管理者ガイド

Amazon DCV

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon DCV: 管理者ガイド

Copyright © 2025 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

Amazon VPC とは 1
- Amazon DCV の仕組み 1
- 機能 1
- 料金 3

Amazon DCV サーバーについて 4
- 要件 4
- サポートされている機能 7

Amazon DCV サーバーのセットアップ 9
- ステップ 1: Amazon DCV サーバーをインストールする 9
  - Windows でのインストール 9
  - Linux でのインストール 17
- ステップ 2: Amazon DCV サーバーのライセンスを取得する 61
  - Amazon DCV ライセンス要件 61
  - 拡張評価ライセンスのインストール 65
  - 実稼働ライセンスのインストール 66
  - 実稼働ライセンスの更新 79
- ステップ 3: Amazon DCV サーバーのイメージングを設定する (オプション) 79
  - イメージの構築 80
  - イメージパイプラインに追加する 80

Amazon DCV サーバーの管理 82
- サーバーの起動 83
- サーバーの停止 84
- サーバーのアップグレード 85
  - 互換性に関する考慮事項 86
  - Windows での Amazon DCV サーバーのアップグレード 86
  - Linux での Amazon DCV サーバーのアップグレード 87
- サーバーをアンインストールする 87
  - Windows での Amazon DCV サーバーのアンインストール 87
  - Linux での Amazon DCV サーバーのアンインストール 88
- QUIC UDP を無効にする 89
- TCP/UDP ポートとアドレスの変更 90
  - サーバーの TCP/UDP ポートの変更 91
  - 特定のエンドポイントのリスニング 94
- TLS 証明書の管理 96

アイドル状態のクライアントの切断 97
Linux での GPU 共有の有効化 100
タッチスクリーンとスタイラスのサポートの有効化 102
スタイラスの設定 104
ゲームパッドサポートの有効化 105
Xbox 360 コントローラーのサポート 106
USB リモート化の有効化 106
スマートカードのキャッシュの設定 108
WebAuthn リダイレクトの設定 110
WebAuthn リダイレクトのブラウザ拡張機能の設定 111
セッションストレージの有効化 113
Windows でのセッションストレージの有効化 113
Linux でのセッションストレージの有効化 114
Linux でのプリンターの設定 115
プリンター問題のトラブルシューティング 117
Linux でのクリップボードの設定 117
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 118
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 119
マルチチャンネルオーディオの設定 119
Windows Amazon DCV サーバーでのオーディオチャンネルの設定 121
Linux Amazon DCV サーバーでのオーディオチャンネルの設定 121
HTTP ヘッダーを設定する 123
Windows Amazon DCV サーバーでの HTTP ヘッダーの設定 123
Linux Amazon DCV サーバーでの HTTP ヘッダーの設定 124
認証の設定 124
Windows での認証の設定 125
Linux での認証の設定 127
外部認証システムによる認証の設定 128
外部認証を使用する 128
認可の設定 132
デフォルトのアクセス許可ファイル 133
カスタムアクセス許可ファイル 133
アクセス許可ファイルについて 133
仮想セッションで X Server へのリモート X 接続を有効にする 139
X Server へのリモート X 接続を有効にする 140
Amazon DCV ウェブブラウザクライアントを iFrame 内に埋め込む 141

Amazon DCV セッションの管理 144
Amazon DCV セッションについて 144
コンソールセッション 145
仮想セッション 146
コマンドラインツールの使用 146
Windows Amazon DCV サーバーでのコマンドラインツールの使用 147
Linux Amazon DCV サーバーでのコマンドラインツールの使用 147
コマンドラインツールの使用 147
セッションの開始 149
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 150
自動コンソールセッションの有効化 155
セッションの停止 156
Syntax 157
例 157
セッションの表示 157
すべてのアクティブなセッションの表示 158
特定のアクティブなセッションの表示 158
アクティブなセッションの管理 160
セッションストレージの管理 160
セッション認可の管理 162
セッション表示レイアウトの管理 163
セッション名の管理 166
アイドル状態のセッションの検出と停止 168
セッションのタイムゾーンの設定 169
Linux でのスクリーンブランキングの管理 170
スクリーンショットの作成 171
構文 171
オプション 171
例 173
トラブルシューティング 175
ログファイルの使用 175
ログファイルの詳細度の変更 176
Linux での仮想セッション作成のトラブルシューティング 179
Linux での仮想セッション作成失敗の調査 179
Linux でのフェイルセーフ仮想セッションの作成 180
UID を変更すると Linux セッションが開始されない 182

Windows のカーソルの問題の解決 182
IntelliJ IDEA へのコピーと貼り付けの修正 182
自己署名証明書によるリダイレクトの説明 183
Windows の NVIDIA GPU でのマルチモニター/全画面表示の失敗 184
Amazon DCV のパフォーマンスと統計のモニタリング 185
Amazon DCV のパフォーマンスカウンターセット 186
Amazon DCV サーバー 186
Amazon DCV サーバープロセス 188
Amazon DCV サーバーセッション 189
Amazon DCV サーバー接続 191
Amazon DCV サーバーチャネル 192
Amazon DCV サーバーイメージング 193
サーバーパラメータリファレンス 195
`audio` パラメータ 196
`clipboard` パラメータ 197
`connectivity` パラメータ 201
`display` パラメータ 209
`display/linux` パラメータ 218
`extensions` パラメータ 220
`input` パラメータ 221
`license` パラメータ 222
`log` パラメータ 224
`printer` パラメータ 229
`redirection` パラメータ 232
`security` パラメータ 233
`session-management` パラメータ 244
`session-management/automatic-console-session` パラメータ 249
`session-management/defaults` パラメータ 251
`smartcard` パラメータ 252
`webauthn` パラメータ 253
`webcam` パラメータ 254
`windows` パラメータ 256
設定パラメータの変更 257
Windows Amazon DCV サーバー 257
Linux Amazon DCV サーバー 258
Amazon DCV のサポート終了 260

EOSL タイムライン 260
お客様向けの EOSL パス 262
EOSL に関するよくある質問 262
セキュリティ 264
データ保護 264
データの暗号化 265
コンプライアンス検証 266
リリースノートとドキュメント履歴 268
リリースノート 268
Amazon DCV 2024.0-19030 270
Amazon DCV 2024.0-19030 271
Amazon DCV 2024.0-19030 272
Amazon DCV 2024.0-18131 273
Amazon DCV 2024.0-17979 274
Amazon DCV 2023.1-17701 275
Amazon DCV 2023.1-17701 277
Amazon DCV 2023.1-16388 278
Amazon DCV 2023.1-16388 279
Amazon DCV 2023.1-16388 281
Amazon DCV 2023.1-16220 282
Amazon DCV 2023.0-15487 284
Amazon DCV 2023.0-15065 285
Amazon DCV 2023.0-15022 286
Amazon DCV 2023.0-14852 288
Amazon DCV 2022.2-14521 289
Amazon DCV 2022.2-14357 290
Amazon DCV 2022.2-14175 290
Amazon DCV 2022.2-14126 291
Amazon DCV 2022.2-13907 292
Amazon DCV 2022.1-13300 293
Amazon DCV 2022.1-13216 294
Amazon DCV 2022.1-13067 294
Amazon DCV 2022.0-12760 295
Amazon DCV 2022.0-12627 296
Amazon DCV 2022.0-12123 296
Amazon DCV 2022.0-11954 297

