウェブブラウザクライアントを iFrame 内に埋め込む

クリックジャック攻撃から保護するために、NICE DCV ではデフォルトで、ウェブブラウザクライアントを iFrame 内に埋め込むことができないようになっています。ただし、このデフォルトの動作をオーバーライドすれば、ウェブブラウザクライアントが iFrame 内で実行できるようにすることができます。

クリックジャック攻撃防止対策の詳細については、「[コンテンツセキュリティポリシーのチートシート](#)」を参照してください。

ウェブブラウザを iFrame 内で実行できるようにするには、次の追加の HTTP レスポンスヘッダーがウェブブラウザクライアントに送信されるように NICE DCV サーバーを設定する必要があります。

- web-x-frame-options
- web-extra-http-headers

ウェブブラウザ間で最適な互換性を確保するために、両方のヘッダーを追加することをお勧めします。

**Note**

NICE DCV 接続ゲートウェイ経由で接続する場合は、ゲートウェイ設定内で x フレームオプションを定義する必要があります。これを行うには、ゲートウェイ設定の [\[web-resources\]セクション](#) で local-resources-http-headers パラメータを使用します。

### Windows server

1. Windows レジストリエディタを開き、HKEY\_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/connectivity/ キーに移動します。
2. web-x-frame-options パラメータを開きます。[Value data] (値のデータ) に "ALLOW-FROM https://*server\_hostname*" を入力します。

**Note**

このパラメータが存在しない場合は、新しい文字列パラメータを作成して web-x-frame-options という名前を付けます。

3. web-extra-http-headers パラメータを開きます。[Value data] (値のデータ) に [("Content-Security-Policy", "frame-ancestors https://*server\_hostname*")]

**Note**

このパラメータが存在しない場合は、新しい文字列パラメータを作成して web-extra-http-headers という名前を付けます。

4. Windows レジストリエディタを閉じます。
5. NICE DCV サーバーを [停止](#) して [再起動](#) します。

## Linux server

1. 任意のテキストエディタで `/etc/dcv/dcv.conf` を開きます。
2. `[[connectivity]]` セクションで、次の操作を行います。
  - `web-x-frame-options` に `"ALLOW-FROM https://server_hostname"` と入力します。
  - `web-extra-http-headers` に `[("Content-Security-Policy", "frame-ancestors https://server_hostname")]` と入力します。

例:

```
[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. ファイルを保存して閉じます。
4. NICE DCV サーバーを[停止](#)して[再起動](#)します。

ほとんどのブラウザでは、デフォルトで、マイクアクセスや全画面アクセスなどの一部の機能へのアクセスが禁止されています。これらの機能へのアクセスを許可するには、ウェブページの `iFrame` 要素を変更します。例えば、マイクと全画面表示モードへのアクセスを許可するには、`iFrame` 要素を次のように変更します。

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

# NICE DCV のトラブルシューティング

この章では、NICE DCV で発生する可能性のある問題を識別して解決する方法について説明します。

## トピック

- [ログファイルの使用](#)
- [Linux での仮想セッション作成のトラブルシューティング](#)
- [UID を変更すると Linux セッションが開始されない](#)
- [Windows のカーソルの問題の解決](#)
- [IntelliJ IDEA へのコピーと貼り付けの修正](#)
- [自己署名証明書によるリダイレクトの説明](#)
- [Windows の NVIDIA GPU でのマルチモニター/全画面表示の失敗](#)
- [NICE DCV のパフォーマンスと統計のモニタリング](#)

その他のサポートについては、以下のいずれかのリソースを使用してください。

- NICE DCV オンプレミスのお客様で、追加のサポートが必要な場合は、NICE DCV リセラーにお問い合わせください。
- Amazon EC2 で NICE DCV を使用している場合は、[AWSサポートに](#)サポートチケットを登録できます。
- AWS サポートプランに加入していない場合は、[AWS re:Post](#)に質問を投稿すれば NICE DCV コミュニティのヘルプを受けることができます。

## ログファイルの使用

NICE DCV ログファイルは、NICE DCV サーバーの問題を特定して解決するために使用できます。NICE DCV ログファイルは、NICE DCV サーバーの次の場所にあります。

- Windows サーバー

```
C:\ProgramData\NICE\dcv\log\server.log
```

**Note**

デフォルトでは、ProgramData フォルダは非表示になっている場合があります。ProgramData フォルダが表示されない場合は、非表示の項目を表示するようにファイルブラウザを設定します。または、アドレスバーに「%programdata%」と入力し、Enter キーを押します。

- Linux サーバー

```
/var/log/dcv/server.log
```

NICE DCV サーバーにより、ログファイルの詳細レベルを設定できるようになります。以下の詳細レベルを使用できます。

- error - 最小限の詳細を提供します。エラーのみが含まれています。
- warn - エラーと警告を含みます。
- info - デフォルトの詳細レベルです。エラー、警告、情報メッセージが含まれています。
- debug - 最も詳細な情報を提供します。問題のデバッグに役立つ詳細情報を提供します。

## Windows でのログファイルの詳細レベル変更

ログファイルの詳細レベルを設定するには、Windows レジストリエディタを使用して level パラメータを設定する必要があります。

Windows でログファイルの詳細レベルを変更するには

1. Windows レジストリエディタを開きます。
2. [HKEY\_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/log/] キーに移動します。
3. [level] パラメータをダブルクリックして開きます。[値のデータ] に、必要な詳細レベルに応じて「error」、「warn」、「info」、または「debug」のいずれかを入力します。
4. [OK] を選択して Windows レジストリエディタを閉じます。

## Linux でのログファイルの詳細レベル変更

ログファイルの詳細レベルを設定するには、level ファイルの dcv.conf パラメータを設定する必要があります。

Linux でログファイルの詳細レベルを変更するには

1. /etc/dcv/ に移動し、任意のテキストエディタで dcv.conf を開きます。
2. level セクションで [log] パラメータを見つけ、既存の詳細レベルを error、warn、info または debug のいずれかで置き換えます。

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

3. ファイルを保存して閉じます。

## Linux での仮想セッション作成のトラブルシューティング

トピック

- [Linux での仮想セッション作成失敗の調査](#)
- [Linux でのフェイルセーフ仮想セッションの作成](#)

仮想セッションに接続した結果 No session available または The sessionId *session* is not available のエラーが発生した場合、仮想セッションの作成に失敗し、終了したことが原因と考えられます。

dcv list-sessions コマンドを使用して、セッションが存在するかどうかを確認できます。実行中のセッションの検査の詳細については [the section called “セッションの表示”](#) を参照してください。セッションがリストにない場合は、セッションが失敗した可能性があります。

## Linux での仮想セッション作成失敗の調査

Linux では以下のコマンドで仮想セッションが[作成](#)されます。

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

このコマンドは、セッションの作成に失敗した場合にのみエラーを返します。ただし、セッションが最初は正常に作成されたものの、ユーザーが接続する前に終了する場合があります。例えば `dcv list-sessions` または `dcv describe-session session` コマンドで既存のセッションを確認してもセッションがリストアップされていない場合に、これに気づく場合があります。

ほとんどの場合、デスクトップセッションが作成されてすぐに失敗することが原因です。例えば、`init` スクリプトによって起動されたアプリケーションの 1 つがクラッシュまたは失敗した場合や、必要なツールの 1 つが不足している場合などです。

セッションの作成に失敗した場合は以下を確認してください。

- 新しいセッションプロセスを作成する `dcv` コンポーネントに関連するログを含む `/var/log/dcv/sessionlauncher.log` ファイルを確認します。
- `dcv init` スクリプトに関連するログを含む `/var/log/dcv/dcv-session.user.session.log` ファイルを確認します。
- セッション所有者に対応するホームディレクトリの `$HOME/.xsession-errors` ファイルを確認します。このファイルには、システムの X セッションの `init` スクリプトによって生成されたログが含まれ、通常はデスクトップセッションマネージャーまたはスクリプトによって呼び出された他のアプリケーションによって生成されたログが含まれます。
- 失敗したシステムおよびコンポーネントの詳細を知るにはシステムログを確認します。まず、`dmesg` (プロセスが失敗した場合など) と `journalctl -xe` の出力を確認してください。
- [フェイルセーフセッションでテストして](#)、問題が使用中のセッションマネージャーに依存していないことを確認します。

特定のユーザーにのみ問題が発生する場合は、以下の方法も試行してください。

- ユーザー設定 (特にユーザー設定が削除されたり名前が変更されたりするとどうなるか) を確認します。

デスクトップ環境とバージョンによっては、設定ディレクトリがユーザーディレクトリ内の `.gnome`、`.kde`、または `.config` になっている場合があります。

- ユーザー `PATH` や環境に影響する特定のユーザー設定がないかを確認します。多くの場合、特定のユーザーがセッション開始に失敗するのは、`anaconda` などのフレームワークが標準のネイティブコマンドを上書きし、セッション初期化中の `dbus` の接続が失敗するためです。
- 権限の問題がないかを確認します。ローカルの `~/.dbus` または `~/.Xauthority` に誤った権限を設定した場合 (ユーザーではなく `root` によって所有されている場合など)、デスクトップセッションが直ちに終了する可能性があります。

## Linux でのフェイルセーフ仮想セッションの作成

セッション作成の失敗がデスクトップ環境の起動に関係しているかどうかを確認する一般的な方法は、最小限のセッションを作成することです。このセッションを「フェイルセーフ」セッションと呼びます。フェイルセーフセッションの作成が正常に機能すれば、デフォルトシステムのデスクトップ環境が起動できないために通常のセッションが失敗したと推測できます。逆に、フェイルセーフセッションにも失敗した場合、問題は NICE DCV サーバーの設定に関連している可能性が高いと考えられます。

通常フェイルセーフセッションは、シンプルなウィンドウマネージャーとターミナルのみを含むデスクトップセッションで構成されます。これにより、ユーザーは、使用中の特定のセッション環境 (通常は gnome や KDE) に関連するセッション作成の問題があるかどうかを確認できます。

フェイルセーフセッションを作成するには、以下のような内容を含むユーザー用の init スクリプトを作成する必要があります。

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

これにより metacity ウィンドウマネージャーが起動し、xterm ターミナルが開始されます。xterm プロセスが終了すると同時にセッションも終了します。

システム上で使用可能な場合は、任意の別のセッションマネージャーまたはターミナルを使用することもできます。

### Note

スクリプトがすぐに終了しないことを確認する必要があります。そのためには、スクリプトが終了するまでに、即時終了しないプログラムを起動しておく必要があります。最後のコマンドが終了すると (この例では xterm)、init セッションも終了します。同時に、Windows マネージャーの後に別のツールを起動する場合は、そのツールがバックグラウンドで実行されていることを (この例では & を追加して) 確認し、次のコマンドが確実に呼び出されるようにする必要があります。

次に、init スクリプトが実行可能であることを確認する必要があります。

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

ユーザーシェルから指定した `init` スクリプトを使用してセッションを作成するには、以下のコマンドを実行します。ここで、`init.sh` は以前に作成したスクリプトです。

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

スーパーユーザーとして別のユーザーのセッションを作成するには、代わりに以下のコマンドを実行します。

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

最後に、たとえば `dcvgltest` (`nice-dcv-glttest` パッケージがインストールされている場合のみ) や `glxgears` のようなテストアプリケーションを起動して、OpenGL などのアプリケーションが正しく動作していることを確認できます。

## UID を変更すると Linux セッションが開始されない

Linux ホストでは、ユーザーのユーザー ID (UID) を変更したり、ユーザーの UID を変更する別の Active Directory 構成を使用したりすると、ホストで NICE DCV セッションを開始できない場合があります。

この問題は、新しい UID で実行される DCV セッションのプロセスが、以前の UID を保持しているファイルやフォルダにアクセスする権限がないことが原因です。特に、次のことに注意してください。

- NICE DCV ログディレクトリ内の [ログファイル](#)
- ユーザーのホームフォルダ

この問題は、コンソールセッションと仮想セッションの両方に影響します。

この問題を解決するには、ユーザーのホームフォルダとそこに含まれるファイルに正しい UID が設定されていることを確認し、以前の UID を持つ古い [NICE DCV ログファイル](#) を削除します。

## Windows のカーソルの問題の解決

NICE DCV サーバーが Windows Server 2012 または Windows 10 以降で実行されている場合、マウスカーソルは常に矢印として表示されます。これは、テキスト入力フィールドまたはシングルクリックのナビゲーション項目で一時停止する場合にも発生します。これは、サーバーに接続されている物

理マウスがない場合、またはデバイスマネージャーにマウスデバイスが表示されていない場合に発生します。

問題を解決するには

1. コントロールパネルを開き、[簡単アクセスセンター] を選択します。
2. [マウスを使いやすくする] を選択します。
3. [マウスキーを有効にする] を選択します。
4. [適用]、[OK] の順に選択します。

## IntelliJ IDEA へのコピーと貼り付けの修正

macOS NICE DCV クライアントからテキストをコピーして IntelliJ IDEA に貼り付けることはできません。IntelliJ は、NICE DCV でデフォルトにより使用されるクロスプラットフォーム形式に対応していません。NICE DCV でクロスプラットフォームテキストを無効にして IntelliJ にテキストを貼り付けるには、NICE DCV サーバの `disabled-targets` フィールドに変更を加えます。

この変更により、NICE DCV ウェブクライアントでのコピーと貼り付けが機能しなくなります。この変更を行う前に、IntelliJ IDEA のコピーと貼り付けが NICE DCV クライアントでのみ動作することを確認してください。

IntelliJ IDEA にテキストを貼り付けるためにサーバーを設定するには

1. `/etc/dcv/` に移動し、任意のテキストエディタで `dcv.conf` を開きます。
2. `disabled-targets` パラメータを `[clipboard]` セクションで見つけます。`disabled-targets` セクションも `[clipboard]` セクションも存在しない場合は、それらを手動で追加します。
3. `disabled-targets` の値を定義するには、次の内容を追加します。

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

4. ファイルを保存して閉じます。
5. NICE DCV セッションを [停止](#)して [再起動](#)します。

## 自己署名証明書によるリダイレクトの説明

ウェブベースのポータルまたはアプリケーションから NICE DCV セッションにリダイレクトする場合、証明書が以前に信頼されていなかった場合は、自己署名証明書によってセッションに対するブラウザの信頼が失われる可能性があります。このような状況が発生する例は以下のとおりです。

1. ユーザーが企業のポータルサイトに接続し、そこからアプリケーションが読み込まれます。
2. アプリケーションは、自己署名証明書を使用して NICE DCV サーバーとの直接かつ安全な接続を確立しようとしています。
3. 証明書が自己署名であるため、ブラウザは安全な接続を拒否します。
4. 接続が確立されていないため、ユーザーにはリモートサーバーが表示されません。

信頼の問題はステップ 3 で発生しています。ユーザーが自己署名証明書を使用して Web サイトに接続すると (たとえば、<https://example.com> を開く場合)、ブラウザは証明書を信頼するかどうかを尋ねます。ただし、HTTP または HTTPS 経由で提供される Web アプリケーション/ページが DCV サーバーへの安全な WebSocket 接続を確立しようとした場合で、証明書が自己署名である場合は、ブラウザはその証明書が以前に信頼されていたかどうかを確認します。その証明書が以前に信頼されていなかった場合は、証明書を信頼するかどうかをユーザーに確認することなく接続を拒否します。

この場合に考えられる解決策:

- 企業が自社のマシンにカスタムドメインを使用している場合は、DCV サーバーマシンの有効な証明書をを用意します。証明書については、DCV にエンタープライズ証明書を配布できます。

### Example

ユーザー ---[有効な証明書]---> DCV サーバーインスタンス

- DCV サーバーフリートをプロキシ/ゲートウェイで保護します。この場合に限り、プロキシ/ゲートウェイに有効な証明書が必要で、DCV サーバーインスタンスは自己署名証明書を保持できます。このオプションでは [DCV 接続ゲートウェイ](#)、ALB/NLB、または別のプロキシソリューションを使用できます。

### Example

ユーザー/Cx---[ここで有効な証明書が必要です]---> プロキシ/ゲートウェイ---[自己署名証明書]---> DCV サーバーインスタンス

- [SDK](#) 経由で接続を開始する前に、ユーザーに自己署名証明書を信頼してもらいます。これは、この URL を別のタブ/ウィンドウ/ポップアップで開くだけで可能です。https://example.com/version

#### Note

/version エンドポイントは、HTTPS 接続で DCV サーバーバージョン用の簡単なウェブページを返信します。

同じ自己署名証明書を後で実際の DCV サーバー接続で使用できます。

## Windows の NVIDIA GPU でのマルチモニター/全画面表示の失敗

Windows サーバーホストに NVIDIA GPU が搭載されている場合、DCV 全画面表示/マルチモニター機能が動作しないことがあります。この場合、ディスプレイが全画面表示モードに入るのを拒否したり、複数のリモートモニターによるディスプレイレイアウトをサーバーが構成できなくなったりします。

この問題の原因は、NVIDIA ドライバーとの統合の失敗です。

サーバーホストで C:\ProgramData\NICE\dcv\log\ を確認することで問題を特定できます。以下のエラーが報告されます。

```
WARN display - Cannot change display layout
```

このエラーが複数回 (20 ~ 30 回) 表示されると、次に以下が表示されます。

```
EDID not set on output x gpu x after attempt x INFO DLMNVAPI:display -  
Unable to set EDID on output x, gpu x: NVAPI_ERROR (-1)
```

問題が再現されると、ホストに異常が発生します。サーバーはマルチモニターレイアウトを常に構成することができなくなり、問題を永続的に修正する有効な方法はありません (わずかな一時的緩和策のみ)。

この問題の原因は、マルチモニターの使用中にサーバー OS が再起動されることです。つまり、サーバーホストに仮想モニターが存在するときにホストがシャットダウンされる場合です。そのため、この問題を回避するには、サーバーをシャットダウンする前にサーバー側のすべてのモニターを削除する必要があります。これを確実に実行するには、以下のコマンド (管理者権限で実行) を使用できます。