Amazon DCV 2021.3-11591 299
Amazon DCV 2021.2-11445 299
Amazon DCV 2021.2-11190 300
Amazon DCV 2021.2-11135 300
Amazon DCV 2021.2-11048 301
DCV 2021.1-10851 304
DCV 2021.1-10598 304
DCV 2021.1-10557 305
DCV 2021.0-10242 306
DCV 2020.2-9662 307
DCV 2020.2-9508 307
DCV 2020.1-9012 309
DCV 2020.1-9012 309
DCV 2020.1-8942 309
DCV 2020.0-8428 311
DCV 2019.1-7644 312
DCV 2019.1-7423 313
DCV 2019.0-7318 313
DCV 2017.4-6898 315
DCV 2017.3-6698 316
DCV 2017.2-6182 319
DCV 2017.1-5870 321
DCV 2017.1-5777 322
DCV 2017.0-5600 322
DCV 2017.0-5121 323
DCV 2017.0-4334 324
DCV 2017.0-4100 324
ドキュメント履歴 324
..... cccxxxii

Amazon VPC とは

> **Note**
>
> Amazon DCV は以前は NICE DCV と呼ばれていました。

Amazon 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。Amazon 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

- Amazon DCV の仕組み
- Amazon DCV の機能
- Amazon DCV の料金

Amazon DCV の仕組み

Amazon DCV を使用するには、サーバーに Amazon DCV サーバーソフトウェアをインストールします。Amazon 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、Amazon 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、Amazon 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。

Amazon DCV の機能

Amazon 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 Amazon 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 Amazon DCV サーバーのみ) — Linux Amazon 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 Amazon DCV サーバーのみ) — Linux Amazon 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 Amazon 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 — Amazon DCV セッションで周辺機器をローカルコンピュータ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使用できる 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。

Amazon DCV の料金

Amazon EC2 インスタンスで Amazon 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、「ステップ 2: Amazon DCV サーバーのライセンスを取得する」を参照してください。

Amazon DCV サーバーについて

Amazon DCV は、ユーザーセッションを作成する専有サーバーにインストールされます。Amazon 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 Amazon DCV サーバーを選択します。次の表は、Windows および Linux の Amazon 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

Amazon 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、Amazon DCV サーバーから Amazon 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 Amazon DCV サーバーをインストールする場合は、Amazon EC2 G3, G4dn, G4ad, G5、または G6 インスタンスタイプ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」、Amazon EC2 G5 インスタンス」、およびAmazon EC2 G6 インスタンス」を参照してください。

Amazon 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 Amazon DCV 仮想ディスプレイドライバー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	• Windows 10 • Windows 11 • Windows Server 2016 • [Windows Server 2019] • Windows Server 2022 Note 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。	• Amazon Linux 2 • Amazon Linux 2023 • CentOS Stream 9 • RHEL 8.x • RHEL 9.x • SP5 以降の SUSE Linux Enterprise 12 • SUSE Linux Enterprise 15 with SP5 • Rocky Linux 8.5 以降 • Rocky Linux 9 • Ubuntu 20.04 • Ubuntu 22.04 • Ubuntu 24.04
サポートされているアーキテクチャ	64 ビット x86	• 64 ビット x86 • 64 ビット ARM (Amazon Linux 2、Amazon Linux 20Amazon EC23、RHEL 8.x/9.x、CentOS 9、Rocky Linux 8/9、Ubuntu 22.04、および Ubuntu 24.04 を実行している Amazon EC2 インスタンスでのみサポート)
GPU	(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。 Note • 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 3.5 以上が必要です。	

	Windows Server	[Linux サーバー]
	• 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 `amf-amdgpu-pro` 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。	
		仮想セッション間での GPU 共有には NVIDIA GPU が必要です。 **Note** 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。
ネットワーク	デフォルトでは、Amazon DCV サーバーは、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。サーバーで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

CentOS 7 や RHEL 7 などのサポート終了に達したオペレーティングシステムを実行している場合でも、Amazon DCV はサポートされている DCV バージョンのサポート終了までサポートを提供します。Amazon 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。

Amazon DCV クライアント要件の詳細については、「Amazon DCV ユーザーガイド」の「Amazon 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 Amazon 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	Windows Amazon DCV サーバー	Linux Amazon DCV サーバー
コンソールセッション	✓	✓
仮想セッション	✗	✓
QUIC (UDP) トランスポートプロトコル	✓	✓
設定可能な TCP/UDP ポートおよびアドレス	✓	✓
カスタム TLS 証明書	✓	✓
アイドル状態のクライアントの切断	✓	✓
GPU 共有	✗	✓
USB リモート化	✓	✓
スマートカードのサポート	✓	✓
ウェブカメラのサポート	✓ (Windows 10 および Server 2016 以降)	✗
セッションストレージとファイル転送	✓	✓
コピーと貼り付け	✓	✓
カスタム HTTP ヘッダー	✓	✓
セッションからの印刷	✓	✓
ステレオ 2.0 オーディオ再生	✓	✓

機能	Windows Amazon DCV サーバー	Linux Amazon DCV サーバー
サラウンドサウンドオーディオ再生	✓ (7.1 まで)	✓ (5.1 まで)
ステレオ 2.0 オーディオ録音	✓	✓
タッチスクリーンのサポート	✓ (Windows 10 および Server 2016 以降)	✓
スタイラスのサポート	✓ (Windows 10 および Server 2019)	✓
ゲームパッドのサポート	✓ (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 のリダイレクト	✓	✗

Amazon DCV クライアントの機能の詳細については、「Amazon DCV ユーザーガイド」の「Amazon DCV クライアントの機能」を参照してください。

Amazon DCV サーバーのセットアップ

Amazon DCV を使用するには、Amazon DCV セッションをホストするサーバーに Amazon DCV 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。

次のトピックでは、Amazon DCV サーバーをインストールしてライセンスを取得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 インスタンスでの Amazon DCV サーバーの使用にはライセンスが不要であるためです。

トピック

- ステップ 1: Amazon DCV サーバーをインストールする
- ステップ 2: Amazon DCV サーバーのライセンスを取得する
- ステップ 3: Amazon DCV サーバーのイメージングを設定する (オプション)

ステップ 1: Amazon DCV サーバーをインストールする

次のトピックでは、Amazon DCV サーバーの最新バージョンを Windows および Linux に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、オンプレミスサーバー、またはクラウドベースサーバーに Amazon DCV をインストールする場合は、以下のステップを実行します。

> **Note**
>
> Amazon DCV サーバーを旧バージョンから最新バージョンにアップグレードする場合は、「Amazon DCV サーバーのアップグレード」を参照してください。

トピック

- Windows への Amazon DCV サーバーのインストール
- Linux への Amazon DCV サーバーのインストール

Windows への Amazon DCV サーバーのインストール

Amazon DCV サーバーソフトウェアを Windows サーバーにインストールし、そこから Amazon DCV セッションを実行できます。ソフトウェアをインストールする前に、サーバーがソフトウェア

を実行するための前提条件を満たしていることを確認してください。ソフトウェアをインストールするプロセスは、インストールウィザードを使用して手動で実行するか、Amazon DCV によって自動的にインストールできます。

トピック

- Amazon EC2 インスタンスでの Windows Amazon DCV サーバーの前提条件
- Windows への Amazon DCV サーバーのインストール

Amazon EC2 インスタンスでの Windows Amazon DCV サーバーの前提条件

このトピックでは、Amazon 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 の Windows インスタンスに Amazon DCV サーバーをインストールしない場合は、これらの前提条件をスキップしてください。

トピック

- すべてのインスタンスの前提条件
- 高速コンピューティングインスタンスの前提条件
- その他のインスタンスファミリーの前提条件

すべてのインスタンスの前提条件

バージョン 2024.0 以降、Windows バージョンの Amazon DCV では、Visual Studio 2017 用の Microsoft Visual C++ 再頒布可能パッケージではなく、Visual Studio 2022 用の Microsoft Visual C++ 再頒布可能パッケージを要件としています。

ベストプラクティスは、サーバー管理者が Amazon DCV サーバーをインストールする前に、Visual Studio 用の Microsoft Visual C++ 再頒布可能パッケージがインストールしていることです。2024.0 Amazon DCV MSI インストーラは依存関係をチェックし、見つからない場合は、Amazon DCV のインストール前に要件のインストールを試みます。この動作はフォールバックメカニズムであり、今後のリリースで削除されます。インストールに自動化を使用する管理者は、Amazon DCV サーバーの前に、自動化を更新して Visual Studio 用の Microsoft Visual C++ 再配布可能パッケージをインストールする必要があります。また、Microsoft Visual C++ 再頒布可能パッケージは、インストールの一環としてホストを再起動する場合があることに注意してください。

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、G5 などのインスタンス) を使用している場合は、適切な NVIDIA ドライバーまたは 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。GPU ドライバーにより次の機能が有効になります。

- 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 グラフィックスインスタンスに 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。

- NVIDIA GPU とともに使用するインスタンス (G2、G3、G4dn、G5 などのインスタンス) 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。
- AMD GPU を使用するインスタンス (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、ブログ投稿記事「新しい Amazon EC2 G4ad インスタンスの詳細」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 などのインスタンス) を使用している場合は、適切な NVIDIA 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。NVIDIA GPU ドライバーにより、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が有効になります。

高速コンピューティングインスタンスに 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「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 NVIDIA GPU 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、NVIDIA GRID ドライバーをインストールします。詳細については、NVIDIA ウェブサイトの「NVIDIA vGPU ソフトウェア」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については、Windows 2016 を使用しているか、2023.1 以前のバージョンの Amazon DCV サーバーを実行している場合は、Amazon 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。

Amazon DCV 仮想ディスプレイドライバーをインストールすると、以下が有効になります。

- 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 解像度に対するサポート

Amazon DCV サーバーに接続されたサーバーモニターを、Windows コントロールパネルで管理することはできません。

> **Note**
>
> Amazon 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) が DCV サーバーにパッケージ化されているため、DCV サーバー 2023.1 以降で Windows Server 2019 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。IDD が推奨されますが、GetConsoleScreenshot 機能は期待どおりに動作しません。

> **Important**
>
> NVIDIA GPU ドライバーなどといった他の GPU ドライバーと一緒に Amazon DCV 仮想ディスプレイ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。競合を回避するために、Amazon 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 Amazon DCV 仮想ディスプレイドライバーをインストールする方法

1. Amazon DCV 仮想ディスプレイドライバーのインストーラを Amazon DCV ウェブサイトからダウンロードします。
2. 次のいずれかを実行して、ドライバーをインストールします。
   - インストールウィザードを実行します。
   - インストールファイルをダブルクリックします。
   - 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-88.msi /quiet /norestart
```

3. インスタンスを再起動します。
4. インスタンスに再接続します。

Windows への Amazon DCV サーバーのインストール

インストールウィザードを使用して、Amazon DCV サーバーを Windows ホストサーバーにインストールすることができます。ウィザードで一連のステップを実行して、Amazon DCV サーバーのインストールをカスタマイズすることができます。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
- 自動インストールの使用

ウィザードの使用

Amazon DCV サーバーインストールウィザードを使用して、ガイド付きインストールを行います。

ウィザードを使用して Amazon DCV サーバーを Windows にインストールする方法

1. Amazon DCV サーバーのインストール先となるサーバーを起動して接続します。
2. Amazon DCV ウェブサイトから Amazon DCV サーバーインストーラをダウンロードします。

> **Note**
>
> Amazon DCV サーバーは、64 ビットバージョンでのみ利用可能で、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すれば、最新の Amazon DCV パッケージを自動的に取得できます。

3. `nice-dcv-server-x64-Release-2024.0-`*`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くなる] を選択します。
7. [DCV Service Configuration (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) Amazon DCV サーバーをインストール後に手動で起動するには、[いいえ、DCV サービスを手動で開始します] を選択します。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で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. [次へ] を選択します。
9. [DCV Session Management Configuration (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. [インストール] を選択します。

自動インストールの使用

Amazon DCV では、サーバーソフトウェアを自動的にインストールしてアクティブ化できます。これは「自動インストール」と呼ばれます。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。
- Amazon 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` — Amazon DCV サーバーがインストール後に自動的に開始されないようにします。
- `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: マイクロフォンドライバー
- audioSpkDriver: スピーカードライバー
- printerDriver: プリンタードライバー
- usbDriver: USB デバイスリモート化ドライバー (デフォルトでは無効)
- webcamDriver: ウェブカメラのドライバー
- gamepadDriver: ゲームパッドドライバー
- webClient: ウェブクライアント
- webauthn: Webauthn リダイレクト
- iddDriver: 間接ディスプレイドライバー (推奨)
- webrtc: WebRTC リダイレクトコンポーネント
- ALL: すべてのコンポーネント

自動インストールを使用して Windows に Amazon DCV サーバーをインストールする方法

1. Amazon DCV サーバーのインストール先となるサーバーを起動して接続します。
2. Amazon DCV ウェブサイトから Amazon DCV サーバーインストーラをダウンロードします。

   > **Note**
   >
   > Amazon DCV サーバーは、64 ビットバージョンでのみ利用可能で、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:\>  msiexec.exe /i nice-dcv-server-x64-Release-2024.0-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:\>  msiexec.exe /i nice-dcv-server-x64-Release-2024.0-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:\>  msiexec.exe /i nice-dcv-server-x64-Release-2024.0-version_number.msi
ADDLOCAL=audioMicDriver,audioSpkDriver,printerDriver,webcamDriver /quiet /
norestart /l*v dcv_install_msi.log
```