```
C:\Program Files\NICE\DCV\Server\bin\dcvnvedid.exe --remove
```

考えられる緩和策は、Nvidia ドライバーを再インストールまたは更新し、ホストを再起動することです。

## NICE DCV のパフォーマンスと統計のモニタリング

NICE DCV 2023.1 以降のサーバーでは、Windows パフォーマンスカウンターを使用してプロトコルパフォーマンスのさまざまな側面を監視し、NICE DCV のセッションと接続に関する統計情報を収集できます。

### パフォーマンスカウンターを収集するツール

- [パフォーマンスモニター \(PerfMon\)](#): パフォーマンスデータをリアルタイムで、またはログファイルから視覚化できる Windows ネイティブのツール。
- [LogMan](#): 指定した条件に基づいてログ記録を開始および停止できるコマンドラインツール。
- [TypePerf](#): パフォーマンスデータをコマンドウィンドウまたはログファイルに書き込むコマンドラインツール。
- [PowerShell](#): パフォーマンスデータの収集と操作に使用できる Windows スクリプト言語。
- サードパーティーツール: これらのカウンターを収集して詳細な分析情報を提供できる複数のサードパーティーの監視ソリューションがあります。

DCV パフォーマンスカウンターは 5 つのカウンターセットにグループ化されています。

## カウンターセット

### NICE DCV サーバー

このカウンターセットには、ホストの DCV サーバーサービスに関するグローバル統計が含まれています。また、他のカウンターセットでも利用できる多数のカウンターの集約されたバリエーションも含まれており、サーバーのライフタイム全体にわたって集約された情報に静的パスでアクセスする方法となります (このカウンターセット内のカウンターを読み取るためにセッション識別子や接続識別子を取得する必要はありません)。

**Note**

他のカウンターセット (例: 「\DCV Server Connections(\_Total)\Sent Bytes」) のいずれかから集約されたインスタンスは、アクティブなすべての接続の合計を返します。一方グローバルカウンターはサーバーの起動以降に累積されたもので、終了した接続も含まれます。

カウンター名	説明	単位	メモ
アクティブなセッション	ホスト上のアクティブなセッション数	Count (カウント)	
Total Sessions	終了したセッションを含む、ホストで作成されたセッション数	Count (カウント)	
Active Connections	サーバーへのアクティブな接続の数	Count (カウント)	
Total Connections	サーバーへの接続数 (アクティブなクライアント、再接続されたクライアント、切断されたクライアントを含む)	Count (カウント)	
Idle Disconnections	操作がないために切断された接続の数	Count (カウント)	
Receive Rate bits/sec	サーバーがデータを受信する速度 (ビット/秒)	ビット/秒	
Received Bytes	サービスの開始以降に受信した総バイト数	バイト	

カウンター名	説明	単位	メモ
Send Rate bits/sec	サーバーがデータを送信する速度 (ビット/秒)	ビット/秒	
Sent Bytes	サービスの開始以降に送信された総バイト数	バイト	
HTTP Download Rate bits/sec	送信 HTTP トラフィックの帯域幅 (ビット/秒)	ビット/秒	ファイルストレージのクライアントからサーバーへのトラフィックは受信速度にカウントされます。
HTTP Downloaded Bytes	サービスの開始以降に HTTP 経由で送信された総バイト数	バイト	ファイルストレージのクライアントからサーバーへのトラフィックは受信バイト数にカウントされます。
Round-Trip Time ms	サーバーとクライアント間の平均往復遅延 (ミリ秒)	ミリ秒	5 秒に 1 回測定および更新されます。
Minimum Round-Trip Time ms	サーバーの起動以降に検出された最小往復遅延 (ミリ秒)	ミリ秒	5 秒に 1 回更新されます。

## DCV サーバードロセス

このカウンターセットには、個々の NICE DCV プロセスに関する情報が含まれています。

agent\_type can be one of: session\_agent, system\_agent, user\_agent

カウンターは 1 秒に 1 回更新されます。

カウンター名	説明	単位	メモ
% Processor Time	プロセスが使用するプロセッサ時間の割合	割合 (%)	パーセンテージは 1 つの CPU コアに対する相対値です (つまり、100% はプロセスが 1 つのスレッドを占有していることを意味します)。  \Process(NAME)\% Processor Time と同じです。
Physical Memory Bytes	プロセスが使用する現在の物理メモリの量 (バイト)	バイト	\Process(NAME)\Working Set と同じです。
Virtual Memory Bytes	プロセスの仮想アドレス空間の現在のサイズ (バイト)	バイト	
Process Identifier	数字によるプロセス識別子 (PID)	-	

## NICE DCV サーバーセッション

このセットのカウンターは、1 つのセッションに関する情報を提供します。ユーザーが接続しているかどうかにかかわらず、作成されたセッションごとにカウンターセットのインスタンスが 1 つ作成されます。

管理者がセッションを閉じると、対応するインスタンスは削除されます。管理者が同じ名前のセッションを再作成すると、すべてのカウンターがゼロから再開されます。

カウンター名	説明	単位
Session Duration sec	セッションが開かれていた時間の合計 (秒)	[Seconds] (秒)
Total Pixels	表示領域のピクセル数 (セッション内のすべてのディスプレイの合計ピクセル数。)	ピクセル
Display Count	セッション内のディスプレイ数	Count (カウント)

以下のカウンターは NICE DCV サーバーカウンターセットのものと同じですが、説明に若干の違いがあります。

カウンター名	説明
Active Connections	セッションインスタンスへのアクティブな接続の数
Total Connections	アクティブなクライアント、再接続されたクライアント、切断されたクライアントを含むセッションインスタンスへの接続数
Idle Disconnections	操作がないために切断されたセッションインスタンスへの接続の数
Ungraceful Disconnections	エラーにより切断されたセッションインスタンスへの接続数
Receive Rate bits/sec	セッション内でデータを受信する速度 (ビット/秒)
Received Bytes	セッションの開始以降に受信した総バイト数
Send Rate bits/sec	セッション内でデータを送信する速度 (ビット/秒)

カウンター名	説明
Sent Bytes	セッションの開始以降に送信された総バイト数
HTTP Download Rate bits/sec	セッション内の送信 HTTP データの帯域幅 (ビット/秒)
HTTP Downloaded Bytes	セッション内で HTTP 経由で送信された総バイト数
Round-Trip Time ms	セッション内のサーバーとクライアント間の平均往復遅延 (ミリ秒)
Minimum Round-Trip Time ms	セッションの確立後に検出された最小往復遅延 (ミリ秒)

## NICE DCV サーバー接続

このセットのカウンターは、1つのクライアント接続に関する情報を提供します。カウンターセットインスタンスは、クライアントがサーバーに接続すると作成され、クライアントが切断されると削除されます。connection\_id は数字で、1つのサーバーセッション内でのみ一意です。

カウンター名	説明	単位
Connection Duration sec	接続が開かれていた合計時間 (秒)。	[Seconds] (秒)

以下のカウンターは「DCV サーバー」カウンターセットのものと同じですが、説明に若干の違いがあります。

カウンター名	説明
Receive Rate bits/sec	接続内でデータを受信する速度 (ビット/秒)
Received Bytes	接続確立後に受信した総バイト数
Send Rate bits/sec	接続内でデータを送信する速度 (ビット/秒)

カウンター名	説明
Sent Bytes	接続確立後に送信した総バイト数
HTTP Download Rate bits/sec	接続内の送信 HTTP データの帯域幅 (ビット/秒)
HTTP Downloaded Bytes	接続確立後に HTTP 経由で送信された総バイト数
Round-Trip Time ms	接続の平均往復遅延 (ミリ秒)
Minimum Round-Trip Time ms	接続の確率後に検出された最小往復遅延 (ミリ秒)

## NICE DCV サーバーチャネル

このセットのカウンターは、クライアント接続内の個々のチャネルに関する情報を提供します。拡張機能用のチャネルが含まれる場合もあります。

チャネル名は以下のとおりです。

- dcv::main
- dcv::display
- dcv::input
- dcv::audio
- dcv::filestorage
- dcv::clipboard

受信ファイルストレージトラフィックは dcv::filestorage チャネルに帰属されます。

送信ファイルストレージトラフィックは DCV サーバー接続の HTTP ダウンロードカウンターに含まれます。

### Note

このセットのカウンターは DCV サーバー接続のカウンターのサブセットです。

カウンター名	説明
Receive Rate bits/sec	チャンネル経由でデータを受信する速度 (ビット/秒)
Received Bytes	チャンネル経由で受信した総バイト数
Send Rate bits/sec	チャンネル経由でデータが送信される速度 (ビット/秒)
Sent Bytes	チャンネル経由で送信された総バイト数

## NICE DCV サーバーイメージング

このセットのカウンターは、画面の取得、エンコーディング、配信を行うサブシステムに関する情報を提供します。

このセットのカウンターは以下の 2 つのグループに分かれています。

- 最初のグループのカウンターでは、NICE DCV はセッションごとに 1 つの値を収集し、`$session_name` インスタンスに公開します。
- 2 番目のグループでは、NICE DCV は各セッションのエンコーダーごとに 1 つの値を収集します。以下の 3 つのアクティブエンコーダーがあります。
  - 1 つのフルフレームエンコーダー
  - 1 つのタイルベースのエンコーダー
  - 1 つのロスレスエンコーダー

これらのカウンターは `$session_name:$encoder_name` インスタンスで公開されます。

カウンター名	説明	単位	インスタンス
Grabbed Frames/sec	キャプチャされたフレームレート (フレーム数/秒)	カウント/秒	セッション

カウンター名	説明	単位	インスタンス
Grabbed Frames	セッション開始以降にキャプチャされたフレームの総数	Count (カウント)	セッション
Sent Frames/sec	1 秒あたりに接続しているクライアントに送信されたスクリーンフレーム数	カウント/秒	セッション
Dropped Frames/sec	1 秒あたりに接続しているクライアントに送信されなかったスクリーンフレーム数	カウント/秒	セッション
Display Latency ms	フレームのキャプチャから表示までの平均時間 (ミリ秒)	ミリ秒	セッション
Available Bandwidth bits/sec	接続で使用可能な推定帯域幅 (ビット/秒)	ビット/秒	セッション
Encoded Frames/sec	1 秒あたりにエンコードされたスクリーンフレーム数	カウント/秒	セッション:エンコーダー
Encoding Time ms	1 スクリーンフレームのエンコードにかかった平均時間 (ミリ秒)	ミリ秒	セッション:エンコーダー
Encoding Time per Megapixel ms	100 万ピクセルのエンコードに使用された平均時間 (ミリ秒)	ミリ秒	セッション:エンコーダー
Frame Quality %	平均フレーム圧縮品質 (パーセンテージ)	割合 (%)	セッション:エンコーダー

カウンター名	説明	単位	インスタンス
Frame Compression Ratio %	フレームサイズ (バイト) と圧縮済みフレームサイズの比率である平均フレーム圧縮率	割合 (%)	セッション:エンコーダー

# NICE DCV サーバパラメータリファレンス

以下の表に、NICE DCV サーバをカスタマイズするために設定可能なパラメータを示します。

## Note

各テーブルの再ロードコンテキスト列には、パラメータがいつ再ロードされるかが示されます。可能なコンテキストには、以下が含まれます。

- `server` — このパラメータはサーバの起動時に一度ロードされます。パラメータ値が更新されると、サーバの再起動時に新しい値がロードされます。
- `session` — このパラメータはセッションの作成時にロードされます。パラメータ値が更新されると、新しい値が後続のセッション用にロードされます。
- `connection` — このパラメータは新しいクライアント接続が確立されたときにロードされます。パラメータ値が更新されると、新しい値が後続のクライアント接続に使用されません。
- `custom` — このパラメータがロードされる条件は、このパラメータに固有です。詳細については、パラメータの説明を参照してください。

## トピック

- [audio パラメータ](#)
- [clipboard パラメータ](#)
- [connectivity パラメータ](#)
- [display パラメータ](#)
- [display/linux パラメータ](#)
- [extensions パラメータ](#)
- [input パラメータ](#)
- [license パラメータ](#)
- [log パラメータ](#)
- [printer パラメータ](#)
- [redirection パラメータ](#)
- [security パラメータ](#)
- [session-management パラメータ](#)

- [session-management/automatic-console-session](#) パラメータ
- [session-management/defaults](#) パラメータ
- [smartcard](#) パラメータ
- [webauthn](#) パラメータ
- [webcam](#) パラメータ
- [windows](#) パラメータ
- [設定パラメータの変更](#)

## audio パラメータ

以下の表に、Linux NICE DCV サーバーの `/etc/dcv/dcv.conf` ファイルの `[audio]` セクションの設定パラメータと、Windows NICE DCV サーバーの audio レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
avsync-support	string	セッ ション	'auto'	クライアントによるオーディオ/ビデオ同期の有効化を許可するかどうかを決める — 接続されたクライアントによるオーディオ/ビデオ同期の有効化を許可します。有効値は 'enabled'、'disabled'、'auto' (デフォルトは 'auto') です。'auto' を指定すると、オーディオ/ビデオの同期はコンソールセッションでのみ有効になり、高速ビデオ圧縮が使用可能な場合のみ有効になります。 — バージョン <a href="#">2021.1-10557</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
source-channels	integer - DWORD (32 ビット)	セッ ション	2	Linux のスピーカーデバイスのチャンネル数 — Linux のスピーカーデバイスのチャンネル数を設定します。この値は、デバイスによってサポートされるチャンネル数以下とします。許容される値は、2 (ステレオ)、4 (4.0 チャンネルステレオ)、6 (5.1 サラウンド)、8 (7.1 サラウンド) です。デフォルト値は 2 (ステレオ) です。 — バージョン <a href="#">2020.0-8428</a> から使用可能。

## clipboard パラメータ

以下の表に、Linux NICE DCV サーバーの /etc/dcv/dcv.conf ファイルの [clipboard] セクションの設定パラメータと、Windows NICE DCV サーバーの clipboard レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
enabled	true ま たは false - DWORD	セッ ション	Linux: true - Windows: 1	クリップボード機能を有効にするかどうか - クリップボード機能を有効にするかどうかを指定します。クリップ

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
	(32 ビット)			ボード機能が無効になっている場合、ユーザーはクリップボードのリモート化を使用できなくなります。クリップボードのモニタリングも無効になります。—バージョン <a href="#">2020.0-8428</a> から使用可能。
max-image-area	integer - DWORD (32 ビット)	セッ ション	-1	クリップボードのイメージの最大領域 — サーバーとクライアントの間で転送できるクリップボードのイメージの最大領域 (ピクセル数) を指定します。この値を指定しないか、-1 に設定されている場合、制限は適用されません。—バージョン <a href="#">2017.0-4334</a> から使用可能。
max-payload-size	integer - DWORD (32 ビット)	セッ ション	20971520	クリップボードのデータの最大サイズ — サーバーとクライアントの間で転送できるクリップボードのデータの最大サイズ (バイト) を指定します。サポートされる最大値は 20 MB です。この値を指定しない場合、最大値が適用されます。—バージョン <a href="#">2017.0-4334</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
max-text-len	integer - DWORD (32 ビット)	セッ ション	-1	クリップボードのテキストの最大文字数 — サーバーからクライアントに転送できるクリップボードテキストの最大文字数を指定します。それよりも長い文字は切り詰められます。この値を指定しないか、-1 に設定されている場合、制限は適用されません。 — バージョン <a href="#">2017.0-4334</a> から使用可能。
primary-selection-copy	true ま たは false - DWORD (32 ビット)	セッ ション	Linux: false - Windows: 0	Linux からのプライマリ選択コピーを有効にする — Linux デスクトップは複数のクリップボード (汎用クリップボードとプライマリ選択) をサポートします。コンテンツを選択すると、プライマリ選択が更新またはコピーされます。その後、マウスの中央ボタンを使用するか、Shift+Insert キーの組み合わせを使用して貼り付けられます。有効にすると、プライマリ選択がモニタリングされ、更新がクライアントに伝播されます。 — バージョン <a href="#">2019.0-7318</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
primary-selection-paste	true ま たは false - DWORD (32 ビット)	セッ ション	Linux: false - Windows: 0	Linux でのプライマリ選択の貼り付けの有効化 — Linux デスクトップでは、複数のクリップボード (汎用クリップボードとプライマリ選択) をサポートします。コンテンツを選択すると、プライマリ選択が更新またはコピーされます。その後、マウスの中央ボタンまたは Shift+Insert キーの組み合わせを使用して貼り付けられます。有効にすると、クライアントのクリップボードのコンテンツもプライマリ選択に挿入されます。 — バージョン <a href="#">2019.0-7318</a> から使用可能。
update-timeout	integer - DWORD (32 ビット)	セッ ション	200	イベント通知タイムアウトの更新 — クライアントに通知を送信するための最終更新イベントからの待機時間をミリ秒単位で指定します。デフォルト値は 200 ミリ秒です。 — バージョン <a href="#">2020.1-8942</a> から使用可能。

## connectivity パラメータ

以下の表に、Linux NICE DCV サーバーの `/etc/dcv/dcv.conf` ファイルの `[connectivity]` セクションの設定パラメータと、Windows NICE DCV サーバーの `connectivity` レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再ロ ード	デフォルト値	説明
<code>disconnect-on-lock</code>	true または false - DWORD (32 ビット)	カスタム	Linux: false - Windows: 0	OS のセッションロック時にクライアントを切断するかどうか — 有効化すると、リモート OS セッションがロックされたときにクライアントを強制的に切断します。有効化しない場合、クライアントはリモートセッションのストリーミングを続行します。現在はコンソールセッションでのみサポートされています。このパラメータ値は、リモート OS セッションがロックされるたびに読み込まれます。 — バージョン 2023.1-16220 から使用可能。 <a href="#">???</a>
<code>disconnect-on-logout</code>	true または false - DWORD (32 ビット)	カスタム	Linux: false - Windows: 0	OS のユーザーログアウト時にクライアントを切断するかどうか — 有効化すると、OS ユーザーがログアウトしたとき (つまり OS セッションが終了したとき) にクライアントのを強制的に切断します。有効化しない場合、クライアント

[パラメータ]	タイプ - Windowsレジストリの種類	コンテキストの再ロード	デフォルト値	説明
				はリモートセッションのストリーミングを続行します。現在はコンソールセッションでのみサポートされています。このパラメータ値は、リモート OS ユーザーがログアウトするたびに読み込まれます。— バージョン 2023.1-16220 から使用可能。 <a href="#">???</a>
enable-quick-frontend	true または false - DWORD (32 ビット)	server	Linux: false - Windows: 0	QUIC フロントエンドを有効にするかどうか - QUIC フロントエンドを有効にするかどうかを指定します。— バージョン <a href="#">2020.2-9508</a> から使用可能。
idle-timeout	integer - DWORD (32 ビット)	custom	60	アイドルタイムアウト — アイドルクライアントを切断するまでの待機時間 (分) を指定します。0 に指定すると、アイドルクライアントは切断されません。このパラメータ値は毎秒読み込まれます。— バージョン 2017.0-4100 <a href="#">から使用可能</a> 。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
idle-timeout-warni ng	integer - DWORD (32 ビット)	custom	350	アイドルタイムアウトの警告 — アイドル状態のクライアント に対してアイドルタイムア ウトの切断を警告するまでの アイドルタイムアウト関連待 機時間 (秒) を指定します。ア イドル状態のクライアントに 警告しないようにするには、0 を指定します。— バージョン <a href="#">2017.4-6898</a> から使用可能。

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
quic-listen-endpoints	string	server	['0.0.0.0', ':::']	<p>DCV により着信 QUIC 接続がリッスンされるエンドポイントを指定する — DCV により着信 QUIC 接続がリッスンされるエンドポイントのリストを指定します。エンドポイントは、ローカルでバインド可能な IPv4 アドレスのリスト (潜在的なアドレスをすべてワイルドカード化するには '0.0.0.0') か、コロン (':') で区切られた任意のポートを含むバインド可能な IPv6 アドレス (潜在的なすべてのアドレスをワイルドカード化するには ':::') にできます。たとえば '1.2.3.4:5678' とした場合、'1.2.3.4' アドレスに関連付けられたインターフェイスの着信接続がポート 5678 でリッスンされます。ポートが指定されていない場合は、'quic-port' の設定がデフォルトとして使用されます。IPv6 アドレスでポートを指定するには、アドレスを角括弧で囲みます (例: ':::1]:8443')。明示的なインターフェイスを含む IPv6 アドレスもサポートされています (例: ':::%eth1]:8443')。 —</p>

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
				バージョン <a href="#">2022.0-11954</a> から使用可能。
quic-port	integer - DWORD (32 ビット)	server	8443	QUIC フロントエンドの UDP ポート - クライアント接続のために DCV サーバーによりリスンされる UDP ポートを指定します。ポート番号は 1024 ~ 65535 の間である必要があります。この設定の適用方法の詳細については、quic-listen-endpoints 「」設定を参照してください。— バージョン <a href="#">2020.2-9508</a> から使用可能。
web-extra-http-headers	string	server	[]	HTTP/HTTPS ヘッダーにさらに追加ヘッダーの配列を設定 - これを使用してヘッダーを追加します。配列には [('header_name','header_content')] のような組み合わせで入力する必要があります。複数のヘッダーを追加できます。— バージョン <a href="#">2017.2-6182</a> から使用可能。

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
web-listen-endpoints	string	server	['0.0.0.0', ':::]	<p>DCV により着信ウェブ接続がリッスンされるエンドポイントを指定する — DCV により着信ウェブ接続がリッスンされるエンドポイントのリストを指定します。エンドポイントは、ローカルでバインド可能な IPv4 アドレスのリスト (潜在的なアドレスをすべてワイルドカード化するには '0.0.0.0') か、コロン (':') で区切られた任意のポートを含むバインド可能な IPv6 アドレス (潜在的なすべてのアドレスをワイルドカード化するには ':::') にできます。たとえば '1.2.3.4:5678' とした場合、'1.2.3.4' アドレスに関連付けられたインターフェイスの着信接続がポート 5678 でリッスンされます。ポートが指定されていない場合は、'web-port' の設定がデフォルトとして使用されます。IPv6 アドレスでポートを指定するには、アドレスを角括弧で囲みます (例: '[::1]:8443')。明示的なインターフェイスを含む IPv6 アドレスもサポートされています (例: '[:::%eth1]:8443')。 — バー</p>

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
				ジョン <a href="#">2022.0-11954</a> から使用可能。
web-port	integer - DWORD (32 ビット)	server	8443	クライアントの TCP ポート — クライアント接続で DCV サーバーがリッスンする TCP ポートを指定します。ポート番号は 1024 ~ 65535 の間である必要があります。この設定の適用方法の詳細については、web-listen-endpoints 「」設定を参照してください。 — バージョン <a href="#">2017.0-4100</a> から使用可能。
web-root	string	server	"	埋め込みウェブサーバーのドキュメントのルート — 埋め込みウェブサーバーのドキュメントのルート指定します。 — バージョン <a href="#">2017.0-4100</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
web-url-path	string	server	'/'	埋め込みウェブサーバーの URL パス — 埋め込みウェブサーバーの URL パスを指定します (先頭は必ず '/)。たとえば、/test/foo に設定すると、ウェブサーバーは https://host:port/test/foo にアクセスできるようになります。 — バージョン <a href="#">2017.0-4100</a> から使用可能。
web-use-hsts	true ま たは false - DWORD (32 ビット)	server	Linux: true - Windows: 1	HSTS を使用するかどうか — これを有効にして、ブラウザが HTTP 経由で送信される通信を強制的に禁止するようにします。ウェブページ (およびすべてのサブドメイン) へのすべての転送は、代わりに HTTPS を使用して行われます。 — バージョン <a href="#">2017.0-4100</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
web-x-frame-options	string	server	'DENY'	X-Frame-Options 値の設定 – デフォルト値は DENY に設定されます。これを変更する場合は、クリックジャック攻撃を防ぐために別の保護方法を導入する必要があります。他の保護機能がない場合は、この設定を変更しないでください。 – バージョン <a href="#">2017.1-5870</a> から使用可能。
ws-keepalive-interval	integer - DWORD (32 ビット)	server	10	WebSocket のキープアライブ間隔 – キープアライブメッセージを送信する間隔 (秒) を指定します。0 に設定すると、キープアライブメッセージは無効になります。 – バージョン <a href="#">2017.0-4100</a> から使用可能。

## display パラメータ

以下の表に、Linux NICE DCV サーバーの /etc/dcv/dcv.conf ファイルの [display] セクションの設定パラメータと、Windows NICE DCV サーバーの display レジストリキーを示します。

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
console-session-default-layout	string	セッション	[]	<p>コンソールセッションにおけるデフォルトの画面解像度と画面位置 — コンソールセッションにおけるデフォルトの画面解像度と画面位置を指定します。設定すると、DCV によって、リクエストされたレイアウトが起動時に設定されます。モニタごとに、解像度 (w,h) と位置 (x,y) で設定できます。指定されたモニタはすべて、有効になります。デフォルトレイアウトの値の例: [{ 'w':&lt;800&gt;, 'h':&lt;600&gt;, 'x':&lt;0&gt;, 'y': &lt;0&gt;}, { 'w':&lt;1024&gt;, 'h':&lt;768&gt;, 'x':&lt;800&gt;, 'y':&lt;0&gt;}] — バージョン <a href="#">2017.0-5600</a> から使用可能。</p>
cuda-devices	string	connection	[]	<p>ストリームエンコーディングに使用される CUDA デバイス — DCV がエンコーディングと CUDA ワークロードを分散するために使用するローカル CUDA デバイスのリストを指定します。各デバイスは、nvidia-smi コマンドから取得できる番号によって識別されます。例えば、cuda-devices= ['0', '2']</p>

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
				<p>は、DCV が 2 つの GPU を使用し、ID が 0 と 2 であることを示します。この設定は CUDA_VISIBLE_DEVICES 環境変数と似ていますが、DCV にのみ適用されます。このオプションが設定されていない場合、DCV は 0 から始まる増分セッションインデックスを使用して、次に使用するデバイスを選択します。—バージョン <a href="#">2017.2-6182</a> から使用可能。</p>
enable-client-resize	true ま たは false - DWORD (32 ビット)	セッ ション	Linux: true - Windows: 1	<p>クライアントによるディスプレイレイアウトの設定を許可するかどうか — クライアントによるディスプレイレイアウトの設定を許可するかどうかを指定します。—バージョン <a href="#">2017.0-4100</a> から使用可能。</p>
enable-qu	true ま たは false - DWORD (32 ビット)	セッ ション	Linux: true - Windows: 1	<p>品質更新を送信するかどうか — 品質更新を送信するかどうかを指定します。—バージョン <a href="#">2017.0-4100</a> から使用可能。</p>

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
enable-yuv444-encoding	string	セッション	'default-off'	YUV444 エンコーディングを有効にするかどうか — YUV444 エンコーディングを有効または無効にします。'always-on' の場合、サーバーは色精度が高くなるように最適化された YUV444 形式を優先します。'always-off' の場合、サーバーはストリーミングパフォーマンスに向けて最適化された形式を優先します。'default-on' と 'default-off' の値は同じ意味で、クライアントに決定を委ねます。使用できる値: 'always-on'、'always-off'、'default-on'、'default-off'。 — バージョン <a href="#">2022.0-11954</a> から使用可能。

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
grabber-target-fps	integer - DWORD (32 ビット)	セッション	0	フレームグラバの 1 秒あたりのターゲットフレーム — 1 秒あたりにフレームをグラブする上限です。値を 0 にすると、各フレームバッファリーダーの標準動作 (target-fps へのフォールバックや、グラブを制限しないなど) にデフォルト設定されます。すべてのフレームキャプチャバックエンドでこの設定が優先されるわけではありません。— バージョン <a href="#">2017.1-5870</a> から使用可能。
max-compressor-threads	integer - DWORD (32 ビット)	セッション	4	最大圧縮スレッド — 圧縮スレッドの最大数を指定します。— バージョン <a href="#">2017.0-4100</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
max-head-resolutio n	string	custom	(4096、2160)	<p>最大ヘッド解像度 — クライアントがリクエストできるディスプレイヘッドの最大解像度。ディスプレイヘッドは、ホストモニターに相当します。この設定は、クライアントのレイアウトリクエストごとに再ロードされます。クライアントに、より大きなヘッド解像度がリクエストされると、サーバーはこのオプションで設定された最大幅と最大高の値と一致するように解像度を調整します。サポートされている解像度の最大値は (4096、4096) です。— バージョン 2017.0-4100 <a href="#">から使用可能</a>。</p>

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
max-layout-area	integer - DWORD (32 ビット)	custom	0	<p>最大レイアウト領域 (ピクセル単位) — クライアントがリクエストできる表示レイアウトの最大領域 (ピクセル単位)。この制限より大きいレイアウトは無視されます。この最大値は、表示レイアウトジオメトリに制約を与えずに、送信する必要のある表示データの量に上限を提供することを目的としています。0 に設定すると、レイアウト領域に制限は適用されません。この設定は、クライアントのレイアウトリクエストごとに再ロードされます。 — バージョン <a href="#">2019.1-7423</a> から使用可能。</p>

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再ロ ード	デフォルト値	説明
max-num-heads	integer - DWORD (32 ビット)	custom	4	<p>ヘッドの最大数 — クライアントがリクエストできるディスプレイヘッドの最大数を指定します。ディスプレイヘッドは、ホストモニターに相当します。この設定は、クライアントのレイアウトリクエストごとに再ロードされます。クライアントからより多くのヘッドがリクエストされると、サーバーはその値がこのオプションで設定された値を超えないようにヘッドの数を調整します。 — バージョン <a href="#">2017.0-4100</a> から使用可能。</p>

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
min-head-resolution	string	custom	(640、480)	<p>最小ヘッド解像度 — クライアントがリクエストできるディスプレイヘッドの最小解像度。ディスプレイヘッドは、ホストモニターに相当します。この設定は、クライアントのレイアウトリクエストごとに再ロードされます。クライアントに、より小さい解像度がリクエストされると、サーバーはこのオプションで設定された最小幅と最大高の値と一致するように解像度を調整します。 — バージョン <a href="#">2017.0-4100</a> から使用可能。</p>
target-fps	integer - DWORD (32 ビット)	セッション	-1	<p>1 秒あたりの目標フレーム — 1 秒あたりに許可される最大フレームを指定します。値を 0 にすると、無制限に設定されます。値を -1 にすると、サーバー特性とセッションタイプに応じて target-fps 値が決定されます。2020.2 より前のバージョンでは、値 -1 は認識されず、デフォルト値は 25 になります。 — バージョン <a href="#">2017.0-4100</a> から使用可能。</p>

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再ロ ード	デフォルト値	説明
use-grabber-dirty-region	true ま たは false - DWORD (32 ビット)	セッ ション	Linux: true - Windows: 1	ダーティー領域を使用するか どうか — ダーティー画面領 域を使用するかどうかを指定 します。有効にすると、グラ バーは、画面からダーティー 領域における新しいフレーム を計算します。 — バージョン <a href="#">2017.0-4100</a> から使用可能。
web-client-max-head解像度	string	custom	(1920、1080)	ウェブクライアントの最大 ヘッド解像度 — ウェブクライ アントによってリクエストさ れるディスプレイヘッドの最 大解像度。ディスプレイヘッ ドは、ホストモニターに相当 します。この設定は、クライ アントのレイアウトリクエス トごとに再ロードされます。 この設定は、ウェブクライ アントが最大解像度を明示的に 設定している場合に無視さ れます。 max-head-resolutio n 制限オプションは、このオ プションで設定された最大幅 と高さの値の上に適用されま す。値が (0、0) に設定されて いる場合は無視されます。 — バージョン <a href="#">2020.0-8428</a> から 使用可能。

## display/linux パラメータ

次の表は、Linux NICE DCV サーバーの `/etc/dcv/dcv.conf` ファイルの `[display/linux]` セクションの設定パラメータを示しています。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
<code>gl-displays</code>	string	セッ ション	<code>[:0.0]</code>	3D 高速 X ディスプレイ — 仮想セッションで OpenGL レンダリングに DCV が使用するローカル 3D 高速 X 表示および画面のリストを指定します。この値がない場合は、仮想セッションで OpenGL アプリケーションを実行できません。この設定は、コンソールセッションでは無視されません。バージョン <a href="#">2017.0-4100</a> から使用可能。

## extensions パラメータ

以下の表に、Linux NICE DCV サーバーの `/etc/dcv/dcv.conf` ファイルの `[extensions]` セクションの設定パラメータと、Windows NICE DCV サーバーの `extensions` レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
enabled	true ま たは false - DWORD (32 ビット)	connectio n	Linux: true - Windows: 1	拡張機能を有効にするかどうか — 拡張機能を有効にするかどうかを指定します。拡張機能が無効になっている場合、ユーザーは DCV にサードパーティーの拡張機能を使用することはできません。 — バージョン <a href="#">2023.0-14852</a> から使用可能。

## input パラメータ

以下の表に、Linux NICE DCV サーバーの /etc/dcv/dcv.conf ファイルの [input] セクションの設定パラメータと、Windows NICE DCV サーバーの input レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