Linux への Amazon DCV サーバーのインストール

Amazon DCV サーバーソフトウェアを Linux サーバーにインストールし、そこから Amazon DCV セッションを実行できます。ソフトウェアをインストールする前に、サーバーがソフトウェアを実行するための前提条件を満たしていることを確認してください。ソフトウェアをインストールするプロセスは、インストールウィザードを使用して手動で実行するか、 AWSによって自動的にインストールできます。

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

トピック

- Linux Amazon DCV サーバーの前提条件
- Linux に Amazon 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の実行

Linux Amazon DCV サーバーの前提条件

Amazon DCV を使用すると、クライアントが Linux サーバーのリモートのグラフィカル 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。Amazon DCV では、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

グがサポートされています。コンソールおよび仮想セッションの詳細については、「Amazon 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 サーバーで Amazon DCV を使用するための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 プロトコルの無効化 (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
- グラフィックスインスタンス用の GPU ドライバーをインストールする
- GPU 以外のインスタンスに 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

Linux サーバーでの Amazon DCV のエクスペリエンスが向上するように、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できます。

デスクトップ環境は、Linux オペレーティングシステムとやり取りするの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。いくつかのデスクトップ環境があり、Amazon DCV はそれらの多くで動作します。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。

RHEL、CentOS、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 と 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
```

Amazon Linux 2

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
```

Amazon Linux 2023

Amazon Linux 2023 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 2023 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 dnf groupinstall '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るようにします。

```
$ sudo dnf upgrade
```

3. Linux サーバーを再起動します。

```
$ sudo reboot
```

Ubuntu 20.x, 22.x, and 24.x

Ubuntu 20.x/22.x/24.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 Amazon DCV でサポートされなくなりました。

Ubuntu 20.x/22.x/24.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
```

GDM3 のインストール

```
$ sudo apt install gdm3
```

2. GDM3 がデフォルトのデスクトップマネージャ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
```

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るようにします。

```
$ sudo apt upgrade
```

4. Linux サーバーを再起動します。

```
$ sudo reboot
```

> **Note**
>
> 2022.2 より古いバージョンの Amazon DCV を仮想セッションで使用すると、既知の GDM の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。
>
> - GPU を搭載していないサーバー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を無効にできます。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
> ```

- GPU を搭載したサーバーでは、デスクトップマネージャ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で X サーバ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 &
```

Amazon DCV 2022.2 以降の場合、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**
>
> 2022.2 より古いバージョンの Amazon DCV を仮想セッションで使用すると、既知の GDM の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。
>
> - GPU を搭載していないサーバー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を無効にできます。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
> ```

- GPU を搭載したサーバーでは、デスクトップマネージャ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で X サーバ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 &
```

Amazon DCV 2022.2 以降の場合、この問題の影響を受けません。

Wayland プロトコルの無効化 (GDM3 のみ)

Amazon DCV は Wayland プロトコルに対応していません。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。GDM3 を使用している場合は、このステップをスキップしてください。

Wayland プロトコルを無効にする方法

1. 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、SUSE Linux Enterprise 15.x、Amazon Linux 2023

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

   - Ubuntu

     ```
     /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、CentOS、Amazon Linux 2023

```
$ sudo systemctl restart gdm
```

- Ubuntu

```
$ 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 サーバーが正しく設定され、実行されている X サーバーがあることを確認する必要があります。

> **Note**
> GPU 共有を使用せずに仮想セッションを使用する場合、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。

Linux 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 DCV GL 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、DCV GL 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, Amazon Linux 2, and Amazon Linux 2023

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てください。

```
$ 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てください。

```
$ 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てください。

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

OpenGL ソフトウェアレンダリングの検証

GPU 非搭載の Linux 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。GPU 非搭載の Linux サーバーを使用していて、OpenGL を使用する予定の場合は、Mesa ドライバーがインストールされ、Linux サーバーで正しく設定されていることを確認してください。

> **Note**
> この条件は GPU 非搭載の Linux 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 Xdcv | 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
```

グラフィックスインスタンス用の GPU ドライバーをインストールする

トピック

- NVIDIA ドライバーをインストールして設定する
- AMD ドライバーをインストールして設定する

NVIDIA ドライバーをインストールして設定する

専用 NVIDIA GPU 搭載の Linux サーバーでは、適切な NVIDIA 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する必要があります。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 EC2 ユーザーガイド」の「Linux サーバーでの NVIDIA ドライバーのイントール」を参照してください。

> **Note**
>
> - この条件は NVIDIA GPU 搭載の Linux サーバーにのみ適用されます。
> - GRID ドライバーでは、インストールされた各 GPU に対して最大 4 つの 4K ディスプレイがサポートされます。ゲーミングドライバーでは、インストールされている各 GPU に対して 4 K ディスプレイが 1 つだけサポートされます。

Linux サーバーに NVIDIA ドライバーをインストールしたら、`xorg.conf` を更新してください。

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てください。

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

G3、G4、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
```

NVIDIA GPU がハードウェアベースのビデオエンコーディングに対応しているか確認する方法

NVENC エンコーディングがサポートされていることと、Ubuntu 20 では 3.0 以上、または 3.5 以上のコンピューティング機能があることを確認します。

NVENC のサポートを確認するには、「NVIDIA Video Encode and Decode GPU Support Matrix」(NVIDIA ビデオエンコード/デコードの GPU 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 Compute Capacility」(NVIDIA コンピューティングキャパシティ) テーブルを参照してください。

NVIDIA GPU で NVENC エンコーディングがサポートされていない場合、または必要なコンピューティング機能が備わっていない場合は、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 Xdcv | 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

AMD GPU がアタッチされたインスタンス (G4ad インスタンスなど) 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 AMD GPU 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、ブログ投稿記事「新しい Amazon EC2 G4ad インスタンスの詳細」を参照してください。

GPU 以外のインスタンスに XDummy ドライバーをインストールする

トピック

- xDummy ドライバーをインストールして設定する

xDummy ドライバーをインストールして設定する

専用 GPU が搭載されていない Linux サーバーでコンソールセッションを使用する場合は、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、実際の GPU が存在しない場合に X サーバー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、その設定で定義されている解像度しか対応できません。

RHEL, CentOS, Rocky Linux, Amazon Linux 2, and Amazon Linux 2023

xDummy ドライバーをインストールする方法

次のコマンドを実行してください。

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

Ubuntu

xDummy ドライバーをインストールする方法

次のコマンドを実行してください。

```
$ 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方法

次のコマンドを実行してください。

```
$ 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、`xorg.conf` を更新します。

xorg.conf で xDummy を設定する方法

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 Amazon DCV サーバーをインストールする

Amazon 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 パッケージまたは .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**
>
> Amazon 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。

Amazon DCV サーバーのインストール

Amazon Linux 2

Amazon DCV サーバーは、64 ビット x86 と 64 ビット ARM のアーキテクチャをベースにした Amazon Linux 2 サーバーに利用できます。

> **Important**
>
> `nice-dcv-gl` と `nice-dcv-gltest` のパッケージは、64 ビット ARM アーキテクチャに基づくサーバーには使用できません。

Amazon Linux 2 に Amazon DCV サーバーをインストールするには

1. Amazon DCV サーバーのインストール先となるサーバーを起動して接続します。
2. Amazon DCV サーバーパッケージには、安全な GPG 署名によるデジタル署名があります。パッケージマネージャーで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いて Amazon DCV GPG キーをインポートします。

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

3. Amazon DCV ダウンロードのウェブサイトからパッケージをダウンロードします。RPM パッケージおよび deb パッケージは `.tgz` アーカイブにパッケージ化されています。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/2024.0/Servers/nice-
   dcv-2024.0-19030-amzn2-x86_64.tgz
   ```

   - 64 ビット ARM

   ```
   $ wget https://d1uj6qtbmh3dt5.cloudfront.net/2024.0/Servers/nice-
   dcv-2024.0-19030-amzn2-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すれば、最新の Amazon 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-amzn2-x86_64.tgz
> ```
>
> - 64 ビット ARM
>
> ```
> $ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-amzn2-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-2024.0-19030-amzn2-x86_64.tgz && cd nice-
     dcv-2024.0-19030-amzn2-x86_64
     ```

   - 64 ビット ARM

     ```
     $ tar -xvzf nice-dcv-2024.0-19030-amzn2-aarch64.tgz && cd nice-
     dcv-2024.0-19030-amzn2-aarch64
     ```

5. Amazon 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-2024.0.19030-1.el7.x86_64.rpm
     ```

   - 64 ビット ARM

     ```
     $ sudo yum install nice-dcv-server-2024.0.19030-1.el7.aarch64.rpm
     ```

6. (オプション) Amazon 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-2024.0.19030-1.el7.x86_64.rpm
     ```

   - 64 ビット ARM

     ```
     $ sudo yum install nice-dcv-web-viewer-2024.0.19030-1.el7.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-2024.0.654-1.el7.x86_64.rpm
     ```

   - 64 ビット ARM

```
$ sudo yum install nice-xdcv-2024.0.654-1.el7.aarch64.rpm
```

8. (オプション) GPU 共有を使用する予定がある場合は、`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-2024.0.1096-1.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。

9. (オプション) Amazon DCV EnginFrame で Amazon DCV を使用する予定がある場合は、`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-2024.0.266-1.el7.x86_64.rpm
   ```

   - 64 ビット ARM

   ```
   $ sudo yum install nice-dcv-simple-external-
   authenticator-2024.0.266-1.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、DCV USB 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、Dynamic Kernel Module Support (DKMS) がサーバーにインストールされている必要があります。DKMS をインストールするには、以下のコマンドを使用します。

    DKMS は、Extra Packages for Enterprise Linux (EPEL) リポジトリからインストールできます。EPEL リポジトリを有効にするには、以下のコマンドを実行します。

```
$ 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、以下のコマンドを実行して DKMS をインストールします。

```
$ sudo yum install dkms
```

DKMS をインストールしたら、以下のコマンドを実行して DCV USB 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
```

Amazon Linux 2023

Amazon 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 2023 サーバーで使用できます。

> **Important**
>
> `nice-dcv-gl` と `nice-dcv-gltest` のパッケージは、64 ビット ARM アーキテクチャに基づくサーバーには使用できません。

Amazon Linux 2023 に Amazon DCV サーバーをインストールするには

1. Amazon DCV サーバーのインストール先にするサーバーを起動して接続します。
2. Amazon DCV サーバーパッケージには、安全な GPG 署名によるデジタル署名があります。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、Amazon DCV 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いて Amazon DCV GPG キーをインポートします。

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

3. Amazon DCV ダウンロードのウェブサイトからパッケージをダウンロードします。RPM パッケージおよび deb パッケージは `.tgz` アーカイブにパッケージ化されています。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/2024.0/Servers/nice-dcv-2024.0-19030-amzn2023-x86_64.tgz
     ```

   - 64 ビット ARM

     ```
     $ wget https://d1uj6qtbmh3dt5.cloudfront.net/2024.0/Servers/nice-dcv-2024.0-19030-amzn2023-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すれば、最新の Amazon 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-amzn2023-x86_64.tgz
   > ```
   >
   > - 64 ビット ARM
   >
   > ```
   > $ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-amzn2023-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-2024.0-19030-amzn2023-x86_64.tgz && cd nice-dcv-2024.0-19030-amzn2023-x86_64
     ```

- 64 ビット ARM

```
$ tar -xvzf nice-dcv-2024.0-19030-amzn2023-aarch64.tgz && cd nice-
dcv-2024.0-19030-amzn2023-aarch64
```

5. Amazon DCV サーバーをインストールします。

   - 64 ビット x86

   ```
   $ sudo dnf install nice-dcv-server-2024.0.19030-1.amzn2023.x86_64.rpm
   ```

   - 64 ビット ARM

   ```
   $ sudo dnf install nice-dcv-server-2024.0.19030-1.amzn2023.aarch64.rpm
   ```

6. (オプション) Amazon 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予定がある場合は、`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 dnf install nice-dcv-web-viewer-2024.0.19030-1.amzn2023.x86_64.rpm
   ```

   - 64 ビット ARM

   ```
   $ sudo dnf install nice-dcv-web-viewer-2024.0.19030-1.amzn2023.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 dnf install nice-xdcv-2024.0.654-1.amzn2023.x86_64.rpm
   ```