enable-autorepeat	true ま たは false - DWORD (32 ビット)	セッ ション	Linux: true - Windows: 1	Linux での自動リPEATを許可するかどうか — 単一のキーに対して自動リPEATを許可するかどうかを指定します。 — バージョン <a href="#">2017.2-6182</a> から使用可能。
enable-gamepad	true ま たは	セッ ション	Linux: true - Windows: 1	ゲームパッド入力を許可するかどうか — ゲームパッドを有

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
	false - DWORD (32 ビット)			効にするかどうかを指定しま す。 — バージョン <a href="#">2022.0-11 954</a> から使用可能。
enable-relative-mo use	true ま たは false - DWORD (32 ビット)	セッ ション	Linux: true - Windows: 1	相対マウス移動を許可するか どうか — 相対マウス移動を 許可するかどうかを指定しま す。 — バージョン <a href="#">2017.0-51 21</a> から使用可能。
enable-stylus	true ま たは false - DWORD (32 ビット)	セッ ション	Linux: true - Windows: 1	スタイラス入力を許可するか どうか — スタイラスを有効に するかどうかを指定します。 — バージョン <a href="#">2019.0-7318</a> か ら使用可能。
enable-touch	true ま たは false - DWORD (32 ビット)	セッ ション	Linux: true - Windows: 1	タッチ入力を許可するかどうか — タッチを有効にするか どうかを指定します。 — バ ージョン <a href="#">2017.3-6698</a> から使用 可能。

## license パラメータ

以下の表に、Linux NICE DCV サーバーの /etc/dcv/dcv.conf ファイルの [license] セクションの設定パラメータと、Windows NICE DCV サーバーの license レジストリキーを示します。

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
license-file	string	セッション	"	<p>ライセンス — EC2 以外のインスタンスで実行する場合には、DCV サーバーに使用するライセンスを指定します。ライセンスは RLM ライセンスによって付与されます。ライセンス仕様のリストを含めることができ、Windows では ';' で区切り、Linux では ':' で区切ります。各ライセンス仕様は、延長評価ライセンスの場合はローカルライセンスファイルに、フローティングライセンスの場合は「ポート@ホスト名」形式で指定された RLM サーバーのポートとホスト名にすることができます。複数のライセンスが指定されている場合、サーバーは最初のライセンスが検証されるまで (たとえば、ライセンスファイルが正しく認識されるか、リモート RLM サーバーに接続できるまで)、それぞれを順番に試行します。値が指定されていない場合、サーバーは Linux では '/usr/share/license.lic'、Windows では 'C:\Program Files\NICE\DCV\Server\license\license.lic' でデフォルト</p>

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
				トのライセンスファイルを検索します。デフォルトのライセンスファイルが見つからない場合は、デモライセンスが使用されます。このパラメータは EC2 インスタンスでは無視されます。—バージョン <a href="#">2017.0-4100</a> から使用可能。

## log パラメータ

以下の表に、Linux NICE DCV サーバーの `/etc/dcv/dcv.conf` ファイルの `[log]` セクションの設定パラメータと、Windows NICE DCV サーバーの `log` レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
<code>directory</code>	<code>string</code>	<code>server</code>	<code>"</code>	ログ出力ディレクトリ — ログの保存先を指定します。指定しない場合、Windows では「 <code>C:\ProgramData\NICE\DCV\log</code> 」、Linux では「 <code>/var/log/dcv/</code> 」がデフォルトになります。—バージョン <a href="#">2017.0-4100</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
enable-image-audit	true ま たは false - DWORD (32 ビット)	server	Linux: false - Windows: 0	転送された画像のコンテンツ 監査を有効にする — 転送さ れた画像のコンテンツを別の ファイルに保存する必要があ るかどうかを指定します。画 像はログサブディレクトリ に保存され、ファイル名は監 査 CSV ファイルで報告され ます。transfer-audit が無効 にされている場合、値は無 視されます。 — バージョン <a href="#">2023.0-14852</a> から使用可能。
level	string	custom	'info'	ログレベル — ログファイル の詳細レベルを指定します 。詳細レベル (詳細度が高い 順) は、'error'、'warn'、'in fo'、'debug' です。新しい値 は、設定上で変更され、DCV エージェントプロセスに伝播 されるとすぐに有効になりま す。バージョン <= 2019.1 の 場合、DCV エージェントプロ セスのログレベルは、開始時 にのみ設定されます。 — バ ージョン <a href="#">2017.0-4100</a> から使用 可能。

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
max-file-size	integer - DWORD (32 ビット)	server	0	ローテーション MegaBytes 前の のログファイルの最大サイズ — ローテーションがトリガーされる前のログファイルの最大サイズを指定します。値が '0' の場合、サイズによるローテーションは無効になり、ファイルを生成するプロセスが再開したときにファイルがローテーションされます。 — バージョン <a href="#">2022.1-13067</a> から使用可能。
rotate	integer - DWORD (32 ビット)	server	10	ログファイルのローテーションの数 — 削除される前にログファイルがローテーションされる回数を指定します。値を 0 にした場合、古いバージョンはローテーションされずに削除されます。 — バージョン <a href="#">2017.0-4100</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
rotation-interval	string	server	'none'	2 回連続するログファイル ローテーションの最大時間間 隔 — 2 回連続するログファ イルローテーションの最大 時間間隔を指定します。値 が 'none' の場合、ファイルは 時間に基づいてローテーショ ンされません。その他の可能 な値は、「every-minute」、 every-twenty-minutes「」、 「every-hour」、「every- day」です。 — バージョン <a href="#">2022.1-13067</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
rotation-suffix	string	server	'counter'	ローテーションされたログファイルに追加するサフィックス — ローテーションされたログファイルに追加するサフィックスを指定します。'counter' を指定すると、ローテーションされる各ログファイルに、単純に増加するカウンターのサフィックスが追加されます。'timestamp' を指定すると、'YYYY-MM-DD-HH-MM' 形式のタイムスタンプがログファイルに適用されません。そのタイムスタンプが適用されたローテーションするファイルがログフォルダにすでに存在する場合は、タイムスタンプに数値カウンターが追加されます。 — バージョン <a href="#">2022.1-13067</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
transfer-audit	string	server	'none'	監査を転送する方向 — 監査を転送する方向を指定します。このパラメータが有効になっている場合、新しい CSV ファイルのログはサーバーとクライアント間で転送されます。指定できる値は、「none」、server-to-client「」、client-to-server「」、 <code>all</code> です。この値が設定されていないか、「none」でない場合、転送監査は無効になり、ファイルは作成されません。 — バージョン <a href="#">2017.0-4100</a> から使用可能。

## printer パラメータ

以下の表に、Linux NICE DCV サーバーの `/etc/dcv/dcv.conf` ファイルの `[printer]` セクションの設定パラメータと、Windows NICE DCV サーバーの printer レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
file-printer-name	string	カスタム	'DCV printer'	ファイルダウンロード用の仮想 DCV プリンターの名前 —

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
				<p>DCV サーバー上の仮想 DCV プリンターの名前を表す文字列です。Linux では、この値は新しい Linux DCV セッションが作成されるたびに設定から読み取られます。この設定が空ではなく、値に PREFIX という文字列が含まれている場合、'PREFIX-SESSION-NUMBER' という名前の新しい仮想プリンターが CUPS に登録されます。この設定が空の場合、仮想 DCV プリンターは登録されません。Windows では、この設定はシステムのデフォルトプリンターを変更するために使用されます。空の文字列に設定すると、DCV によって現在の既定のプリンターは変更されません。—バージョン <a href="#">2022.0-11954</a> から使用可能。</p>

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
use-default-printer	string	カスタム	'client-decides'	<p>デフォルトプリンターの設定方法を決定する — どのプリンターをデフォルトプリンターとして設定するかをサーバーが決定します 指定できる値は、'client-decides'、'always-on'、'always-off' です。この設定の値が 'always-off' の場合、サーバーはいずれのプリンターもデフォルトとして設定しません。値が「always-on」の場合、は「printer」セクションのfile-printer-name「」設定で指定されているプリンターを設定します。値が 'client-decides' の場合、クライアントから送信されたデフォルトプリンターが設定されます。クライアントからデフォルトのプリンターが送信されない場合、「printer」セクションのfile-printer-name「」で指定されたプリンターが設定されます。デフォルト値は 'client-decides' です。現在 Windows でのみサポートされています。 — バージョン <a href="#">2022.2-13907</a> から使用可能。</p>

## redirection パラメータ

以下の表に、Linux NICE DCV サーバーの `/etc/dcv/dcv.conf` ファイルの `[redirection]` セクションの設定パラメータと、Windows NICE DCV サーバーの `redirection` レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
<code>enable-timezone-re direction</code>	string	セッ ション	'client-decides'	クライアントからサーバーへのタイムゾーンのリダイレクトを許可または拒否 — クライアントからサーバーへタイムゾーンがリダイレクトされることを許可または拒否します。指定できる値は、'always-on'、'always-off'、'client-decides' です。'always-on' に設定すると、優先ユーザークライアントがタイムゾーンをサーバーに送信し、それがサーバーのタイムゾーンに設定されます。'always-off' に設定すると、サーバーはクライアントに独自のタイムゾーンを表示します。クライアントのタイムゾーンメッセージはすべて破棄されます。'client-decides' に設定すると、優先ユーザークライアントが自分のタイムゾーンをサーバーに送信でき、それがサーバーのタイムゾーンに設定されます。クライアントは、自分

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
				のタイムゾーンをサーバーに送信しないことも選択できます。デフォルト値は 'client-decides' です。— バージョン <a href="#">2022.2-13907</a> から使用可能。

## security パラメータ

以下の表に、Linux NICE DCV サーバーの /etc/dcv/dcv.conf ファイルの [security] セクションの設定パラメータと、Windows NICE DCV サーバーの security レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
allowed-http-host-regex	string	server	'^.+\$\$'	許可されたホストの正規表現 — この DCV サーバーで処理可能なホスト名を表す正規表現パターンを指定します。着信 HTTP リクエストのホストヘッダーがこのパターンに一致していない場合、リクエストそのものが、HTTP 403 (禁止) ステータスコードで失敗します。これは、HTTP ホストヘッダー攻撃を回避するためのセキュリティ対策です

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
				<p>。Javascript のように有効な正規表現のパターンである必要があります。パターンの文字は、大文字と小文字の両方に一致します。例: <code>^(www\.)?example\.com\$</code> — バージョン <a href="#">2017.0-4100</a> から使用可能。</p>

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
allowed-ws-origin-regex	string	server	'^https://.+\$\$'	<p>許可されたオリジン — この DCV サーバーが受け入れるオリジンを表す正規表現パターンを指定します。WebSocket 接続を確立する場合、クライアントのハンドシェイクのオリジンヘッダーフィールドは、接続を確立するスクリプトのオリジンを示します。着信 HTTP リクエストのオリジンヘッダーがこのパターンに一致していない場合、リクエストそのものが、HTTP 403 (禁止) ステータスコードで失敗します。これは、クロスサイト WebSocket ハイジャック (CSWSH) 攻撃を防ぐためのセキュリティ対策です。JavaScript のように有効な正規表現のパターンである必要があります。パターンの文字は、大文字と小文字の両方に一致します。オリジンヘッダーの形式は &lt;scheme&gt; "://" &lt;host&gt; [ ":" &lt;port&gt; ] です。例: '^https://(www\.)?example\.com(:443)?\$\$'。 — バージョン <a href="#">2017.0-4100</a> から使用可能。</p>

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
auth-connection-setup-timeout	integer - DWORD (32 ビット)	server	120	認証チャンネル接続セットアップのタイムアウト — タイムアウトまでに認証チャンネル接続セットアップ手順の完了に費やす時間 (秒) を指定します。手順にこれ以上の時間がかかった場合、チャンネルは閉じられます。0 に設定すると、認証チャンネル接続セットアップのタイムアウトは無効になります。 — バージョン <a href="#">2017.0-4100</a> から使用可能。
auth-token-verifier	string	server	"	認証トークンの検証装置のエンドポイント — DCV サーバーで使用される認証トークンの検証装置のエンドポイント (URL) を指定します。空にすると、内部認証トークンの検証装置が使用されます。 — バージョン <a href="#">2017.0-4100</a> から使用可能。
auth-token-verifier-timeout	integer - DWORD (32 ビット)	server	100	認証トークン検証装置のタイムアウト (秒単位) — DCV サーバーによって認証トークン検証装置が使用されるのを待つ時間 (秒単位) を指定します。 — バージョン <a href="#">2023.0-14852</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
認証	string	server	'system'	認証メソッド — DCV サー バーで使用されるクライアン ト認証方法を指定します。ク ライアント認証を基盤となる オペレーティングシステムに 委任するには、「system」 を使用します。クライアン ト認証を無効にし、すべての クライアントへのアクセスを 付与するには、「none」を 使用します。 — バージョン <a href="#">2017.0-4100</a> から使用可能。
authentication-thr eshold	integer - DWORD (32 ビット)	server	3	認証しきい値 — サーバーの 接続が閉じられるまでの各ク ライアントの認証試行回数 を指定します。試行回数を 無制限に設定するには、0 を 使用します。 — バージョン <a href="#">2017.0-4100</a> から使用可能。
ca-file	string	server	"	CA ファイル — DCV サー バーで信頼されている認証機関 (CA) を含むファイルを指定し ます。空にすると、システム によって提供されるデフォル トの信頼ストアが使用されま す。 — バージョン <a href="#">2017.0-41 00</a> から使用可能。

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
certificate-to-user-file	string	カスタム	"	証明書からユーザーへのマッピングファイル — 証明書からユーザーへのマッピングリストを含むファイルを指定します。 — バージョン <a href="#">2022.0-11954</a> から使用可能。
ciphers	string	server	'ECDHE-RSA-AES128-GCM-SHA256:ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA384'	TLS 接続で使用される暗号リスト — TLS 接続で使用される暗号リストを指定します。暗号は、「:」文字を使用して区切る必要があります。また、openssl とクライアントでサポートされている必要があります。 — バージョン <a href="#">2017.0-4100</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
connection-estab-t imeout	integer - DWORD (32 ビット)	server	5	接続確立のタイムアウト — タイムアウトまでに接続手順 の完了に費やす時間 (秒) を指 定します。手順にこれ以上の 時間がかかった場合、接続は 閉じられます。0 に設定する と、接続の確立はタイムアウ トしません。 — バージョン <a href="#">2017.0-4100</a> から使用可能。
connection-setup-t imeout	integer - DWORD (32 ビット)	server	5	チャネル接続セットアップの タイムアウト — タイムアウ トまでにチャネル接続セット アップ手順の完了に費やす時 間 (秒) を指定します。手順に これ以上の時間がかかった場 合、チャネルは閉じられます 。0 に設定すると、チャネル 接続セットアップはタイムア ウトしません。 — バージョン <a href="#">2017.0-4100</a> から使用可能。
crl-file	string	カスタ ム	"	CRL ファイル — 証明書失効 リスト (CRL) を含むファイル を指定します。 — バージョン <a href="#">2022.0-11954</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
enable-gssapi	true ま たは false - DWORD (32 ビット)	server	Linux: false - Windows: 0	GSSAPI SASL メカニズムを有効にする — Kerberos を使用して DCV 認証を可能にする GSSAPI SASL メカニズムを有効または無効にします。 — バージョン <a href="#">2017.3-6698</a> から使用可能。
max-connections-per-user	integer - DWORD (32 ビット)	server	10	ユーザーの最大接続数 — ユーザーあたりに許可される同時接続の最大数を指定します。この数を超えると接続は拒否されます。 — バージョン <a href="#">2017.0-4100</a> から使用可能。
no-tls-strict	true ま たは false - DWORD (32 ビット)	server	Linux: false - Windows: 0	厳格な証明書検証を有効または無効にする — 外部認証トークンの検証装置に接続する際、厳格な証明書検証を有効または無効にします。認証トークンの検証装置で自己署名証明書が使用される場合は厳格な認証の検証を無効にする必要があります。 — バージョン <a href="#">2017.0-4100</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
os-auto-lock	true ま たは false - DWORD (32 ビット)	セッ ション	Linux: true - Windows: 1	最後のクライアント接続が終了したときに OS セッションをロックするかどうか — 有効にすると、最後のクライアント接続が閉じられたときに、OS セッションがロックされます。 — バージョン <a href="#">2017.1-5777</a> から使用可能。
pam-service-name	string	server	'dcv'	PAM サービス名 — DCV で使用される PAM 設定ファイルの名前を指定します。デフォルトの PAM サービス名は「dcv」であり、/etc/pam.d/dcv 設定ファイルに該当します。このパラメータは、「system」の認証メソッドを使用する場合にのみ使用されます。 — バージョン <a href="#">2017.0-4100</a> から使用可能。

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
passwd-file	string	server	"	パスワードファイル — ユーザーの認証情報を確認するために使用するパスワードファイルを指定します (dcv 認証モードの場合のみ)。空にすると、 <code>\${XDG_CONFIG_HOME}/NICE/dcv/passwd</code> (Linux の場合)、 <code>%CSIDL_LOCAL_APPDATA%\NICE\dcv\passwd</code> (Windows の場合) のデフォルトファイルが使用されます。 — バージョン <a href="#">2017.0-4100</a> から使用可能。
server-fqdn	string	server	"	サーバー FQDN — サーバーの完全修飾ドメイン名。空は <code>gethostname()</code> を意味します。 — バージョン <a href="#">2017.3-6698</a> から使用可能。
service-name	string	server	'dcv'	サービス名 — サービスの登録名 (通常はプロトコル名)。 — バージョン <a href="#">2020.0-8428</a> から使用可能。

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
supervision-control	string	カスタム	無効	<p>セッションの監視制御のタイプ — セッションの監視制御のタイプを指定します 指定できる値は 'disabled' と 'enforced' です。この値を 'enforced' に設定すると、監視対象外アクセス権を設定して、コラボレーションセッションのユーザーによる所有者なしのアクセスを許可または拒否できます。ユーザーに監視対象外アクセスを許可すると、ユーザーは所有者なしでセッションにアクセスできます。デフォルトでは、所有者を除くすべてのユーザーに対してこの権限が拒否されます。この値が 'disabled' (デフォルト) に設定されている場合、サーバーはこの監視制御と権限を適用しません。新しい値は、設定で変更されるとすぐに有効になります。 — バージョン <a href="#">2021.3-11591</a> から使用可能。</p>
user-realm	string	server	"	<p>サーバーユーザー領域 — サーバーのユーザー領域を指定します。 — バージョン <a href="#">2017.3-6698</a> から使用可能。</p>

## session-management パラメータ

以下の表に、Linux NICE DCV サーバーの `/etc/dcv/dcv.conf` ファイルの `[session-management]` セクションの設定パラメータと、Windows NICE DCV サーバーの `session-management` レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
create-session	true または false - DWORD (32 ビット)	server	Linux: false - Windows: 0	サーバースタートアップ時に コンソールセッションを作成 する — サーバースタートア ップ時に自動的にコンソール セッション (ID 「console」を 使用) が起動されるようにす るかどうかを指定します。 — バージョン <a href="#">2017.0-4100</a> から 使用可能。
enable-gl-in-virtual セッション	string	セッ ション	'default-on'	dcv-gl 機能を使用するかどうか — dcv-gl 機能を使用するかどうかを指定します (要ライセンス)。使用できる値: 'always-on'、'always-off'、'default-on'、'default-off'。 — バージョン <a href="#">2017.0-4100</a> から使用可能。
max-concurrent-clients	integer - DWORD (32 ビット)	セッ ション	-1	セッションあたりの最大同時 クライアント数 — セッショ ンあたりの最大同時クライ アント数を指定します。-1 に設 定すると、制限は適用されま せん。自動セッションのみに 制限を設定するには、「セッ

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
				セッション管理/」 セクションmax-concurrent-clientsの「」を使用しませんautomatic-console-session。 — バージョン 2017.0-4100 <a href="#">から使用可能</a> 。
max-concurrent-sessions	integer - DWORD (32 ビット)	server	0	同時セッションの最大数 — 許可される同時セッションの最大数を指定します。コンソールセッションは本質的に1つに制限されているため、この制限は現在、仮想セッションにのみ適用されます。制限を強制しない場合は、0を指定します。 — バージョン <a href="#">2019.0-7318</a> から使用可能。
max-sessions-per-user	integer - DWORD (32 ビット)	server	0	ユーザーあたりの最大セッション数 - 各ユーザーが所有できる同時セッションの最大数を指定します。この制限は現在、仮想セッションにのみ適用されます。制限を強制しない場合は、0を指定します。 — バージョン <a href="#">2021.0-10242</a> から使用可能。

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
virtual-session-default-layout	string	セッション	[]	<p>仮想セッションのデフォルトのレイアウト — これが設定されている場合、Xdcv はスタートアップ時に指定したレイアウトを作成するように設定されます。モニタごとに、解像度 (w,h) と位置 (x,y) で設定できます。指定されたモニタはすべて、有効になります。デフォルトのレイアウト例の値: [{"w":&lt;800&gt;, "h":&lt;600&gt;, "x":&lt;0&gt;, "y": &lt;0&gt;}, {"w":&lt;1024&gt;, "h":&lt;768&gt;, "x":&lt;800&gt;, "y":&lt;0&gt;}]</p> <p>この設定では、モニターの最大数 ( virtual-session-monitors 設定で指定) は配列内の要素数よりも優先されます。たとえば、5つのモニターが設定されていて、モニターの最大数が4の場合、最初の4つのモニターだけが作成されます。このキーが設定されている場合、有効なモニターの数 ( virtual-session-monitors 設定で指定) は無視されます。 — バージョン <a href="#">2017.0-5600</a> から使用可能。</p>

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
virtual-session-font-path	string	セッ ション	"	特殊なフォントのパスを追加するかどうか — 特殊なフォントのパスを指定します。アプリケーションによっては、特殊なフォントを X サーバーに渡す必要がある場合があります。 — バージョン <a href="#">2017.0-4100</a> から使用可能。
virtual-session-source-profile	true ま たは false - DWORD (32 ビット)	セッ ション	Linux: false - Windows: 0	セッションスターターでユーザープロファイルを取得するかどうか — セッションスタータースクリプトを実行するシェルがユーザープロファイルを取得するかどうかを指定します。デフォルトは false で、DCV は「bash--noprofile-norc」を使用してセッションスタータースクリプトを実行します。 — バージョン <a href="#">2021.3-11591</a> から使用可能。
virtual-session-xdcv-args	string	セッ ション	"	Xdcv に渡す追加の引数 — 必要に応じて、Xdcv に渡す追加の引数を指定します。 — バージョン <a href="#">2017.0-4334</a> から使用可能。

## session-management/automatic-console-session パラメータ

以下の表に、Linux NICE DCV サーバーの /etc/dcv/dcv.conf ファイルの [session-management/automatic-console-session] セクションの設定パラメータと、Windows NICE DCV サーバーの session-management/automatic-console-session レジストリキーを示します。

[パラメータ]	タイプ - Windows レジストリの種類	コンテキストの再ロード	デフォルト値	説明
client-eviction-policy	string	server	'reject-new-connection'	制限に達した場合のクライアント接続の処理方法を指定する — セッションあたりの同時接続クライアントの最大数に達した場合に、新しい接続を拒否するか、既存の接続を自動的に終了するかを指定します。使用できる値はreject-new-connection「」（着信接続は閉じられます）とsame-user-oldest-connection「」（サーバーは、セッションと長時間対話していない同じユーザーの接続を閉じるか、その情報がない場合は最も古い接続時間で終了します）です。 — バージョン <a href="#">2022.1-13067</a> から使用可能。
max-concurrent-clients	integer - DWORD	server	-1	セッションあたりの最大同時クライアント数 — セッションごとに許可する最大同時クラ

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
	(32 ビット)			イアント数を指定します。-1 に設定すると、制限は適用 されません。—バージョン <a href="#">2017.0-5600</a> から使用可能。
所有者	string	server	"	自動作成された「console」 セッションの所有者 — 「console」セッションの所 有者のユーザーネームを指定 します。空にすると、DCV サーバーを起動したユーザー が所有者になります。この 設定は、サーバー起動時に自 動的に作成した「console」 セッションにのみ適用されま す (create-session 設定を true にした場合)。—バージョン <a href="#">2017.0-5600</a> から使用可能。
permissions-file	string	server	"	「console」の自動セッシ ョンのアクセス許可ファイル —ユーザーによる DCV 機能 へのアクセスを確認するた めに使用するアクセス許可フ ァイルへのパスを指定しま す。空にすると、所有者にセ ッションへのフルアクセスが 許可されます。—バージョン <a href="#">2017.0-5600</a> から使用可能。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
storage-root	string	server	"	ファイルストレージのルートフォルダへのパス — コンソールセッションストレージに使用されるフォルダの完全パスを指定します。storage-root が空の場合や、フォルダが存在しない場合、ファイルストレージは無効になります。 — バージョン <a href="#">2017.0-5600</a> から使用可能。

## session-management/defaults パラメータ

以下の表に、Linux NICE DCV サーバーの /etc/dcv/dcv.conf ファイルの [session-management/defaults] セクションの設定パラメータと、Windows NICE DCV サーバーの session-management/defaults レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
permissions-file	string	セッ ション	"	すべてのセッションに含まれるデフォルトのアクセス許可 — 各セッション用にユーザーによって選択されるアクセス許可に自動的にマージされるアクセス許可ファイルへのパスを指定します。空にする

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
				と、'default.perm' ファイルが使用されます。このファイルは、/etc/dcv/ (Linux の場合)、または DCV のインストールフォルダ (例: 'C:\Program Files\NICE\DCV\Server\conf') (Windows の場合) にあります。 — バージョン <a href="#">2017.0-5600</a> から使用可能。

## smartcard パラメータ

以下の表に、Linux NICE DCV サーバーの /etc/dcv/dcv.conf ファイルの [smartcard] セクションの設定パラメータと、Windows NICE DCV サーバーの smartcard レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
enable-cache	string	custom	'default-on'	スマートカードキャッシュメッセージを有効にするかどうか — スマートカードキャッシュを有効または無効にします。有効にすると、NICE DCV サーバーは、クライアントのスマートカードから受信した最後の値をキャッシュ

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
				<p>します。その後の呼び出しでは、クライアントからではなく、サーバーのキャッシュから直接取得されます。これにより、クライアントとサーバー間で転送されるトラフィックの量が削減されるため、パフォーマンスが向上します。使用できる値には、'always-on'、'always-off'、'default-on'、'default-off' が含まれます。この値は、クライアントスマートカードアプリケーションが起動されるたびに設定から読み取られます。バージョン <a href="#">2017.2-6182</a> から使用可能。</p>

## webauthn パラメータ

以下の表に、Linux NICE DCV サーバーの /etc/dcv/dcv.conf ファイルの [webauthn] セクションの設定パラメータと、Windows NICE DCV サーバーの webauthn レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
enabled	true ま たは false - DWORD (32 ビット)	セッ ション	Linux: true - Windows: 1	Webauthn リダイレクト機能を有効にするかどうか — この設定は、WebAuthn リクエストのリダイレクトを制御します。有効にすると、ユーザーは、YubiKeyWindows Hello、その他のローカル認証機能を使用してウェブリソースを認証できます。この設定を無効にすると、WebAuthn リダイレクトは無効になり、ユーザーはローカル認証機能を使用できません。 — バージョン 2023.1-16220 から使用可能。 <a href="#">???</a>

## webcam パラメータ

以下の表に、Linux NICE DCV サーバーの /etc/dcv/dcv.conf ファイルの [webcam] セクションの設定パラメータと、Windows NICE DCV サーバーの webcam レジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
max-resolution	string	connectio n	(1280, 720)	ウェブカメラの最大解像度 — アプリケーションに適用されるウェブカメラの最大解像度

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
				を指定します。 — バージョン <a href="#">2021.0-10242</a> から使用可能。
preferred-resolution	string	connection	(480, 360)	好ましいウェブカメラの解像度 — クライアントによって提供される解像度のうち、優先するウェブカメラの解像度を指定します。指定した解像度がサポートされていない場合は、一致する解像度のうち最も近いものが選択され、アプリケーションに適用されます。値の1つを0に指定すると、ウェブカメラの共有が無効になります。 — バージョン <a href="#">2021.0-10242</a> から使用可能。

## windows パラメータ

次の表に、Windows NICE DCV サーバーのwindowsレジストリキーを示します。

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
disable-display-sleep	true または false -	セッ ション	Linux: true - Windows: 1	ディスプレイが省電力モードになるのを防ぐ — ディスプレイが省電力モードになら

[パラメータ]	タイプ - Windows レジス トリの 種類	コンテ キスト の再口 ード	デフォルト値	説明
	DWORD (32 ビット)			ないようにするかどうかを 指定します。—バージョン <a href="#">2017.0-4100</a> から使用可能。
printer	string	セッ ション	"	プリンターをデフォルトとし て設定する — 仮想 DCV プリ ンターの名前を指定します。 この名前を使用してシステム のデフォルトプリンターを変 更します。空の文字列に設定 すると、DCV によって現在の 既定のプリンターは変更され ません。非推奨: 「printerfile- printer-name」セクションの 「」を使用してください。— バージョン 2017.0-4100 <a href="#">から</a> <a href="#">使用可能</a> 。

## 設定パラメータの変更

このセクションでは、NICE DCV サーバーの設定パラメータを変更する方法について説明しま  
す。Windows サーバーのレジストリキー、Linux サーバーのセクション、パラメータの名前、タイ  
プ、有効値の詳細については、「[NICE DCV サーバーパラメータリファレンス](#)」を参照してくださ  
い。

### トピック

- [Windows NICE DCV サーバー](#)
- [Linux NICE DCV サーバー](#)

## Windows NICE DCV サーバー

Windows NICE DCV サーバーの場合は、Windows レジストリエディタ PowerShell、またはコマンドラインを使用して設定パラメータを変更します。

Windows レジストリエディタを使用して設定パラメータを変更するには

1. Windows レジストリエディタを開きます。
2. 以下のレジストリパスに移動します。

```
HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/
```

3. パラメータがあるレジストリキーを選択します。そのレジストリキーがない場合は、「[NICE DCV サーバーパラメータリファレンス](#)」に説明されている正確なキー名を使用して作成します。
4. そのパラメータを開きます (ダブルクリックします)。そのパラメータがない場合は、「[NICE DCV サーバーパラメータリファレンス](#)」に説明されているタイプと名前を使用して作成します。

を使用して設定パラメータを変更するには PowerShell

1. 管理者 PowerShell として を実行します。
2. 「[NICE DCV サーバーパラメータリファレンス](#)」に説明されているキー名を使用して、レジストリキーを追加します。

```
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. 「[NICE DCV サーバーパラメータリファレンス](#)」に説明されているタイプと名前を使用して、レジストリキーにパラメータを作成します。

```
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
```

コマンドラインを使用して起動設定を変更するには

1. 管理者としてコマンドラインを実行します。

2. 「[NICE DCV サーバーパラメータリファレンス](#)」に説明されているキー名、パラメータのタイプと名前を使用して、レジストリキーを作成し、パラメータを追加します。

```
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 サーバー

Linux NICE DCV サーバーの場合、設定パラメータはテキストエディタや `crudini` などのコマンドラインツールを使用して変更できます。

テキストエディタを使用して設定パラメータを変更するには

1. お好みのテキストエディタを使用して `/etc/dcv/dcv.conf` を開きます。
2. そのファイルで該当するセクションを見つけます。そのセクションがない場合は、「[NICE DCV サーバーパラメータリファレンス](#)」に説明されているセクション名を使用して追加します。

```
[section]
```

3. そのセクションでパラメータを見つけ、値を変更します。そのパラメータがセクションにない場合は、「[NICE DCV サーバーパラメータリファレンス](#)」に説明されているパラメータ名を使用して追加します。

```
parameter_name="parameter_value"
```

4. ファイルを保存して閉じます。

`crudini` を使用して設定パラメータを変更するには

「[NICE DCV サーバーパラメータリファレンス](#)」に説明されているセクション名とパラメータ名を使用して、セクションを作成し、パラメータを追加します。