   - 64 ビット ARM

   ```
   $ sudo dnf install nice-xdcv-2024.0.654-1.amzn2023.aarch64.rpm
   ```

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

   - 64 ビット x86

```
$ sudo dnf install nice-dcv-gl-2024.0.1096-1.amzn2023.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) Amazon DCV 外部認証ツールを使用する場合は、 nice-dcv-simple-external-authenticatorパッケージを使用します。

   - 64 ビット x86

   ```
   $ sudo dnf install nice-dcv-simple-external-
   authenticator-2024.0.266-1.amzn2023.x86_64.rpm
   ```

   - 64 ビット ARM

   ```
   $ sudo dnf install nice-dcv-simple-external-
   authenticator-2024.0.266-1.amzn2023.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、DCV USB 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、Dynamic Kernel Module Support (DKMS) がサーバーにインストールされている必要があります。DKMS をインストールするには、以下のコマンドを使用します。

    ```
    $ sudo dnf install dkms
    ```

    DKMS をインストールしたら、以下のコマンドを実行して DCV USB 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 dnf install pulseaudio-utils
```

RHEL, CentOS, and Rocky Linux 8.5

Amazon DCV サーバーは、64 ビット x86 と 64 ビット ARM のアーキテクチャをベースにした RHEL および CentOS サーバーと、Rocky Linux 8.5 以降に利用できます。

> **Important**
>
> `nice-dcv-gl` と `nice-dcv-gltest` のパッケージは、64 ビット ARM アーキテクチャに基づくサーバーには使用できません。

Amazon DCV サーバーを RHEL、CentOS、Rocky Linux 8.5 にインストールする方法

1. Amazon DCV サーバーのインストール先にするサーバーを起動して接続します。
2. Amazon DCV サーバーパッケージには、安全な GPG 署名によるデジタル署名があります。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、Amazon DCV 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いて Amazon DCV GPG キーをインポートします。

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

3. Amazon DCV ダウンロードのウェブサイトからパッケージをダウンロードします。RPM パッケージおよび deb パッケージは `.tgz` アーカイブにパッケージ化されています。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/2024.0/Servers/nice-
dcv-2024.0-19030-el8-x86_64.tgz
```

   - 64 ビット ARM

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/2024.0/Servers/nice-
dcv-2024.0-19030-el8-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すれば、最新の Amazon 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-2024.0-19030-el8-x86_64.tgz && cd nice-dcv-2024.0-19030-
   el8-x86_64
   ```

   - 64 ビット ARM

   ```
   $ tar -xvzf nice-dcv-2024.0-19030-el8-aarch64.tgz && cd nice-dcv-2024.0-19030-
   el8-aarch64
   ```

5. Amazon 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-2024.0.19030-1.el8.x86_64.rpm
   ```

   - 64 ビット ARM

   ```
   $ sudo yum install nice-dcv-server-2024.0.19030-1.el8.aarch64.rpm
   ```

6. (オプション) Amazon 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予定がある場合は、`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-2024.0.19030-1.el8.x86_64.rpm
   ```

   - 64 ビット ARM

   ```
   $ sudo yum install nice-dcv-web-viewer-2024.0.19030-1.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-2024.0.654-1.el8.x86_64.rpm
   ```

   - 64 ビット ARM

   ```
   $ sudo yum install nice-xdcv-2024.0.654-1.el8.aarch64.rpm
   ```

8. (オプション) GPU 共有を使用する予定がある場合は、`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-2024.0.1096-1.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) Amazon DCV EnginFrame で Amazon DCV を使用する予定がある場合は、`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-2024.0.266-1.el8.x86_64.rpm
```

- 64 ビット ARM

```
$ sudo yum install nice-dcv-simple-external-
authenticator-2024.0.266-1.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には、DCV USB 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、Dynamic Kernel Module Support (DKMS) がサーバーにインストールされている必要があります。DKMS をインストールするには、以下のコマンドを使用します。

    DKMS は、Extra Packages for Enterprise Linux (EPEL) リポジトリからインストールできます。EPEL リポジトリを有効にするには、以下のコマンドを実行します。

    ```
    $ 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、以下のコマンドを実行して DKMS をインストールします。

    ```
    $ sudo yum install dkms
    ```

    DKMS をインストールしたら、以下のコマンドを実行して DCV USB 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

Amazon DCV サーバーは、64 ビット x86 と 64 ビット ARM のアーキテクチャをベースにした RHEL および CentOS サーバーと、Rocky Linux 9 以降に利用できます。

> **Important**
>
> `nice-dcv-gl` と `nice-dcv-gltest` のパッケージは、64 ビット ARM アーキテクチャに基づくサーバーには使用できません。

Amazon DCV サーバーを RHEL、CentOS、Rocky Linux 9 にインストールする方法

1. Amazon DCV サーバーのインストール先にするサーバーを起動して接続します。
2. Amazon DCV サーバーパッケージには、安全な GPG 署名によるデジタル署名があります。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、Amazon DCV 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いて Amazon DCV GPG キーをインポートします。

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

3. Amazon DCV ダウンロードのウェブサイトからパッケージをダウンロードします。RPM パッケージおよび deb パッケージは `.tgz` アーカイブにパッケージ化されています。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/2024.0/Servers/nice-
     dcv-2024.0-19030-el9-x86_64.tgz
     ```

   - 64 ビット ARM

     ```
     $ wget https://d1uj6qtbmh3dt5.cloudfront.net/2024.0/Servers/nice-
     dcv-2024.0-19030-el9-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すれば、最新の Amazon 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-2024.0-19030-el9-x86_64.tgz && cd nice-dcv-2024.0-19030-
   el9-x86_64
   ```

   - 64 ビット ARM

   ```
   $ tar -xvzf nice-dcv-2024.0-19030-el9-aarch64.tgz && cd nice-dcv-2024.0-19030-
   el9-aarch64
   ```

5. Amazon 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-2024.0.19030-1.el9.x86_64.rpm
   ```

   - 64 ビット ARM

   ```
   $ sudo yum install nice-dcv-server-2024.0.19030-1.el9.aarch64.rpm
   ```

6. (オプション) Amazon 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予定がある場合は、`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-2024.0.19030-1.el9.x86_64.rpm
```

- 64 ビット ARM

```
$ sudo yum install nice-dcv-web-viewer-2024.0.19030-1.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-2024.0.654-1.el9.x86_64.rpm
```

- 64 ビット ARM

```
$ sudo yum install nice-xdcv-2024.0.654-1.el9.aarch64.rpm
```

8. (オプション) GPU 共有を使用する予定がある場合は、`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-2024.0.1096-1.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) Amazon DCV EnginFrame で Amazon DCV を使用する予定がある場合は、`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-2024.0.266-1.el9.x86_64.rpm
```

- 64 ビット ARM