```
$ sudo crudini --set /etc/dcv/dcv.conf section_name parameter_name 'parameter_value'
```

# NICE DCV のサポート終了

NICE DCV のサポート終了 (EOSL、End of Support Life) は、NICE DCV の特定のメジャーバージョン (およびそのすべてのマイナーバージョン) がサポートを受けなくなり、新しいバージョンとの互換性についてテストされなくなった時点を定義するものです。

NICE DCV サポートチームは、EOSL 前日まで、設定問題に対する全面的なサポートを継続します。不具合の解決と機能の要求は、NICE DCV サーバーと NICE DCV クライアントの最新バージョンにのみ反映されます。旧バージョンには反映されません。

EOSL の日付を過ぎると、それ以上のサポートとメンテナンスが提供されなくなります。また、互換性問題のテストも中止します。引き続きサポートを受けるには、最新の NICE DCV バージョンにアップグレードする必要があります。

## トピック

- [EOSL タイムライン](#)
- [お客様向けの EOSL パス](#)
- [EOSL に関するよくある質問](#)

## EOSL タイムライン

次の表に、NICE DCV メジャーバージョンの EOSL タイムラインを示します。

NICE DCV メジャー バージョン	初回リリース日	EOSL 日
NICE DCV 2016.x	2015 年 12 月 31 日	2021 年 3 月 31 日
NICE DCV 2017.x	2017 年 12 月 18 日	2021 年 12 月 31 日
NICE DCV 2019.x	2019 年 8 月 5 日	2022 年 12 月 31 日

NICE DCV メジャー バージョン	初回リリース日	EOSL 日
NICE DCV 2020.x	2020 年 4 月 16 日	2023 年 12 月 31 日
NICE DCV 2021.x	2021 年 4 月 12 日	2024 年 12 月 31 日
NICE DCV 2022.x	2022 年 2 月 23 日	2025 年 12 月 31 日
NICE DCV 2023.x	2023 年 5 月 3 日	2026 年 12 月 31 日

## お客様向けの EOSL パス

AWS で NICE DCV を稼働させている場合、NICE DCV のライセンスは不要です。お支払いいただくのはワークロードに使用する基本的な AWS リソースの料金のみとなります。現在使用している NICE DCV バージョンが EOSL の日付を過ぎている場合は、[NICE ダウンロードページ](#)を使用して最新の NICE DCV バージョンにアップグレードするか、または AWS Marketplace から [NICE DCV AMI](#) を使用して引き続きサポートを受けてください。

オンプレミスで、またはサードパーティーのクラウドサービスプロバイダーを使用して、NICE DCV を実行しており、現在使用している NICE DCV のバージョンが EOSL の日付を過ぎている場合は、リセラーまたはディストリビューターに連絡して、利用可能なアップグレードパスを検討してください。有効なサポート契約がある場合は、NICE DCV の最新バージョンに無償でアップグレードできます。NICE DCV ディストリビューターおよびリセラーについては、[NICE ウェブサイト](#)を参照してください。

## EOSL に関するよくある質問

1. オンプレミスまたはサードパーティーのクラウドサービスプロバイダーで、EOSL に達した NICE DCV のバージョンを使用していますが、既存のサポート契約があります。EOSL の影響を受けますか？

有効なサポート契約をお持ちの場合、NICE DCV サポート契約の条件により、NICE DCV ライセンスを追加料金なしで最新バージョンにアップグレードすることができます。この状況でしたら、影響は最小限に抑えられます。サポート契約の有効期限が切れた場合は、次の方法のいずれかを利用すれば、引き続き完全なサポートを受けることができます。

1. 新たな有料ライセンスを入手して、最新の NICE DCV バージョンにアップグレードする。
2. EOSL タイムラインに達する前にサポート契約を更新して、NICE DCV の最新バージョンへのアップグレードパスを入手する。
3. サポート契約の有効期限が切れた後のサポートサービスに対して、現在の料金の 70% に相当する回復手数料を支払うことで、古いサポート契約を再開する。

2. Amazon EC2 で EOSL に達した NICE DCV のバージョンを使用しています。どうすればサポートされているバージョンにアップグレードできますか？

Amazon EC2 でご利用いただいているお客様は、NICE DCV を追加料金なしでいつでも完全サポートバージョンにアップグレードすることができます。

3. サポートされている NICE DCV サーバーで EOSL に達した NICE DCV クライアントのバージョンを使用できますか？その逆の場合はどうですか？

使用できます。ただし、EOSL に達したバージョンにはバグ修正が適用されなくなるため、クライアントとサーバーソフトウェアの両方を最新バージョンにアップグレードすることを強くお勧めします。

# セキュリティ

のクラウドセキュリティが最優先事項 AWS です。AWS のお客様は、セキュリティを最も重視する組織の要件を満たすように構築されたデータセンターとネットワークアーキテクチャからメリットを得られます。

セキュリティは、AWS とユーザーの間で共有される責任です。[責任共有モデル](#)では、これをクラウドのセキュリティおよびクラウド内のセキュリティと説明しています。

- クラウドのセキュリティ — クラウドで AWS サービスを実行するインフラストラクチャを保護する責任 AWS は AWS にあります。AWS また、は、安全に使用できるサービスも提供します。コンプライアンス [AWS プログラム](#) コンプライアンスプログラムの一環として、サードパーティーの監査者は定期的にセキュリティの有効性をテストおよび検証。に適用されるコンプライアンスプログラムの詳細については NICE DCV、「[コンプライアンスプログラム AWS による対象範囲内のサービスコンプライアンスプログラム](#)」を参照してください。
- クラウドのセキュリティ — お客様の責任は、使用する AWS サービスによって決まります。また、お客様は、データの機密性、会社の要件、適用される法律や規制など、その他の要因についても責任を負います。

このドキュメントは、NICE を使用する際の責任共有モデルの適用方法を理解するのに役立ちます DCV。以下のトピックでは、セキュリティおよびコンプライアンスの目的 NICE DCV を達成するために を設定する方法を示します。また、NICE DCV リソースのモニタリングや保護に役立つ他の AWS のサービスの使用方法についても説明します。

## 内容

- [でのデータ保護 NICE DCV](#)
- [のコンプライアンス検証 NICE DCV](#)

## でのデータ保護 NICE DCV

責任 AWS [共有モデル](#)、NICE でのデータ保護に適用されます DCV。このモデルで説明されているように、AWS はすべての を実行するグローバルインフラストラクチャを保護する責任があります AWS クラウド。お客様は、このインフラストラクチャでホストされているコンテンツに対する管理を維持する責任があります。また、使用する AWS サービス のセキュリティ設定と管理タスクもユーザーの責任となります。データプライバシーの詳細については、[「データプライバシー](#)

[FAQ](#)」を参照してください。欧州でのデータ保護の詳細については、AWS「[セキュリティブログ](#)」の[AWS「責任共有モデル」とGDPR](#)ブログ記事を参照してください。

データ保護の目的で、認証情報を保護し AWS アカウント、AWS IAM Identity Center または AWS Identity and Access Management ( ) を使用して個々のユーザーを設定することをお勧めしますIAM。この方法により、それぞれのジョブを遂行するために必要な権限のみが各ユーザーに付与されます。また、次の方法でデータを保護することもお勧めします:

- 各アカウントで多要素認証 (MFA) を使用します。
- SSL/TLS を使用して AWS リソースと通信します。1TLS.2 が必要で、1.3 TLS をお勧めします。
- を使用して APIとユーザーアクティビティのログ記録を設定します AWS CloudTrail。
- AWS 暗号化ソリューションと、内のすべてのデフォルトのセキュリティコントロールを使用します AWS サービス。
- Amazon Macie などの高度なマネージドセキュリティサービスを使用します。これらは、Amazon S3 に保存されている機密データの検出と保護を支援します。
- コマンドラインインターフェイスまたは AWS を介して にアクセスするときに FIPS 140-3 検証済みの暗号化モジュールが必要な場合はAPI、FIPSエンドポイントを使用します。利用可能なFIPS エンドポイントの詳細については、[「連邦情報処理規格 \(FIPS\) 140-3」](#)を参照してください。

お客様の E メールアドレスなどの極秘または機密情報は、タグ、または名前フィールドなどの自由形式のテキストフィールドに配置しないことを強くお勧めします。これは、コンソール、NICE DCVまたは を使用して または他の AWS サービス を操作する場合API AWS CLIも同様です AWS SDKs。名前に使用する自由記述のテキストフィールドやタグに入力したデータは、課金や診断ログに使用される場合があります。URL を外部サーバーに提供する場合は、そのサーバーへのリクエストを検証URLするために認証情報を に含めないことを強くお勧めします。

## データ暗号化

セキュリティで保護されたサービスの重要な特徴として、情報はアクティブに使用されていないときに暗号化されます。

### 保管中の暗号化

NICE DCV 自体は顧客データを保存しません。NICE DCV サーバーホスト上のデータは、保管時に暗号化できます。NICE DCV でを使用する場合は AWS、「[Amazon EC2ユーザーガイド](#)」の「[保管時の暗号化](#)」セクションと、「[Amazon EC2ユーザーガイド](#)」の「[保管時の暗号化](#)」セクションを参照してください。

## 転送中の暗号化

NICE DCV クライアントとNICE DCVサーバーから送信されるすべてのデータは、HTTPS/TLS 接続を介してすべてを送信することで暗号化されます。

証明書の設定方法については、[TLS 証明書の管理](#) を参照してください。

## のコンプライアンス検証 NICE DCV

サードパーティーの監査者は、複数の コンプライアンスプログラムの一環として AWS サービスのセキュリティと AWS コンプライアンスを評価します。NICE DCV を使用してサービスにアクセスしても、そのサービスのコンプライアンスは変更されません。

特定のコンプライアンスプログラムの対象となる AWS サービスのリストについては、「[コンプライアンスAWS プログラムによる対象範囲内のサービス](#)」を参照してください。一般的な情報については、[AWS 「コンプライアンスプログラム」](#)を参照してください。

を使用して、サードパーティーの監査レポートをダウンロードできます AWS Artifact。詳細については、「[AWS Artifactでレポートをダウンロードする](#)」を参照してください。

を使用する際のお客様のコンプライアンス責任NICE DCVは、お客様のデータの機密性、貴社のコンプライアンス目的、適用される法律および規制によって決まります。では、コンプライアンスに役立つ以下のリソース AWS を提供しています。

- [セキュリティとコンプライアンスのクイックスタートガイド](#) – これらのデプロイガイドでは、アーキテクチャ上の考慮事項について説明し、セキュリティとコンプライアンスに重点を置いたベースライン環境を にデプロイする手順について説明します AWS。
- [AWS コンプライアンスリソース](#) – このワークブックとガイドのコレクションは、お客様の業界や地域に適用される場合があります。
- [「デベロッパーガイド」のルールによるリソースの評価](#) – この AWS Config サービスは、リソース設定が社内プラクティス、業界ガイドライン、および規制にどの程度準拠しているかを評価します。 AWS Config
- [AWS Security Hub](#) – この AWS サービスは、内のセキュリティ状態を包括的に把握 AWS し、セキュリティ業界標準とベストプラクティスへの準拠を確認するのに役立ちます。

# NICE DCV のリリースノートとドキュメント履歴

このページでは、NICE DCV のリリースノートとドキュメント履歴を掲載します。

トピック

- [NICE DCV リリースノート](#)
- [ドキュメント履歴](#)

## NICE DCV リリースノート

このセクションでは、NICE DCV の大幅な更新、機能リリース、バグ修正の概要について説明します。更新はすべてリリースデータ別に整理されています。お客様からお寄せいただいたフィードバックに対応するために、ドキュメントを頻繁に更新しています。

トピック

- [DCV 2023.1-16388 — 2024 年 7 月 3 日](#)
- [DCV 2023.1-16388 — 2024 年 3 月 5 日](#)
- [DCV 2023.1-16388 – 2023 年 12 月 19 日](#)
- [DCV 2023.1-16220 — 2023 年 11 月 9 日](#)
- [DCV 2023.0-15487 — 2023 年 6 月 29 日](#)
- [DCV 2023.0-15065 — 2023 年 5 月 3 日](#)
- [DCV 2023.0-15022 — 2023 年 4 月 21 日](#)
- [DCV 2023.0-14852 — 2023 年 3 月 28 日](#)
- [DCV 2022.2-14521 — 2023 年 2 月 17 日](#)
- [DCV 2022.2-14357 — 2023 年 1 月 18 日](#)
- [DCV 2022.2-14175 — 2022 年 12 月 21 日](#)
- [DCV 2022.2-14126 — 2022 年 12 月 9 日](#)
- [DCV 2022.2-13907 — 2022 年 11 月 11 日](#)
- [DCV 2022.1-13300 — 2022 年 8 月 4 日](#)
- [DCV 2022.1-13216 — 2022 年 7 月 21 日](#)
- [DCV 2022.1-13067 — 2022 年 6 月 29 日](#)

- [DCV 2022.0-12760 — 2022 年 5 月 23 日](#)
- [DCV 2022.0-12627 — 2022 年 5 月 19 日](#)
- [DCV 2022.0-12123 — 2022 年 3 月 23 日](#)
- [DCV 2022.0-11954 — 2022 年 2 月 23 日](#)
- [DCV 2021.3-11591 — 2021 年 12 月 20 日](#)
- [DCV 2021.2-11445 — 2021 年 11 月 18 日](#)
- [DCV 2021.2-11190 — 2021 年 10 月 11 日](#)
- [DCV 2021.2-11135 — 2021 年 9 月 24 日](#)
- [DCV 2021.2-11048 — 2021 年 9 月 1 日](#)
- [DCV 2021.1-10851 — 2021 年 7 月 30 日](#)
- [DCV 2021.1-10598 — 2021 年 6 月 10 日](#)
- [DCV 2021.1-10557 — 2021 年 5 月 31 日](#)
- [DCV 2021.0-10242 — 2021 年 4 月 12 日](#)
- [DCV 2020.2-9662 — 2020 年 12 月 4 日](#)
- [DCV 2020.2-9508 — 2020 年 11 月 11 日](#)
- [DCV 2020.1-9012 — 2020 年 9 月 30 日](#)
- [DCV 2020.1-9012 — 2020 年 8 月 24 日](#)
- [DCV 2020.1-8942 — 2020 年 8 月 3 日](#)
- [DCV 2020.0-8428 — 2020 年 4 月 16 日](#)
- [DCV 2019.1-7644 — 2019 年 10 月 24 日](#)
- [DCV 2019.1-7423 — 2019 年 9 月 10 日](#)
- [DCV 2019.0-7318 — 2019 年 8 月 5 日](#)
- [DCV 2017.4-6898 — 2019 年 4 月 16 日](#)
- [DCV 2017.3-6698 — 2019 年 2 月 24 日](#)
- [DCV 2017.2-6182 — 2018 年 10 月 8 日](#)
- [DCV 2017.1-5870 — 2018 年 8 月 6 日](#)
- [DCV 2017.1-5777 — 2018 年 6 月 29 日](#)
- [DCV 2017.0-5600 — 2018 年 6 月 4 日](#)
- [DCV 2017.0-5121 — 2018 年 3 月 18 日](#)

- [DCV 2017.0-4334 — 2018 年 1 月 24 日](#)
- [DCV 2017.0-4100 — 2017 年 12 月 18 日](#)

## DCV 2023.1-16388 — 2024 年 7 月 3 日

ビルド番号	変更とバグ修正	
• nice-dcv-server: 16388	• ネイティブクライアントに対するバグ修正とパフォーマンスの向上。	
• nice-dcv-client(Windows): 9127		
• nice-dcv-viewer (macOS): 6703		
• nice-dcv-viewer (Linux): 6703		
• nice-xdcv: 565		
• nice-dcv-gl: 1047		
• nice-dcv-		

ビルド番号	変更とバグ修正	
gltest: 325		
• nice-dcv-simple-external-認証: 228		

## DCV 2023.1-16388 — 2024 年 3 月 5 日

ビルド番号	変更とバグ修正	
• nice-dcv-server: 16388	• ディスプレイのスケーリングがとは異なる値に設定されている場合の Windows クライアントの拡張機能の問題を修正しました100%。	
• nice-dcv-client(Windows): 8993	• Windows クライアントの相対マウスモードと高 DPI マウスの問題を修正しました。 • Windows クライアントの Shiftキーを使用したキーボードの組み合わせのリリースに関する問題を修正しました。	
• nice-dcv-viewer (macOS): 6203		
• nice-dcv-viewer		

ビルド番号	変更とバグ修正	
(Linux): 6203		
• nice-xdcv: 565		
• nice-dcv-gl: 1047		
• nice-dcv-gltest: 325		
• nice-dcv-simple-ex ternal- 認証: 228		

## DCV 2023.1-16388 – 2023 年 12 月 19 日

ビルド番号	変更とバグ修正	
• nice-dcv-server: 16388	<ul style="list-style-type: none"> <li>ストリーミング障害や過剰なログ記録を発生させる可能性のある Windows でのエージェント起動時の競合状態を修正しました。</li> </ul>	
• nice-dcv-client(Wi	<ul style="list-style-type: none"> <li>アイドルタイムアウトの設定が実行時に変更された場合に <code>dcv list-connections</code> で報告される、最後の対応時間を修正しました。</li> </ul>	

ビルド番号	変更とバグ修正	
Windows): 8934 • nice- dcv- viewer (macOS 6203	<ul style="list-style-type: none"> <li>• Windows サーバーでの NVIDIA GRID ドライバー 528.89 との互換性の問題を修正しました。</li> <li>• ストリーミング障害を発生させる可能性のあるウェブクライアントでのビデオデコードに関する問題を修正しました。</li> </ul>	
• nice- dcv- viewer (Linux): 6203	<ul style="list-style-type: none"> <li>• サーバーでディスプレイ解像度の変更が無効になっている場合の、Windows クライアントのマルチモニターでの全画面表示に関する問題を修正しました。</li> <li>• Linux および macOS クライアントでのウェブカメラ解像度に関する問題を修正しました。</li> </ul>	
• nice- xdcv: 565	<ul style="list-style-type: none"> <li>• Linux および macOS クライアントでのマウスのダブルクリックとトリプルクリックに関する問題を修正しました。</li> </ul>	
• nice- dcv-gl: 1047	<ul style="list-style-type: none"> <li>• Linux および macOS クライアントで WebAuthN リダイレクトの問題を修正しました。</li> </ul>	
• nice- dcv- gltest: 325		
• nice- dcv- simple- ex ternal- 認証: 228		

## DCV 2023.1-16220 — 2023 年 11 月 9 日

ビルド番号	新機能	変更とバグ修正
<ul style="list-style-type: none"> <li>• nice-dcv-server: 16220</li> <li>• nice-dcv-client(Windows): 8908</li> <li>• nice-dcv-viewer (macOS): 6125</li> <li>• nice-dcv-viewer (Linux): 6125</li> <li>• nice-xdcv: 565</li> <li>• nice-dcv-gl: 1047</li> <li>• nice-dcv-gltest: 325</li> </ul>	<p>NICE DCV に次の機能を追加しました。</p> <ul style="list-style-type: none"> <li>• リモート Google Chrome または Microsoft Edge ブラウザで実行されているウェブアプリケーションからのセッション WebAuthN リクエストのリダイレクトのサポート。リダイレクトされたリクエストはクライアントにチャネルされ、YubiKey や Windows Hello などの FIDO2 準拠の認証者がユーザー ID を検証できるようにします。</li> <li>• Windows ホスト用の新しい間接ディスプレイドライバ (IDD) は、グラフィックパイプラインを最適化し、プロトコルによる全体的な CPU 使用率を大幅に削減します。</li> <li>• Windows パフォーマンスカウンターを使用して、フレームレート、ネットワーク帯域幅、CPU 使用率などのさまざまな DCV プロトコルメトリックを追跡できるようになりました。これにより、ユーザーはネットワークと DCV プロトコルのパフォーマンスを把握できます。</li> </ul>	<ul style="list-style-type: none"> <li>• Windows のクリップボードでの透明画像のサポートが追加されました。</li> <li>• Windows のクリップボードへの同時アクセス時に、一部のアプリケーションでカットアンドペースト操作が正常に行われな問題を修正しました。</li> <li>• Windows でモニターのスケールリング係数が 100% NICE DCV サーバーにリセットされる問題を修正しました</li> <li>• Windows と Linux のコンソールセッションで、ユーザーのログアウト時と画面ロック時にクライアントを自動的に切断する設定を追加しました。</li> <li>• ノイズやサウンドアーティファクトの原因となるオーディオスタックの問題を修正しました。</li> <li>• ウェブカメラのストリーミングは、サーバー上でアプリケーションを終了しなくても、再接続すると再開できます。</li> <li>• Windows ネイティブクライアントでの高 DPI マウスを使用した際のマウスの動作が改善されました。</li> </ul>

ビルド番号	新機能	変更とバグ修正
• nice-dcv-simple-ex-ternal-認証: 228		<ul style="list-style-type: none"><li>• macOS ネイティブクライアントで SmartCard のサポートに関する問題を修正しました。</li><li>• Linux ネイティブクライアントでの高ピクセル密度のサポートが修正されました。</li><li>• Web クライアントと Windows ネイティブクライアントでのユーザーインターフェイスのアクセシビリティが改善されました。</li><li>• macOS で ウェブクライアントを使用する際の、一部のキーボードレイアウトの制限を修正しました。</li><li>• サードパーティの依存関係を最新バージョンに更新しました。</li><li>• Xdcv が XServer のバージョン 21.1.9 に更新されました。</li><li>• Windows Server 2012R2、Ubuntu 18.04、Suse Enterprise Linux 15SP4 のサポートを削除しました。</li><li>• バグを修正してパフォーマンスを改善しました。</li></ul>

## DCV 2023.0-15487 — 2023 年 6 月 29 日

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-server: 15487</li> </ul>	<ul style="list-style-type: none"> <li>Chrome 114 以降を使用しているときに間違った色が表示される可能性がある Web クライアントの問題を修正しました。</li> </ul>
<ul style="list-style-type: none"> <li>nice-dcv-client(Windows): 8771</li> </ul>	<ul style="list-style-type: none"> <li>アンインストール時のエラーを防ぐために、NICE DCV サーバーと Xdcv の el7 rpm パッケージを修正しました。</li> <li>Windows サーバーでの NVIDIA GRID ドライバー 528.89 との互換性の問題を修正しました。</li> </ul>
<ul style="list-style-type: none"> <li>nice-dcv-viewer (macOS): 5629</li> </ul>	<ul style="list-style-type: none"> <li>一部の Windows アプリケーションでクリップボードが正しく動作しなくなる問題を修正しました。</li> <li>パッケージのインストールまたは更新時に設定が正しいことを確認するために、dcv-gl パッケージで最新バージョンの NICE DCV サーバーパッケージが必要になりました。</li> </ul>
<ul style="list-style-type: none"> <li>nice-dcv-viewer (Linux): 5629</li> </ul>	<ul style="list-style-type: none"> <li>サイズ変更後に誤った解像度が使用される可能性がある Windows クライアントの問題を修正しました。</li> </ul>
<ul style="list-style-type: none"> <li>nice-xdcv: 551</li> </ul>	<ul style="list-style-type: none"> <li>macOS クライアントおよび Linux クライアントでの IPv6 アドレスのサポートを修正しました。</li> </ul>
<ul style="list-style-type: none"> <li>nice-dcv-gl: 1039</li> </ul>	<ul style="list-style-type: none"> <li>macOS クライアントで Control + クリックを右クリックとして設定できるようになりました。</li> </ul>
<ul style="list-style-type: none"> <li>nice-dcv-gltest: 318</li> </ul>	<ul style="list-style-type: none"> <li>Web クライアントで、サポートされているブラウザで全画面表示時に特殊キーとキーの組み合わせを使用できるようになりました。</li> </ul>

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-simple-external-認証: 208</li> </ul>	<ul style="list-style-type: none"> <li>OpenSSL サードパーティライブラリを更新しました。</li> </ul>

## DCV 2023.0-15065 — 2023 年 5 月 3 日

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-server: 15065</li> <li>nice-dcv-client(Windows): 8671</li> <li>nice-dcv-viewer (macOS): 5483</li> <li>nice-dcv-viewer (Linux): 5483</li> </ul>	<ul style="list-style-type: none"> <li>ライセンストークンのリリースが妨げられることがある close-session の問題を修正しました。</li> <li>の macOS ネイティブクライアントのクラッシュを修正しました BigSur。</li> </ul>

ビルド番号	変更とバグ修正	
<ul style="list-style-type: none"> <li>nice-xdcv: 547</li> <li>nice-dcv-gl: 1027</li> <li>nice-dcv-gltest: 318</li> <li>nice-dcv-simple-external-認証: 208</li> </ul>		

## DCV 2023.0-15022 — 2023 年 4 月 21 日

ビルド番号	変更とバグ修正	
<ul style="list-style-type: none"> <li>nice-dcv-server: 15022</li> <li>nice-dcv-client(Windows): 8671</li> </ul>	<ul style="list-style-type: none"> <li>画面のサイズを変更するとストリーミングが正しく動作しなくなる同時実行の問題を修正しました。</li> <li>QUIC 接続で障害を引き起こす NICE DCV サーバーの競合状態を修正しました。</li> <li>隠しカーソルがあるアプリケーションに関連する NICE DCV サーバーのクラッシュを修正しました。</li> </ul>	

ビルド番号	変更とバグ修正	
<ul style="list-style-type: none"> <li>nice-dcv-viewer (macOS) 5456</li> </ul>	<ul style="list-style-type: none"> <li>Windows サーバーでの日本語キーボード入力の問題を修正しました。</li> <li>Web カメラのストリーミングにおけるオーディオ/ビデオ同期が改善されました。</li> <li>ICU と libxml2 のサードパーティライブラリを更新しました。</li> </ul>	
<ul style="list-style-type: none"> <li>nice-dcv-viewer (Linux): 5456</li> </ul>	<ul style="list-style-type: none"> <li>Xdcv を XServer のバージョン 21.1.8 に更新し、仮想セッションが開始できなくなる XKB の問題を修正しました。</li> </ul>	
<ul style="list-style-type: none"> <li>nice-xdcv: 547</li> </ul>	<ul style="list-style-type: none"> <li>Windows、macOS、Linux のネイティブクライアントでビデオのデコードが失敗する問題を修正しました。</li> </ul>	
<ul style="list-style-type: none"> <li>nice-dcv-gl: 1027</li> </ul>	<ul style="list-style-type: none"> <li>macOS と Linux のネイティブクライアントの設定に関する問題を修正しました。</li> </ul>	
<ul style="list-style-type: none"> <li>nice-dcv-gltest: 318</li> </ul>		
<ul style="list-style-type: none"> <li>nice-dcv-simple-external-認証: 206</li> </ul>		

## DCV 2023.0-14852 — 2023 年 3 月 28 日

ビルド番号	新機能	変更とバグ修正
<ul style="list-style-type: none"> <li>• nice-dcv-server: 14852</li> <li>• nice-dcv-client(Windows): 8655</li> <li>• nice-dcv-viewer (macOS): 5388</li> <li>• nice-dcv-viewer (Linux): 5388</li> <li>• nice-xdcv: 527</li> <li>• nice-dcv-gl: 1022</li> <li>• nice-dcv-gltest: 318</li> </ul>	<p>NICE DCV に次の機能を追加しました。</p> <ul style="list-style-type: none"> <li>• macOS および Linux 上の NICE DCV クライアントの一部のモニターで全画面表示のサポートを追加しました。</li> <li>• すべてのクライアントで、ドラッグアンドドロップによるファイルのアップロード開始のサポートが追加されました。</li> <li>• Added Red Hat Enterprise Linux 9、Rocky Linux 9、CentOS Stream 9 を追加しました。</li> <li>• Linux の NICE DCV サーバーでのタイムゾーンリダイレクトのサポートを追加しました。</li> </ul>	<ul style="list-style-type: none"> <li>• 帯域幅の推定が不正確になり、視覚アーティファクトの発生にもつながる QUIC トランスポートの問題の一部を修正しました。</li> <li>• macOS および Linux クライアントのユーザーインターフェイスを更新しました。</li> <li>• Windows インストーラーで、ユーザーに表示されるアプリケーション名に一貫して NICE DCV が使用されるようになりました。</li> <li>• 堅牢性の向上のため、Windows でのクリップボードサポートの実装が修正されました。</li> <li>• Windows でドイツ語のキーボードレイアウトを使用する際の Caps Lock キーの問題を修正しました。</li> </ul>

ビルド番号	新機能	変更とバグ修正
• nice-dcv-simple-external-認証: 206		

## DCV 2022.2-14521 — 2023 年 2 月 17 日

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>• nice-dcv-server: 14521</li> <li>• nice-dcv-client(Windows): 8570</li> <li>• nice-dcv-viewer (macOS ): 5125</li> <li>• nice-dcv-viewer (Linux): 4804</li> <li>• nice-xdcv: 519</li> <li>• nice-dcv-gl: 1012</li> <li>• nice-dcv-glttest: 307</li> <li>• nice-dcv-simple-external-認証者: 198</li> </ul>	<ul style="list-style-type: none"> <li>• macOS クライアントの日本語キーボードとスペイン語のキーボードの問題を修正しました。</li> <li>• Windows NICE DCV サーバーのテンキーに関する問題を修正しました。</li> <li>• QUIC 接続でのメモリリークを修正しました。</li> <li>• 古いビデオドライバーを使用する場合の Windows NICE DCV クライアントの安定性が向上しました。</li> <li>• OpenSSL と libsoup のサードパーティーライブラリを更新しました。</li> <li>• Xdcv が XServer のバージョン 21.1.7 に更新されました。</li> </ul>

## DCV 2022.2-14357 — 2023 年 1 月 18 日

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-server: 14357</li> <li>nice-dcv-client(Windows): 8522</li> <li>nice-dcv-viewer (macOS ): 4804</li> <li>nice-dcv-viewer (Linux): 4804</li> <li>nice-xdcv: 487</li> <li>nice-dcv-gl: 1012</li> <li>nice-dcv-gltest: 307</li> <li>nice-dcv-simple-external-認証者: 198</li> </ul>	<ul style="list-style-type: none"> <li>Suse パッケージの最新の更新で発生し始めた、Suse Linux 12 での仮想セッションのクラッシュを修正しました。</li> <li>XPM の処理に関連する DCV-GL のメモリリークを修正しました。</li> <li>DCV-GL を xrestop ツールと統合して、XPM が対応するプロセスに関連付けられるようにしました。</li> <li>Windows サーバー上のウェブカメラとオーディオリダイレクトを改善して、Windows のネイティブの動作とより一貫性のあるものにしました (OS イベントが発生してもストリームは中断されません)。</li> <li>Windows NICE DCV クライアントがインプットメソッドを処理する方法を改善しました。</li> <li>Windows NICE DCV Client のクリップボードで、行区切り文字としてキャリッジリターン文字のみを使用するテキストに関する問題を修正しました。</li> </ul>

## DCV 2022.2-14175 — 2022 年 12 月 21 日

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-server: 14175</li> <li>nice-dcv-client(Windows): 8472</li> <li>nice-dcv-viewer (macOS ): 4804</li> <li>nice-dcv-viewer (Linux): 4804</li> <li>nice-xdcv: 487</li> </ul>	<ul style="list-style-type: none"> <li>WebSocket 接続使用時のサーバー内のファイル記述子のリークを修正しました。</li> <li>Xdcv が XServer のバージョン 21.1.6 に更新されました。</li> </ul>

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-gl: 983</li> <li>nice-dcv-gltest: 307</li> <li>nice-dcv-simple-external-認証者: 198</li> </ul>	

## DCV 2022.2-14126 — 2022 年 12 月 9 日

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-server: 14126</li> <li>nice-dcv-client(Windows): 8472</li> <li>nice-dcv-viewer (macOS ): 4804</li> <li>nice-dcv-viewer (Linux): 4804</li> <li>nice-xdcv: 481</li> <li>nice-dcv-gl: 983</li> <li>nice-dcv-gltest: 301</li> <li>nice-dcv-simple-external-認証者: 198</li> </ul>	<ul style="list-style-type: none"> <li>Windows サーバーで韓国語キーボードを使用する際の問題を修正しました。</li> <li>Windows 11 でハングが発生する可能性がある、Windows サーバー上の USB リダイレクトの問題を修正しました。</li> <li>'rotate' パラメーターが 0 に設定されている場合の、サーバーでのログローテーションの問題を修正しました。</li> <li>特定のネットワーク状況でストリームがフリーズする macOS クライアントおよび Linux クライアントの問題を修正しました。</li> <li>全画面表示にするとサイズが正しく変更されない Windows ネイティブクライアントの問題を修正しました。</li> <li>ファイルのアップロード時にクラッシュする macOS と Linux の問題を修正しました。</li> <li>オーディオが機能しなくなる macOS クライアントの問題を修正しました。</li> <li>NVIDIA GPU を使用するとクラッシュする Linux クライアントの問題を修正しました。</li> <li>タイムゾーンリダイレクト UI がサーバーと同期しなくなるウェブクライアントの問題を修正しました。</li> </ul>

ビルド番号	変更とバグ修正
	<ul style="list-style-type: none"> <li>セッション後のページを読み込めないウェブクライアントの問題を修正しました。</li> <li>LibTIFF と MIT-Kerberos のオープンソース依存関係を更新しました。</li> </ul>

## DCV 2022.2-13907 — 2022 年 11 月 11 日

ビルド番号	新機能	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-server: 13907</li> <li>nice-dcv-client(Windows): 8427</li> <li>nice-dcv-viewer (macOS): 4653</li> <li>nice-dcv-viewer (Linux): 4653</li> <li>nice-xdcv: 481</li> </ul>	<p>NICE DCV に次の機能を追加しました。</p> <ul style="list-style-type: none"> <li>Windows の NICE DCV クライアントの一部のモニターに、全画面表示のサポートを追加しました。</li> <li>macOS で、高ピクセル密度ディスプレイのネイティブクライアントのサポートが追加されました。</li> <li>macOS および Linux で NICE DCV クライアントのプリンターリダイレクトを追加しました。</li> <li>Windows の NICE DCV サーバーでのタイムゾーンリダイレクトのサポートを追加しました。</li> <li>コンソールセッションのシングルサインオンをサポートするために、Ubuntu 22.04 の GNOME シェル拡張を追加しました。</li> <li>オープンソースドライバーを使用する際に、AMD GPU に VA-API ベースのエンコーダーを追加しました。</li> </ul>	<ul style="list-style-type: none"> <li>ウェブクライアントのユーザーインターフェイスを Cloudscape デザインスタイルに更新しました。</li> <li>クライアントの再接続によってトリガーされるエージェント内のメモリリークが修正されました。</li> <li>Ubuntu 20.04 で仮想セッションを使用する際に、GDM3 を使用するシステムのサポートを追加しました。</li> <li>Ubuntu 20.04 の仮想セッションで断続的に黒い画面が表示される問題を修正しました。</li> <li>ウェブクライアントのタブ変更時にクリップボードが更新されない問題を修正しました。</li> <li>数字用キーパッドの Enter キーの問題を修正しました。</li> </ul>

ビルド番号	新機能	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-gl: 983</li> <li>nice-dcv-gltest: 301</li> <li>nice-dcv-simple-external-認証者: 198</li> </ul>		

## DCV 2022.1-13300 — 2022 年 8 月 4 日

ビルド番号	変更とバグ修正
<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-external-認証: 193</li> </ul>	<ul style="list-style-type: none"> <li>複数のコラボレーターがセッションに接続しているときに Windows が自動的にロック解除されないようにしました。</li> <li>指定した証明書ファイルをサーバーが読み込めない問題を修正しました。</li> <li>macOS クライアントでオーディオが歪む問題を修正しました。</li> </ul>

## DCV 2022.1-13216 — 2022 年 7 月 21 日

ビルド番号	変更とバグ修正
<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-external-認証: 193</li> </ul>	<ul style="list-style-type: none"> <li>2019.1 以前の NICE DCV サーバーへの接続に失敗する、すべてのクライアントでの問題を修正しました。</li> <li>Windows サーバーでの SmartCard リダイレクトの問題を修正しました。</li> <li>GPU を搭載するホストの NICE DCV サーバーに接続する際にストリーミングが失敗する問題を修正しました。</li> </ul>

## DCV 2022.1-13067 — 2022 年 6 月 29 日

ビルド番号	新機能	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-server: 13067</li> <li>nice-dcv-client(Windows): 8248</li> <li>nice-dcv-viewer (macOS ): 4241</li> </ul>	<p>NICE DCV に次の機能を追加しました。</p> <ul style="list-style-type: none"> <li>サーバーで Ubuntu 22.04 と Rocky Linux 8.5 以降のサポートを追加しました。</li> <li>ネイティブクライアントでの Ubuntu 22.04 のサポートが追加されました。</li> <li>Windows、macOS、および Linux ネイティブクライアントのコラボレーションエクスペリエンスが向上しました。</li> </ul>	<ul style="list-style-type: none"> <li>パフォーマンスが向上し、GPU 以外のサーバーで全体の CPU 消費量が最大 30% 削減されました。</li> <li>ログローテーションを設定で構成できるようになり、時間間隔やサイズ制限を指定できるようになりました。</li> <li>初期ハンドシェイクが失敗する可能性のある QUIC トランスポートの問題を修正しました。</li> <li>Linux サーバー上の相対的なマウスモーションが一部のアプリケーションで期待どおりに動作</li> </ul>

ビルド番号	新機能	変更とバグ修正
<ul style="list-style-type: none"> <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-external-認証: 193</li> </ul>		<p>しないことがある問題を修正しました。</p>

## DCV 2022.0-12760 — 2022 年 5 月 23 日

ビルド番号	変更とバグ修正
<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> </ul>	<p>変更:</p> <p>web-url-path オプションを指定するときにウェブクライアントが正常に接続できない問題を修正しました。</p>

ビルド番号	変更とバグ修正
<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-external-認証: 188</li> </ul>	

## DCV 2022.0-12627 — 2022 年 5 月 19 日

ビルド番号	変更とバグ修正
<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> <li>nice-xdcv: 424</li> <li>nice-dcv-gl: 961</li> <li>nice-dcv-gltest: 291</li> <li>nice-dcv-simple-external-認証: 188</li> </ul>	<p>変更:</p> <ul style="list-style-type: none"> <li>帯域幅の推定が不正確になり、視覚アーティファクトの発生にもつながる QUIC トランスポートの問題の一部を修正しました。</li> <li>Windows サーバーのインストーラーのオーディオサービスで、アップデートプロセスが失敗する問題を解決しました。</li> <li>Windows クライアントのインストーラーの USB 処理で、アンインストールプロセスが失敗する問題を解決しました。</li> <li>macOS と Linux クライアントでスクリーンショットを保存する際の問題を修正しました。</li> <li>OpenSSL、zlib、gdk-pixbuf のサードパーティライブラリを更新しました。</li> </ul>

## DCV 2022.0-12123 — 2022 年 3 月 23 日

ビルド番号	新機能	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-server: 12123</li> </ul>	NICE DCV に次の機能を追加しました。	変更:

ビルド番号	新機能	変更とバグ修正
<ul style="list-style-type: none"> <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-認証: 188</li> </ul>	<ul style="list-style-type: none"> <li>macOS および Linux クライアントで高い色精度を有効にするオプションを追加しました。</li> </ul>	<ul style="list-style-type: none"> <li>QUIC トランスポートを使用する際の帯域幅推定と画質が向上しました。</li> </ul> <p>修正内容:</p> <ul style="list-style-type: none"> <li>NVIDIA ドライバー 510.xx を使用する際の、Linux のコンソールセッションの視覚アーティファクトを修正しました。</li> <li>Windows ネイティブクライアントで Bluetooth 経由で接続されている DualShock 4 つのコントローラーの問題を修正しました。</li> <li>macOSクライアントで Web カメラを有効にするときに発生する可能性があったクラッシュを修正しました。</li> </ul>

## DCV 2022.0-11954 — 2022 年 2 月 23 日

ビルド番号	新機能	変更とバグ修正
<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> </ul>	<p>NICE DCV に次の機能を追加しました。</p> <ul style="list-style-type: none"> <li>Windows サーバーと Windows ネイティブクライアントのゲームコントローラーのサポート。</li> <li>NICE DCV ウェブクライアントは、それをサポートす</li> </ul>	<p>変更:</p> <ul style="list-style-type: none"> <li>NICE DCV サーバーを再起動しなくても TLS 証明書を更新できるようになりました。</li> <li>特定のネットワークインターフェイスまたは特定の IPv4 または IPv6 アドレス</li> </ul>

ビルド番号	新機能	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-gl: 961</li> <li>nice-dcv-gltest: 291</li> <li>nice-dcv-simple-external-認証: 188</li> </ul>	<p>るブラウザ WebCodecs を利用するようになりました。</p> <ul style="list-style-type: none"> <li>Windows クライアントと Web クライアントに、高い色精度を有効にするオプションを追加しました。</li> <li>コラボレーションエクスペリエンスの向上: セッションに誰かが参加すると、ユーザーに通知が届くようになりました。</li> <li>サポート対象の Linux ディストリビューションのリストに CentOS 8 ストリームを追加しました。</li> </ul>	<p>でリッスンするように NICE DCV サーバーを構成できるようになりました。</p> <ul style="list-style-type: none"> <li>「DCV プリンター」は Linux システムでも自動的に設定されるようになりました。</li> <li>Windows の NICE DCV プロセスがより高い優先度で実行されるようになりました。</li> </ul> <p>修正内容:</p> <ul style="list-style-type: none"> <li>GPU を搭載するインスタンスを使用している場合に、Windows 2016 でエージェントを再起動するとクラッシュする問題を修正しました。</li> <li>一部の USB デバイスが NICE DCV クライアントからリダイレクトされている際にセッションからログアウトすると、Windows でクラッシュする問題を修正しました。</li> <li>認証チェックを実行する際に、Windows ドメインを含むユーザー名が正規化されます。</li> <li>Windows クライアントの相対マウスモードを改善しました。</li> </ul>

ビルド番号	新機能	変更とバグ修正
		<ul style="list-style-type: none"> <li>• CapsLock キーの同期に関する問題を修正しました。</li> </ul>

## DCV 2021.3-11591 — 2021 年 12 月 20 日

ビルド番号	新機能	変更とバグ修正
<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-認証: 176</li> </ul>	<p>NICE DCV に次の機能を追加しました。</p> <ul style="list-style-type: none"> <li>• Web クライアントのユーザーインターフェイスが更新されました。</li> <li>• EC2 G5 と G5G インスタンスがサポートされました。</li> <li>• Windows サーバー 2022 と Windows 11 がサポート対象のオペレーティングシステムになりました。</li> </ul>	<ul style="list-style-type: none"> <li>• Linux 仮想セッションの init スクリプトはユーザーの Bash プロファイルをロードしなくなったため、環境変数がシステムのデフォルト値を上書きする問題が繰り返し発生しなくなりました。</li> <li>• に Python 3 が必要 nice-dcv-ext-authenticator になりました。</li> </ul>