```
$ sudo yum install nice-dcv-simple-external-
authenticator-2024.0.266-1.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には、DCV USB 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、Dynamic Kernel Module Support (DKMS) がサーバーにインストールされている必要があります。DKMS をインストールするには、以下のコマンドを使用します。

    DKMS は、Extra Packages for Enterprise Linux (EPEL) リポジトリからインストールできます。EPEL リポジトリを有効にするには、以下のコマンドを実行します。

    ```
    $ 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、以下のコマンドを実行して DKMS をインストールします。

    ```
    $ sudo yum install dkms
    ```

    DKMS をインストールしたら、以下のコマンドを実行して DCV USB 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
    ```

SLES 12.x/15.x

Amazon DCV サーバーは、64 ビット x86 アーキテクチャに基づく SUSE Linux Enterprise Server (SLES) 12.x/15.x サーバーにのみ使用できます。

Amazon DCV サーバーを SLES 12.x/15.x にインストールする方法

1. Amazon DCV サーバーのインストール先にするサーバーを起動して接続します。
2. Amazon DCV サーバーパッケージには、安全な GPG 署名によるデジタル署名があります。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、Amazon DCV 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いて NICE GPG キーをインポートします。

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

3. Amazon DCV ダウンロードのウェブサイトからパッケージをダウンロードします。RPM パッケージおよび deb パッケージは `.tgz` アーカイブにパッケージ化されています。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/2024.0/Servers/nice-dcv-2024.0-19030-sles12-x86_64.tgz
   ```

   - SLES 15.x

   ```
   $ curl -O https://d1uj6qtbmh3dt5.cloudfront.net/2024.0/Servers/nice-dcv-2024.0-19030-sles15-x86_64.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すれば、最新の Amazon 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-2024.0-19030-sles12-x86_64.tgz && cd nice-dcv-2024.0-19030-sles12-x86_64
   ```

   - SLES 15.x

   ```
   $ tar -xvzf nice-dcv-2024.0-19030-sles15-x86_64.tgz && cd nice-dcv-2024.0-19030-sles15-x86_64
   ```

5. Amazon DCV サーバーをインストールします。

   - SLES 12.x

   ```
   $ sudo zypper install nice-dcv-server-2024.0.19030-1.sles12.x86_64.rpm
   ```

   - SLES 15.x

   ```
   $ sudo zypper install nice-dcv-server-2024.0.19030-1.sles15.x86_64.rpm
   ```

6. (オプション) Amazon 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予定がある場合は、`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-2024.0.19030-1.sles12.x86_64.rpm
   ```

   - SLES 15.x

   ```
   $ sudo zypper install nice-dcv-web-viewer-2024.0.19030-1.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-2024.0.654-1.sles12.x86_64.rpm
```

- SLES 15.x

```
$ sudo zypper install nice-xdcv-2024.0.654-1.sles15.x86_64.rpm
```

8. (オプション) GPU 共有を使用する予定がある場合は、`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-2024.0.1096-1.sles12.x86_64.rpm
   ```

   - SLES 15.x

   ```
   $ sudo zypper install nice-dcv-gl-2024.0.1096-1.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) Amazon DCV EnginFrame で Amazon DCV を使用する予定がある場合は、`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-2024.0.266-1.sles12.x86_64.rpm
   ```

   - SLES 15.x

   ```
   $ sudo zypper install nice-dcv-simple-external-
   authenticator-2024.0.266-1.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には、DCV USB 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、Dynamic Kernel Module Support (DKMS) がサーバーにインストールされている必要があります。DKMS をインストールするには、以下のコマンドを使用します。

    以下のコマンドを実行して DKMS をインストールします。

    - 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 SP2 を使用している場合は、上記コマンドの *15* を 15.1 または 15.2 のいずれかに置き換えます。

      DKMS をインストールします。

      ```
      $ 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
```

DKMS をインストールしたら、以下のコマンドを実行して DCV USB 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 20.04/22.04/24.04

Amazon DCV サーバーは、64 ビット x86 と 64 ビット ARM のアーキテクチャをベースにした Ubuntu サーバーに使用できます。

> **Important**
>
> `nice-dcv-gl` と `nice-dcv-gltest` のパッケージは、64 ビット ARM アーキテクチャに基づくサーバーには使用できません。

Amazon DCV サーバーを Ubuntu 20.04/22.04/24.04 にインストールする方法

1. Amazon DCV サーバーのインストール先にするサーバーを起動して接続します。
2. Amazon DCV サーバーパッケージには、安全な GPG 署名によるデジタル署名があります。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、Amazon DCV 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いて Amazon DCV 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
```

3. Amazon DCV ダウンロードのウェブサイトからパッケージをダウンロードします。RPM パッケージおよび deb パッケージは `.tgz` アーカイブにパッケージ化されています。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/2024.0/Servers/nice-dcv-2024.0-19030-ubuntu2004-x86_64.tgz
     ```

   - Ubuntu 22.04 (64 ビット x86)

     ```
     $ wget https://d1uj6qtbmh3dt5.cloudfront.net/2024.0/Servers/nice-dcv-2024.0-19030-ubuntu2204-x86_64.tgz
     ```

   - Ubuntu 22.04 (64 ビット ARM)

     ```
     $ wget https://d1uj6qtbmh3dt5.cloudfront.net/2024.0/Servers/nice-dcv-2024.0-19030-ubuntu2204-aarch64.tgz
     ```

   - Ubuntu 24.04 (64 ビット x86)

     ```
     $ wget https://d1uj6qtbmh3dt5.cloudfront.net/2024.0/Servers/nice-dcv-2024.0-19030-ubuntu2404-x86_64.tgz
     ```

   - Ubuntu 24.04 (64 ビット ARM)

     ```
     $ wget https://d1uj6qtbmh3dt5.cloudfront.net/2024.0/Servers/nice-dcv-2024.0-19030-ubuntu2404-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すれば、最新の Amazon 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
```

- Ubuntu 24.04 (64 ビット x86)

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-ubuntu2404-
x86_64.tgz
```

- Ubuntu 24.04 (64 ビット ARM)

```
$ wget https://d1uj6qtbmh3dt5.cloudfront.net/nice-dcv-ubuntu2404-
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-2024.0-19030-ubuntu2004-x86_64.tgz && cd nice-
dcv-2024.0-19030-ubuntu2004-x86_64
```

- Ubuntu 22.04 (64 ビット x86)

```
$ tar -xvzf nice-dcv-2024.0-19030-ubuntu2204-x86_64.tgz && cd nice-
dcv-2024.0-19030-ubuntu2204-x86_64
```

- Ubuntu 22.04 (64 ビット ARM)

```
$ tar -xvzf nice-dcv-2024.0-19030-ubuntu2204-aarch64.tgz && cd nice-
dcv-2024.0-19030-ubuntu2204-aarch64
```