## DCV 2021.2-11445 — 2021 年 11 月 18 日

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>• nice-dcv-server: 11445</li> <li>• nice-dcv-client(Windows): 7792</li> <li>• nice-dcv-viewer (macOS ): 3797</li> <li>• nice-dcv-viewer (Linux): 3797</li> <li>• nice-xdcv: 411</li> <li>• nice-dcv-gl: 946</li> </ul>	<p>修正内容:</p> <ul style="list-style-type: none"> <li>• クライアントが macOS Monterey で正しく動作しない問題を修正しました。</li> <li>• Windows におけるサーバーのセキュリティが強化されました。</li> </ul>

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-glttest: 279</li> <li>nice-dcv-simple-external-認証: 160</li> </ul>	<ul style="list-style-type: none"> <li>特にウェブクライアントの使用時にマルチモニターレイアウトが正しく適用されないことがあるというバグを修正しました。</li> <li>Delete キーが一部の Windows アプリケーションで正しく動作しなくなるという問題を修正しました。</li> <li>Linux のウェブクライアントパッケージを、そのウェブクライアント自体が含まれていた旧バージョンのサーバーパッケージと相互に排他的なものとしてマークしました。</li> </ul>

## DCV 2021.2-11190 — 2021 年 10 月 11 日

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-server: 11190</li> <li>nice-dcv-client(Windows): 7788</li> <li>nice-dcv-viewer (macOS ): 3776</li> <li>nice-dcv-viewer (Linux): 3776</li> <li>nice-xdcv: 411</li> <li>nice-dcv-gl: 946</li> <li>nice-dcv-glttest: 279</li> <li>nice-dcv-simple-external-認証: 160</li> </ul>	<p>修正内容:</p> <ul style="list-style-type: none"> <li>証明書の期限が切れているサーバーに接続すると証明書の検証ダイアログを閉じることができなくなるという Windows クライアントの問題を修正しました。</li> <li>ネイティブクライアントでスタイラスペンのミドルクリックボタンが期待どおりに動作しないという問題を修正しました。</li> <li>レガシー X11 フォントが読み込まれなくなる Xdcv の回帰が修正されました。</li> <li>デッドキーを使用するキーボードレイアウトを使用しているときに、macOS クライアントおよび Linux クライアントでキーボードの組み合わせが正しく動作しない問題を修正しました。</li> </ul>

## DCV 2021.2-11135 — 2021 年 9 月 24 日

ビルド番号	変更とバグ修正
<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-gi: 944</li><li>• nice-dcv-glttest: 279</li><li>• nice-dcv-simple-external-認証: 160</li></ul>	<p>修正内容:</p> <ul style="list-style-type: none"><li>• 2021.2 クライアントを使用して古いサーバーに接続したときに接続問題とパフォーマンス問題を引き起こす可能性がある QUIC パケットサイズネゴシエーションの問題を修正しました。</li><li>• NVENC エンコーダ障害を引き起こす可能性がある NVIDIA デバイス選択のバグを修正しました。</li><li>• Windows と NVIDIA GPU を搭載したマシンで圧縮アーティファクトや色精度アーティファクトが発生する可能性があるという問題を修正しました。</li><li>• Linux サーバーの修飾キーにより一部のキーボードの組み合わせが期待どおりに動作しないことがあるというバグを修正しました。</li><li>• M1 CPU を搭載したマシンでの macOS クライアントのパフォーマンス低下を修正しました。</li><li>• macOS クライアントで一部のキーボードの組み合わせが期待どおりに動作しないというバグを修正しました。</li><li>• Linux 仮想セッションにおけるタッチイベントの処理によりセッションが終了されてしまう可能性があるという問題を修正しました。</li></ul>

## DCV 2021.2-11048 — 2021 年 9 月 1 日

ビルド番号	新機能	変更とバグ修正
<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> <li>• nice-dcv-gl: 944</li> <li>• nice-dcv-gltest: 279</li> <li>• nice-dcv-simple-external-<small>認</small>証: 160</li> </ul>	<p>NICE DCV に次の機能が追加されました。</p> <ul style="list-style-type: none"> <li>• ウェブクライアントのクリップボードの改善。これらの改善により、Google Chrome および Microsoft Edge で NICE DCV ウェブクライアントを使用して、PNG 形式のイメージのコピーと貼り付けを実行できるようになりました。</li> <li>• Windows クライアントと macOS クライアントのスクリーンショットブロック機能。この機能では、ユーザーが NICE DCV セッションコンテンツのスクリーンショットを撮影できないようにすることで、セキュリティレイヤーを追加します。有効にすると、ユーザーがキャプチャしたスクリーンショットがすべて空白の画面になります。</li> <li>• ストリーミングの質の強化。ストリーミング品質は、QUIC プロトコルの使用時に特に build-to-lossless 「」パフォーマンスを向上させることで向上しました。</li> </ul>	<p>変更:</p> <ul style="list-style-type: none"> <li>• NICE DCV ウェブクライアントは、Linux の個別のパッケージになり、かつ Windows インストーラのオプションコンポーネントになりました。この変更により、ウェブクライアントのデプロイをお客様が決定できます。</li> <li>• NVENC エンコーダの使用時に H.264 杯プロファイルがサポートされるようになりました。NVENC エンコーダーと NVIDIA GPU を使用すると、同じ画質を維持しながら帯域幅の使用量を減らすことができます。</li> <li>• NICE DCV サーバーでは、複数の GPU を搭載したマシンで使用可能なすべての GPU が圧縮に使用されるようになりました。</li> <li>• NICE DCV に同梱されている Windows ドライバーはすべて WHQL の認定を受けています。</li> <li>• OpenSSL がバージョン 1.1.1 に更新されました。</li> </ul>

ビルド番号	新機能	変更とバグ修正
	<ul style="list-style-type: none"> <li>• クライアントの動作を指定する <code>certificate-validation-policy</code> オプションが追加されました。これは、自己署名証明書などの信頼できない X.509 証明書がサーバーによって提示される場合に使用できます。</li> <li>• ランタイム時にオーディオドライバで設定されたチャンネル数は変更できません。</li> <li>• <code>dcvinput Xorg</code> モジュールに <code>Pressure2K</code> オプションが追加されました。これを使用すれば、スタイラスの圧力感度範囲を 0~65335 から 0~2048 に変更して、Mari や Nuke などのアプリケーションとの互換性を確保できます。</li> <li>• Google Chrome と Microsoft Edge での実験的な WebCodecs API のサポートが追加されました。ブラウザでこの API を有効にすると、NICE DCV ウェブクライアントでこの API が使用されてビデオデコードが高速化され、フレームレートがさらに高くなります。</li> </ul>	<ul style="list-style-type: none"> <li>• Xdcv が XServer のバージョン 1.20.13 に更新されました。</li> </ul> <p>修正内容:</p> <ul style="list-style-type: none"> <li>• macOS クライアントのテンキーに関する問題を修正しました。</li> <li>• 一部の USB デバイス (ゲームパッドなど) が Windows サーバーに正しくリダイレクトされない問題を修正しました。</li> <li>• 切断時に修飾キーを適切に解放できないバグを修正しました。</li> <li>• Ubuntu 20.04 と Intel GPU の使用時に Linux ネイティブクライアントでクラッシュが発生する問題を修正しました。</li> </ul>

## DCV 2021.1-10851 — 2021 年 7 月 30 日

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>• nice-dcv-server: 10851</li> <li>• nice-dcv-client(Windows): 7744</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-認証: 154</li> </ul>	<p>変更:</p> <ul style="list-style-type: none"> <li>• Windows クライアント、Linux クライアント、macOS クライアントの安定性を強化しました。</li> </ul> <p>修正内容:</p> <ul style="list-style-type: none"> <li>• Windows サーバーの AMD および NVIDIA グラフィックアダプターで画面がちらつくバグを修正しました。</li> <li>• 複数のセッションを実行している Linux サーバーに接続する際に散発する問題を修正しました。</li> <li>• Linux サーバーでの非西洋キーボードレイアウトの処理に関連するバグを修正しました。</li> <li>• Windows クライアントの接続ウィンドウに表示されるビジュアルアーティファクトが修正されました。</li> <li>• Windows の USB リダイレクトドライバーのいくつかのバグを修正し、デバイスの互換性を改善しました。</li> </ul>

## DCV 2021.1-10598 — 2021 年 6 月 10 日

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>• nice-dcv-server: 10598</li> <li>• nice-dcv-client(Windows): 7713</li> <li>• nice-dcv-viewer(macOS ): 3473</li> </ul>	<ul style="list-style-type: none"> <li>• サーバーの Windows インストーラーで session owner フィールドに現在のユーザーがあらかじめ入力されてしまう問題を修正しました。</li> </ul>

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-viewer(Linux): 3473</li> <li>nice-xdcv: 392</li> <li>nice-dcv-gi: 937</li> <li>nice-dcv-gitest: 275</li> <li>nice-dcv-simple-external-認証: 154</li> </ul>	<ul style="list-style-type: none"> <li>macOS クライアントと Linux クライアントの全体的な安定性を改善しました。</li> </ul>

## DCV 2021.1-10557 — 2021 年 5 月 31 日

ビルド番号	新機能	変更とバグ修正
<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-gi: 937</li> <li>nice-dcv-gitest: 275</li> <li>nice-dcv-simple-external-認証: 154</li> </ul>	<ul style="list-style-type: none"> <li>NICE DCV で、GPU があるサーバーに接続する時に高精度のオーディオ/ビデオの同期を可能にするクライアントオプションを追加しました。</li> <li>NICE DCV で、Linux コンソールセッションにおけるマイクのサポートを追加しました。</li> </ul>	<ul style="list-style-type: none"> <li>GPU がない Windows サーバーホストでの CPU 使用率を低下させました。</li> <li>macOS クライアントと Linux クライアント内の .dcv 接続ファイルの読み取り時に発生する問題を修正しました。</li> <li>ハードウェアアクセラレーションデコードに対応していない macOS マシンのソフトウェアデコードにフォールバックを追加しました。</li> <li>システムキーチェーンに保存されている CA 証明書を macOS クライアントで読み取るためのサポートを追加しました。</li> </ul>

## DCV 2021.0-10242 — 2021 年 4 月 12 日

ビルド番号	新機能	変更とバグ修正
<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> <li>• nice-dcv-gltest: 266</li> <li>• nice-dcv-simple-external-<small>認</small>証: 134</li> </ul>	<ul style="list-style-type: none"> <li>• Windows NICE DCV サーバーのウェブカメラリダイレクトのサポートを追加しました。</li> <li>• Linux NICE DCV サーバーのプリンターリダイレクトのサポートを追加しました。</li> <li>• macOS クライアントでの M1 プロセッサのサポートを追加しました。</li> <li>• macOS クライアントのマルチモニターディスプレイのサポートを追加しました。</li> </ul>	<ul style="list-style-type: none"> <li>• Linux サーバー、および NVIDIA GPU を使用した Amazon EC2 インスタンスで、GPU と CPU リソースの使用率を最適化しました。</li> <li>• Linux NICE DCV サーバー用の Amazon EC2 G4ad インスタンスで AMD GPU を使用した GPU アクセラレーションビデオエンコーディングのサポートを追加しました。</li> <li>• オーディオレイテンシーを低減するためにオーディオ処理を最適化しました。</li> <li>• QUIC プロトコルがサーバーで有効になっている場合のクライアントのデフォルトを QUIC プロトコルに変更しました。</li> <li>• DCV コマンドラインツールに新しい get-screenshot コマンドを追加しました。</li> <li>• close-session コマンドの --logout-user オプションを使用する強制ログアウトオプションを追加しました。このオプションは、コンソールセッション</li> </ul>

ビルド番号	新機能	変更とバグ修正
		を閉じるときに使用できません。

## DCV 2020.2-9662 — 2020 年 12 月 4 日

ビルド番号	変更とバグ修正
<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-gltest: 259</li> <li>nice-dcv-simple-external-認証: 125</li> </ul>	<ul style="list-style-type: none"> <li>ウェブブラウザクライアントで使用されるセキュリティプロトコルを強化しました。</li> <li>Windows クライアントで使用される Amazon EC2 G4ad インスタンスのパフォーマンスと堅牢性を向上させました。</li> <li>Windows クライアントの接続設定ダイアログでのポートの選択に関する問題を修正しました。</li> </ul>

## DCV 2020.2-9508 — 2020 年 11 月 11 日

ビルド番号	新機能	変更とバグ修正
<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-gltest: 259</li> </ul>	<ul style="list-style-type: none"> <li>QUIC (UDP ベース) トランスポートプロトコルのサポートを追加しました。</li> <li>SLES 15 と Ubuntu 20.4 のサポートが追加されました。</li> <li>Windows NICE DCV サーバーのスマートカードのサポートを追加しました。</li> </ul>	<ul style="list-style-type: none"> <li>NVIDIA GPU があるサーバーと EC2 インスタンスでホストされるコンソールセッションで、デフォルトの NICE DCV フレームレイトリミッターを 60 FPS に変更しました。</li> <li>NVIDIA GPU がある EC2 インスタンスでホストされている Windows NICE DCV サーバーで使用される GPU</li> </ul>

ビルド番号	新機能	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-simple-external-認証: 125</li> </ul>		<p>および CPU リソースを最適化しました。</p> <ul style="list-style-type: none"> <li>list-endpoints NICE DCV CLI コマンドを追加しました。これには現在アクティブなエンドポイントが記載されています。</li> <li>version NICE DCV CLI コマンドは --json オプションに対応しています。</li> <li>Linux サーバーにおいて、create-session NICE DCV CLI コマンドが --disable-login-monitor オプションに対応するようになりました。</li> <li>Linux NICE DCV サーバーのさまざまなディスプレイマネージャーとの互換性を向上させました。</li> <li>キーボード入力の処理に関するいくつかの問題を修正しました。</li> <li>USB デバイスの許可リストファイルが動的に再ロードされるようになりました。</li> </ul>

## DCV 2020.1-9012 — 2020 年 9 月 30 日

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-server: 9012</li> </ul>	<ul style="list-style-type: none"> <li>欠落している macOS クライアントアイコンを追加しました。</li> </ul>

ビルド番号	変更とバグ修正
<ul style="list-style-type: none"> <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-external-認証: 111</li> </ul>	

## DCV 2020.1-9012 — 2020 年 8 月 24 日

ビルド番号	変更とバグ修正
<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-external-認証: 111</li> </ul>	<ul style="list-style-type: none"> <li>AWS GovCloud リージョンでの Amazon S3 アクセスを修正しました</li> <li>ウェブベースのクライアントの改善</li> </ul>

## DCV 2020.1-8942 — 2020 年 8 月 3 日

ビルド番号	新機能	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-server: 8942</li> <li>nice-dcv-client (Windows): 7342</li> <li>nice-dcv-viewer (macOS ): 1910</li> </ul>	<ul style="list-style-type: none"> <li>Linux NICE DCV サーバーは、M6g, C6g AWS Graviton2-based Arm インスタンスをサポートするようになりました。R6g 詳細</li> </ul>	<ul style="list-style-type: none"> <li>GPU がない Amazon EC2 インスタンスにおいて新しい NICE DCV 仮想ディスクレイドライバのサポートを追加しました。</li> </ul>

ビルド番号	新機能	変更とバグ修正
<ul style="list-style-type: none"> <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-external-認証: 111</li> </ul>	<p>については、「<a href="#">AWS Graviton プロセッサ</a>」を参照してください。</p> <ul style="list-style-type: none"> <li>• Linux NICE DCV サーバーでの RHEL 8.x と CentOS 8.x のサポートを追加しました。</li> <li>• Windows NICE DCV サーバーと Windows NICE DCV クライアントを使用する場合のプリンターリダイレクトのサポートを追加しました。</li> <li>• macOS と Linux のネイティブ NICE DCV クライアントでの圧力感度のスタイラスサポートを追加しました。</li> <li>• Linux NICE DCV サーバーと Linux NICE DCV クライアントのサラウンドサウンド 5.1 サポートを追加しました。</li> <li>• Linux NICE DCV ネイティブクライアントのタッチスクリーンサポートを追加しました。</li> <li>• カスタム名を NICE DCV セッションに関連付けることができるようになりました。</li> <li>• macOS ネイティブ NICE DCV クライアントにおけるハードウェアアクセラレー</li> </ul>	<ul style="list-style-type: none"> <li>• NVENC エンコーダーの使用時にカラースペース変換の結果として視覚アーティファクトが発生する問題を修正しました。</li> <li>• コンソールセッションが存在する場合は常に <code>dcv list-sessions</code> コマンドに含まれるようになりました。</li> <li>• 新しい Linux ディストリビューションでは、コンソールセッションのエージェントがデスクトップセッションの一部として開始されるようになり、GDM 3 などの新しいディスプレイマネージャーのサポートが強化されました。</li> <li>• <code>dcv://</code> スキームを含む URL をアクティブ化すると、ネイティブクライアントが自動的に開くようになりました。</li> <li>• macOS ネイティブクライアントとウェブクライアントでのキーボード修飾子の処理方法を改善しました。</li> <li>• DCV-GL での視覚要素と <code>fbconfig</code> の選択を改善し、一部のアプリケーションのサポートを改善しました。</li> <li>• ファイル転送中の CPU 使用率を低下させました。</li> </ul>

ビルド番号	新機能	変更とバグ修正
	シヨンデコードおよびレンダリングのサポート。	<ul style="list-style-type: none"> <li>ウェブブラウザクライアントでの WebGL レンダリングを改善し、リソースの使用量を削減しました。</li> </ul>

## DCV 2020.0-8428 — 2020 年 4 月 16 日

ビルド番号	新機能	変更とバグ修正
<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-external-認証: 87</li> </ul>	<ul style="list-style-type: none"> <li>Linux サーバーでの画面上のスタイラスとタッチ操作に関するサポートを追加しました。</li> <li>Windows サーバーの 7.1 サラウンドサウンド再生サポートを Windows ネイティブクライアントに追加しました。</li> <li>Linux ネイティブクライアントのハードウェアアクセラレーションとスタイラスのサポートを追加しました。</li> <li>サーバー側で表示レイアウトを設定するための新しい API コマンドを追加しました。</li> <li>Microsoft Edge ブラウザ (バージョン 79.0.309 以降) のマルチモニターウェブクライアントのディスプレイサポートを追加しました。</li> </ul>	<ul style="list-style-type: none"> <li>Windows クライアントのツールバーグリップを、全画面表示モードで非表示にできるようになりました。</li> <li>Windows ネイティブクライアントでの NTLM プロキシサポートを追加しました。</li> <li>NVIDIA アダプタを使用した Windows ヘッドレス物理ホストのサポートが向上しました。</li> <li>レガシー NVIDIA NVIFR ライブラリのサポートを削除しました。</li> <li>最新の Windows 10 の Windows グラフィックキャプチャ API 向けのサポートが追加されました。</li> <li>Amazon EC2 のサポートを追加しました。</li> </ul> <p>EC2 インスタンス上のインスタンスメタデータサービス (IMDS) v2。</p>

ビルド番号	新機能	変更とバグ修正
		<ul style="list-style-type: none"> <li>• DCV CLIには、クライアントがセッションに接続または切断するタイミングを検出する新しい <code>on-client-connected / disconnected</code> コマンドが用意されています。</li> <li>• 外部認証システムの証明書をバインドするホスト名を指定するためのサポートが追加されました。</li> <li>• DCV-GL は、このライブラリをサポートするシステムで GL Vendor-Neutral Dispatch ライブラリ (GLvnd) を使用するようになりました。</li> </ul>

## DCV 2019.1-7644 — 2019 年 10 月 24 日

ビルド番号	変更とバグ修正
<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-external-認証: 77</li> </ul>	<ul style="list-style-type: none"> <li>• NICE EnginFrame およびその他のセッションマネージャーが使用する統合 API の問題を修正しました。</li> <li>• 32 ビット版の Windows ネイティブクライアントの問題が修正されました。</li> </ul>

## DCV 2019.1-7423 — 2019 年 9 月 10 日

ビルド番号	変更とバグ修正
<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-external-認証: 77</li> </ul>	<ul style="list-style-type: none"> <li>Windows 上の DCV サーバーのセキュリティが強化されました。</li> <li>Linux 上の Autodesk Maya のレンダリングに関する問題が修正されました。</li> <li>キーボードの操作に関する改善とバグ修正が追加されました。</li> </ul>

## DCV 2019.0-7318 — 2019 年 8 月 5 日

ビルド番号	新機能	変更とバグ修正
<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> <li>nice-xdcv: 224</li> <li>nice-dcv-gl: 529</li> <li>nice-dcv-gltest: 218</li> <li>nice-dcv-simple-external-認証: 72</li> </ul>	<ul style="list-style-type: none"> <li>ウェブクライアントでのマルチモニターをサポート。</li> <li>Windows Server 2019 でのスタイラス入力をサポート。</li> <li>macOS および Linux のネイティブクライアントでのオーディオ入出力。</li> <li>Linux サーバーでのクリップボード機能の拡張 (ミドルクリックで貼り付け)。</li> </ul>	<ul style="list-style-type: none"> <li>Windows タッチ入力の圧力感度の互換性を強化して追加しました。</li> <li>Windows で異種グラフィックアダプタを使用するシステムでの動作が向上しました。</li> <li>非アクティブな接続の検出に必要な時間の短縮 (クライアントでの有線ネットワークから Wi-Fi ネットワークへの変更時の対応など)。</li> <li>Linux でカーソルアイコンをキャプチャできない場合のログ記録を削減しました。</li> </ul>

ビルド番号	新機能	変更とバグ修正
		<ul style="list-style-type: none"><li>• 仮想セッション Xdcv コンポーネントで Composite 拡張を無効にするサポート。</li><li>• 同時仮想セッション数の制限にオプションを追加しました。</li><li>• Bash 5 がインストールされているシステムとのスクリプト互換性を改善しました。</li><li>• OpenGL と GLES が検出されて Linux クライアントでのレンダリングに自動的に使用されるようにデフォルトを変更しました。</li><li>• GL ウィンドウの表示が変更されたときの DCV-GL 画面のバッファを更新しました。</li><li>• Windows 7 の Windows クライアントでのマウスホイールの検出を修正しました。</li><li>• Windows クライアントが一部の Windows 7 システムでライブラリのロードに失敗する問題が修正されました。</li><li>• ドキュメントを横向きで印刷するときの Windows クライアントでの印刷が改善されました。</li></ul>

## DCV 2017.4-6898 — 2019 年 4 月 16 日

ビルド番号	新機能	変更とバグ修正
<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-gi: 490</li> <li>• nice-dcv-glttest: 216</li> <li>• nice-dcv-simple-external-認証: 70</li> </ul>	<ul style="list-style-type: none"> <li>• macOS 用の新しいネイティブクライアント。</li> </ul>	<ul style="list-style-type: none"> <li>• Windows ネイティブクライアントは、システムで使用可能な場合、デコードとレンダリングにハードウェアアクセラレーションを使用するようになりました。</li> <li>• dcv コマンドラインツールは、Windows と Linux の両方で同じオプションと出力形式を使用するようになりました。</li> <li>• dcv コマンドラインツールは、ライセンスに関する情報をレポートするようになりました。</li> <li>• クライアントは、非アクティブにより切断する前にユーザーに警告を表示するようになりました。</li> <li>• 複数の修飾子を使用するキーボードの組み合わせのサポートが向上しました。</li> <li>• 通信障害に対する Reprise ライセンスマネージャーとの対話の堅牢性が向上しました。</li> <li>• dcvusers コマンドラインツールでは、既定で Linux 上の dcv ユーザーのホームディレクトリにデータを保存するようになりました。</li> </ul>

ビルド番号	新機能	変更とバグ修正
		<ul style="list-style-type: none"> <li>Linux 上で複数の GPU を搭載した NVENC ハードウェアエンコーダーを使用する場合、nvidia-smi ツールを使用する場合と同じ順序になりました。</li> <li>Linux クライアントは、Windows DCV プリンターから出力されたファイルを受信して処理するようになりました。</li> </ul>

## DCV 2017.3-6698 — 2019 年 2 月 24 日

ビルド番号	新機能	変更とバグ修正
<ul style="list-style-type: none"> <li>nice-dcv-server: 6698</li> <li>nice-dcv-client: 5946</li> <li>nice-dcv-viewer (Linux): 683</li> <li>nice-xdcv: 207</li> <li>nice-dcv-gl: 471</li> <li>nice-dcv-gltest: 210</li> <li>nice-dcv-simple-external-認証者: 66</li> </ul>	<ul style="list-style-type: none"> <li>Kerberos (GSSAPI) 認証のサポートが追加されました。</li> <li>サポートされている Windows 版で、タッチイベントのサポートが追加されました。</li> <li>システム認証 (Windows 資格情報プロバイダー) を使用しているときに、Windows セッションを自動的にロック解除します。</li> </ul>	<ul style="list-style-type: none"> <li>YUV444 エンコーディングにオプトインするオプションが追加されました。</li> <li>EL6 RPM には NVENC エンコーダーモジュールが搭載されました。</li> <li>Windows システム認証で name@domain 形式が受容されるようになりました。</li> <li>Yubikey USB デバイスが許可リストに追加されました。</li> <li>日本語キーボードのサポートが向上しました。</li> <li>入力の認証許可はよりきめ細かくなりました。仮想カーソルを処理する</li> </ul>

ビルド番号	新機能	変更とバグ修正
		<p>pointer の許可が追加されました。相対マウスモードは、マウス (モーションインジェクション用) とポインタ (モーションフィードバック用) によって異なります。Windows 上で SAS を扱う keyboard-sas の許可が追加されました (Control+Alt+Del)。keyboardsas は keyboard の許可に依存します。</p> <ul style="list-style-type: none"><li>• 非同期クリップボード API をサポートするブラウザのウェブクライアントで、空のクリップボードイベントに関する問題が修正されました。</li><li>• クライアントが最初のフレームを受信できなくなっていたキャプチャモジュールの競合が修正されました。</li><li>• 同時ファイルストレージ転送の処理が向上しました。</li><li>• 新しい NVIDIA ドライバーを搭載した Windows 上の NVIFR が修正されました。新しいドライバーの動作が変更されました。ドライバーのバージョンが自動的に検出され、それに応じてメモリ処理が実行されません。</li></ul>

ビルド番号	新機能	変更とバグ修正
		<ul style="list-style-type: none"><li>• RLM ライセンストークンの再取得を中止せずに継続します。これにより、長期間経過しても licensing error 状態から回復できます。</li><li>• Windows クライアントで全画面表示のキーボードショートカットを設定するオプションを追加しました。</li><li>• Windows クライアントで複数のモニタ間でウィンドウをドラッグするときの自動調整ロジックを改善しました。</li><li>• Windows クライアントで Ulin により切断がトリガーされない場合のプロンプト再接続オプションを修正しました。</li><li>• NVIDIA ドライバ 410.xx と DCV-GL の非互換性が修正されました。</li><li>• Matlab および Blender アプリケーションでの DCV-GL のパフォーマンス低下が修正されました。</li></ul>

## DCV 2017.2-6182 — 2018 年 10 月 8 日

ビルド番号	新機能	変更とバグ修正
<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-external-認証: 58</li> </ul>	<ul style="list-style-type: none"> <li>• Linux 仮想セッションでオーディオ再生のサポートが追加されました。</li> <li>• スマートカードのパフォーマンスが向上しました。</li> <li>• Linux クライアントでファイル転送のサポートが追加されました。</li> </ul>	<ul style="list-style-type: none"> <li>• キーボードの処理に関する改善とバグ修正。</li> <li>• ログレベルの設定を変更しても、サーバーを再起動する必要がなくなりました。</li> <li>• Windows サーバーインストーラは、既にインストールされている場合、Microsoft C ランタイム再頒布可能パッケージのインストールをスキップするようになりました。</li> <li>• EC2 で実行しているときに、ライセンスの S3 へのアクセスに失敗すると、ユーザーインターフェイスに通知が表示されます。</li> <li>• Linux dcv コマンドラインツールでは、list-connections および describe-session サブコマンドがサポートされるようになり、JSON 出力を実行するオプションが追加されました。</li> <li>• display セクションの cuda-devices 設定を追加しました。これにより、さまざまな CUDA デバイスに NVENC エンコーディ</li> </ul>

ビルド番号	新機能	変更とバグ修正
		<p>ングを配信するようにサーバーが設定されます。</p> <ul style="list-style-type: none"> <li>• 複数の同時コマンドを処理する場合の、セッション作成コードの堅牢性が向上しました。</li> <li>• デフォルトのクリップボードの上限を 20 MB に引き上げました。</li> <li>• Windows クライアントはレガシー .dvc ファイルを検出し、DCV 2016 Endstation を起動するようになりました (インストールされている場合)。</li> <li>• DCV のシンプルな外部認証システムは、環境で設定されているものではなく、常にシステム Python インタプリタを使用するようになりました。</li> <li>• DCV-GL のリードバック方式が改善され、パフォーマンスと堅牢性が向上しました。</li> <li>• DCV-GL は、フロントバッファのリードバック後にウィンドウのサイズが変更されたかどうかをチェックするようになりました。これにより、Coot アプリケーションのレンダリングの問題が修正されます。</li> </ul>

## DCV 2017.1-5870 — 2018 年 8 月 6 日

ビルド番号	新機能	変更とバグ修正
<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-gi: 366</li> <li>nice-dcv-gitest: 198</li> <li>nice-dcv-simple-external-認証: 53</li> </ul>	<p>Ubuntu 18.04 用のパッケージをリリースしました。コンソールモードで作業する場合、LightDM または他の任意のディスプレイマネージャーを使用するようにシステムを設定する必要があります。GDM は必要な X11 ディスプレイ情報を公開していないためです。仮想セッションはこの制限の影響を受けません。</p>	<ul style="list-style-type: none"> <li>セッションの作成時ライセンス設定が読み取られるようになりました。これにより、管理者はサーバーを再起動せずにこの設定を変更できます。</li> <li>一部のシステムでプログラムが予期せず終了する Windows クライアントの安定性の問題が解決されました。</li> <li>エラーの可能性のある状態でのログ記録が削減されました。</li> </ul>

## DCV 2017.1-5777 — 2018 年 6 月 29 日

ビルド番号	新機能	変更とバグ修正
<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-gi: 366</li> <li>nice-dcv-gitest: 189</li> <li>nice-dcv-simple-external-認証: 51</li> </ul>	<ul style="list-style-type: none"> <li>Linux ネイティブクライアントが追加されました。</li> <li>3DConnexion マウスおよび USB ストレージデバイスのサポートが追加されました。</li> <li>最後のクライアントが切断されると、Windows セッションが自動的にロックされます。</li> </ul>	<ul style="list-style-type: none"> <li>Linux 版でのパフォーマンスの向上。</li> <li>新しい NVIDIA ドライバーの NVIFR に関する問題を回避するため、NVIDIA デバイスのデフォルトのハードウェアエンコーダーは NVENC に変更されました。</li> <li>Linux でのスマートカードのサポートが向上しました。</li> </ul>

ビルド番号	新機能	変更とバグ修正
		<ul style="list-style-type: none"> <li>Linux コンソールセッション使用時のアップロードファイルに対するファイルアクセス許可が修正されました。</li> </ul>

## DCV 2017.0-5600 — 2018 年 6 月 4 日

ビルド番号	新機能	変更とバグ修正
<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-gi: 279</li> <li>nice-dcv-glttest: 184</li> <li>nice-dcv-simple-external-認証: 48</li> </ul>	<ul style="list-style-type: none"> <li>Linux でのマルチモニターサポートが追加されました。</li> <li>Windows クライアントのパフォーマンスが向上しました。</li> <li>Chrome 66 以降で新しいクリップボード API が使用されます。</li> <li>Windows 用 NVENC エンコーダーが追加されました。</li> </ul>	<ul style="list-style-type: none"> <li>EC2 での使用には、DCV サーバーを実行しているインスタンスから S3 にアクセスできる機能が必要です。</li> <li>サーバーフレーム処理と Windows クライアントデコードのパフォーマンスが向上しました。</li> <li>NumPad およびブロックされた修飾子に関連するキーボードの問題を修正しました。</li> <li>Linux で外部認証システムを使用している場合に、ファイル記述子のリークを防止します。</li> <li>スマートカード接続で発生する可能性があったエラーが修正されました。</li> </ul>

## DCV 2017.0-5121 — 2018 年 3 月 18 日

ビルド番号	新機能	変更とバグ修正
<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-external-認証者: 46</li> </ul>	<ul style="list-style-type: none"> <li>Windows ネイティブクライアントが DPI 対応になりました。</li> <li>相対マウス移動モードのサポートが追加されました。</li> </ul>	<ul style="list-style-type: none"> <li>Linux の Ansys cfx5solve でのハングを防止しました。</li> <li>Windows 10 で発生する可能性があったエージェントがハングする問題が修正されました。</li> <li>ウェブクライアントのユーザーインターフェイスを改良しました。</li> <li>ドメインが指定されている場合、Windows ユーザー名は正規化されます。</li> <li>RHEL6 の外部認証システムが修正されました。</li> </ul>

## DCV 2017.0-4334 — 2018 年 1 月 24 日

ビルド番号	変更とバグ修正
<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-external-認証: 45</li> </ul>	<ul style="list-style-type: none"> <li>キーボードの処理が向上しました。</li> <li>RHEL6 で、セッションを閉じたときに新しいセッションを作成できない DBus 問題を修正しました。</li> <li>ネイティブクライアントでの SOCKS5 プロキシのサポートが向上しました。</li> <li>仮想セッションで実行中の Headwave でのクラッシュと、仮想セッションで実行中の Chimera でのクラッシュを引き起こすバグを修正しました。</li> </ul>

ビルド番号	変更とバグ修正
	<ul style="list-style-type: none"> <li>仮想セッションでのフォントサポートの向上。</li> </ul>

DCV 2017.0-4100 — 2017 年 12 月 18 日

ビルド番号
<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> <li>nice-dcv-simple-external-認証: 35</li> </ul>

## ドキュメント履歴

次の表は、NICE DCV の今回のリリースの内容をまとめたものです。

変更	説明	日付
NICE DCV バージョン 2023.1	Window、macOS、Linux Client 2023.1 に対する NICE DCV の修正。詳細については、「 <a href="#">DCV 2023.1-16388 — 2024 年 7 月 3 日</a> 」を参照してください。	2024 年 7 月 3 日
NICE DCV バージョン 2023.1	Windows クライアント 2023.1 に対する NICE DCV の修正。詳細については、「 <a href="#">DCV 2023.1-16388 — 2024 年 3 月 5 日</a> 」を参照してください。	2024 年 3 月 5 日

変更	説明	日付
NICE DCV バージョン 2023.1	NICE DCV の 2023.1 での修正。詳細については、「 <a href="#">DCV 2023.1-16388 – 2023 年 12 月 19 日</a> 」を参照してください。	2023 年 12 月 19 日
NICE DCV バージョン 2023.1	NICE DCV 2023.1 が利用可能になりました。詳細については、「 <a href="#">DCV 2023.1-16220 — 2023 年 11 月 9 日</a> 」を参照してください。	2023 年 11 月 9 日
NICE DCV バージョン 2023.0	NICE DCV は、サポートが終了したオペレーティングシステムをサポートしなくなりました。	2023 年 6 月 30 日
NICE DCV バージョン 2023.0	NICE DCV の 2023.0 での修正。詳細については、「 <a href="#">DCV 2023.0-15487 — 2023 年 6 月 29 日</a> 」を参照してください。	2023 年 6 月 29 日
NICE DCV バージョン 2023.0	NICE DCV の 2023.0 での修正。詳細については、「 <a href="#">DCV 2023.0-15065 — 2023 年 5 月 3 日</a> 」を参照してください。	2023 年 5 月 3 日
NICE DCV バージョン 2023.0	NICE DCV の 2023.0 での更新と修正。詳細については、「 <a href="#">DCV 2023.0-15022 — 2023 年 4 月 21 日</a> 」を参照してください。	2023 年 4 月 21 日

変更	説明	日付
NICE DCV バージョン 2023.0	NICE DCV 2023.0 が利用可能になりました。詳細については、「 <a href="#">DCV 2023.0-14852 — 2023 年 3 月 28 日</a> 」を参照してください。	2023 年 3 月 28 日
NICE DCV バージョン 2022.2	NICE DCV 2022.2 が利用可能になりました。詳細については、「 <a href="#">DCV 2022.2-13907 — 2022 年 11 月 11 日</a> 」を参照してください。	2022 年 11 月 11 日
NICE DCV バージョン 2022.1	NICE DCV 2022.1 が利用可能になりました。詳細については、「 <a href="#">DCV 2022.1-13067 — 2022 年 6 月 29 日</a> 」を参照してください。	2022 年 1 月 29 日
NICE DCV バージョン 2022.0	NICE DCV 2022.0 が利用可能になりました。詳細については、「 <a href="#">DCV 2022.0-11954 — 2022 年 2 月 23 日</a> 」を参照してください。	2022 年 2 月 23 日
NICE DCV バージョン 2021.3	NICE DCV 2021.3 が利用可能になりました。詳細については、「 <a href="#">DCV 2021.3-11591 — 2021 年 12 月 20 日</a> 」を参照してください。	2021 年 12 月 20 日
NICE DCV バージョン 2021.2	NICE DCV 2021.2 が利用可能になりました。詳細については、「 <a href="#">DCV 2021.2-11048 — 2021 年 9 月 1 日</a> 」を参照してください。	2021 年 9 月 1 日

変更	説明	日付
NICE DCV バージョン 2021.1	NICE DCV 2021.1 が利用可能になりました。詳細については、「 <a href="#">DCV 2021.1-10557 — 2021 年 5 月 31 日</a> 」を参照してください。	2021 年 5 月 31 日
NICE DCV バージョン 2021.0	NICE DCV 2021.0 が利用可能になりました。詳細については、「 <a href="#">DCV 2021.0-10242 — 2021 年 4 月 12 日</a> 」を参照してください。	2021 年 4 月 12 日
NICE DCV ウェブクライアント SDK	NICE DCV ウェブクライアント SDK が利用可能になりました。NICE DCV ウェブクライアント SDK は、エンドユーザーが実行中の NICE DCV セッションに接続して操作するために使用できる独自の NICE DCV ウェブブラウザクライアントアプリケーションを開発するために使用できる JavaScript ライブラリです。詳細については、「 <a href="#">NICE DCV ウェブクライアント SDK デベロッパーガイド</a> 」を参照してください。	2021 年 3 月 24 日
NICE DCV バージョン 2020.2	NICE DCV 2020.2 が利用可能になりました。詳細については、「 <a href="#">DCV 2020.2-9508 — 2020 年 11 月 11 日</a> 」を参照してください。	2020 年 11 月 11 日

変更	説明	日付
NICE DCV バージョン 2020.1	NICE DCV 2020.1 が利用可能になりました。詳細については、「 <a href="#">DCV 2020.1-8942 — 2020 年 8 月 3 日</a> 」を参照してください。	2020 年 8 月 3 日
NICE DCV バージョン 2020.0	NICE DCV 2020.0 には、サウンド7.1、タッチ操作とスタイラス、新しい Microsoft Edge ブラウザを使用したマルチモニターのサポートが含まれています。詳細については、「NICE DCV 管理者ガイド」の「 <a href="#">NICE DCV サーバーのインストール</a> 」を参照してください。	2020 年 4 月 16 日
HTTP レスポンスヘッダー	NICE DCV では、追加の HTTP レスポンスヘッダーを送信するように設定できません。	2019 年 8 月 26 日
macOS クライアント	NICE DCV で macOS クライアントを使用できるようになりました。詳細については、「NICE DCV ユーザーガイド」の「 <a href="#">macOS クライアント</a> 」を参照してください。	2019 年 4 月 18 日

変更	説明	日付
スマートカードのキャッシュ	NICE DCV サーバーでは、クライアントから受け取ったスマートカードデータをキャッシュして、パフォーマンスを向上させられるようになりました。詳細については、「NICE DCV 管理者ガイド」の「 <a href="#">スマートカードキャッシュの設定</a> 」を参照してください。	2018 年 10 月 8 日
Linux クライアント	NICE DCV には、RHEL 7、CentOS 7、SLES 12、および Ubuntu 16.04/18.04 向けの Linux クライアントがあります。詳細については、「 <a href="#">NICE DCV ユーザーガイド</a> 」の「Linux クライアント」を参照してください。	2018 年 8 月 29 日
パラメータリファレンスを更新	パラメータのリファレンスが更新されました。詳細については、「NICE DCV 管理者ガイド」の「 <a href="#">NICE DCV サーバーパラメータリファレンス</a> 」を参照してください。	2018 年 8 月 7 日
USB リモート化	NICE DCV では、クライアントで特殊な USB デバイス (例: 3D ポインティングデバイス、グラフィックタブレット) を使用できます。詳細については、「NICE DCV 管理者ガイド」の「 <a href="#">USB リモート化の有効化</a> 」を参照してください。	2018 年 8 月 7 日

変更	説明	日付
NICE DCV の初回リリース	このコンテンツの初版です。	2018 年 6 月 05 日

翻訳は機械翻訳により提供されています。提供された翻訳内容と英語版の間で齟齬、不一致または矛盾がある場合、英語版が優先します。