- Ubuntu 24.04 (64 ビット x86)

```
$ tar -xvzf nice-dcv-2024.0-19030-ubuntu2404-x86_64.tgz && cd nice-
dcv-2024.0-19030-ubuntu2404-x86_64
```

- Ubuntu 24.04 (64 ビット ARM)

```
$ tar -xvzf nice-dcv-2024.0-19030-ubuntu2404-aarch64.tgz && cd nice-
dcv-2024.0-19030-ubuntu2404-aarch64
```

5. Amazon 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_2024.0.19030-1_amd64.ubuntu2004.deb
```

- Ubuntu 22.04 (64 ビット x86)

```
$ sudo apt install ./nice-dcv-server_2024.0.19030-1_amd64.ubuntu2204.deb
```

- Ubuntu 22.04 (64 ビット ARM)

```
$ sudo apt install ./nice-dcv-server_2024.0.19030-1_arm64.ubuntu2204.deb
```

- Ubuntu 24.04 (64 ビット x86)

```
$ sudo apt install ./nice-dcv-server_2024.0.19030-1_amd64.ubuntu2404.deb
```

- Ubuntu 24.04 (64 ビット ARM)

```
$ sudo apt install ./nice-dcv-server_2024.0.19030-1_arm64.ubuntu2404.deb
```

6. (オプション) Amazon 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予定がある場合は、`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_2024.0.19030-1_amd64.ubuntu2004.deb
```

- Ubuntu 22.04 (64 ビット x86)

```
$ sudo apt install ./nice-dcv-web-viewer_2024.0.19030-1_amd64.ubuntu2204.deb
```

- Ubuntu 22.04 (64 ビット ARM)

```
$ sudo apt install ./nice-dcv-web-viewer_2024.0.19030-1_arm64.ubuntu2204.deb
```

- Ubuntu 24.04 (64 ビット x86)

```
$ sudo apt install ./nice-dcv-web-viewer_2024.0.19030-1_amd64.ubuntu2404.deb
```

- Ubuntu 24.04 (64 ビット ARM)

```
$ sudo apt install ./nice-dcv-web-viewer_2024.0.19030-1_arm64.ubuntu2404.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_2024.0.654-1_amd64.ubuntu2004.deb
```

- Ubuntu 22.04 (64 ビット x86)

```
$ sudo apt install ./nice-xdcv_2024.0.654-1_amd64.ubuntu2204.deb
```

- Ubuntu 22.04 (64 ビット ARM)

```
$ sudo apt install ./nice-xdcv_2024.0.654-1_arm64.ubuntu2204.deb
```

- Ubuntu 24.04 (64 ビット x86)

```
$ sudo apt install ./nice-xdcv_2024.0.654-1_amd64.ubuntu2404.deb
```

- Ubuntu 24.04 (64 ビット ARM)

```
$ sudo apt install ./nice-xdcv_2024.0.654-1_arm64.ubuntu2404.deb
```

9. (オプション) GPU 共有を使用する予定がある場合は、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_2024.0.1096-1_amd64.ubuntu2204.deb
```

> **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。

10. (オプション) Amazon DCV EnginFrame で Amazon DCV を使用する予定がある場合は、`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_2024.0.266-1_amd64.ubuntu2004.deb
```

- Ubuntu 22.04 (64 ビット x86)

```
$ sudo apt install ./nice-dcv-simple-external-
authenticator_2024.0.266-1_amd64.ubuntu2204.deb
```

- Ubuntu 22.04 (64 ビット ARM)

```
$ sudo apt install ./nice-dcv-simple-external-
authenticator_2024.0.266-1_arm64.ubuntu2204.deb
```

- Ubuntu 24.04 (64 ビット x86)

```
$ sudo apt install ./nice-dcv-simple-external-
authenticator_2024.0.266-1_amd64.ubuntu2404.deb
```

- Ubuntu 24.04 (64 ビット ARM)

```
$ sudo apt install ./nice-dcv-simple-external-
authenticator_2024.0.266-1_arm64.ubuntu2404.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には、DCV USB 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、Dynamic Kernel Module Support (DKMS) がサーバーにインストールされている必要があります。DKMS をインストールするには、以下のコマンドを使用します。

    DKMS は、Ubuntu の公式リポジトリで入手できます。以下のコマンドを実行して DKMS をインストールします。

    ```
    $ sudo apt install dkms
    ```

    DKMS をインストールしたら、以下のコマンドを実行して DCV USB 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 のみの場合、システム認証で SSO 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の実行

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

内容

- Amazon DCV サーバーに到達できることを確認します。
- X サーバーがアクセス可能であることを確認する
- DCV GL が正しくインストールされていることを確認する

- Amazon DCV DEB パッケージの署名を検証する

Amazon DCV サーバーに到達できることを確認します。

デフォルトでは、Amazon 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、Amazon 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、「Amazon DCV サーバーの 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 インスタンスで Amazon DCV をセットアップする場合は、セキュリティグループを作成します。これは、Amazon 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、「EC2 でセキュリティグループを設定する方法」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

Amazon DCV の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

Amazon DCV サーバーがインストールされると、`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てください。

```
$ sudo DISPLAY=:0 XAUTHORITY=$(ps aux | grep "X.*\-auth" | grep -v Xdcv | 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、Rocky、CentOS、Amazon Linux 2、Ubuntu、SUSE Linux Enterprise

```
$ 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てください。

```
$ sudo DISPLAY=:0 XAUTHORITY=$(ps aux | grep "X.*\-auth" | grep -v Xdcv | 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。以下のコマンドを実行して X サーバーを再起動し、DCV GL を無効にしてから再び有効にします。

- RHEL、Rocky、CentOS、Amazon Linux 2、Ubuntu、SUSE Linux Enterprise

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

```
$ sudo dcvgladmin disable
```

```
$ sudo dcvgladmin enable
```

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

DCV GL が正しくインストールされていることを確認する

DCV GL パッケージをインストールすると、dcvgldiag ユーティリティが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 サーバーの設定が DCV GL の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。

Amazon DCV DEB パッケージの署名を検証する

Amazon DCV をインストールした後、Debian パッケージ (DEB) の署名を検証できます。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 パッケージ署名を検証する方法

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_2024.0.19030-1_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。*[Example Key]* の代わりに該当するキーが表示されます。

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

ステップ 2: Amazon DCV サーバーのライセンスを取得する

Amazon DCV サーバーソフトウェアをインストールしたら、Amazon DCV を使用するためのライセンスをダウンロードしてインストールする必要があります。Amazon DCV のライセンス要件は、Amazon DCV サーバーのインストール先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、Amazon DCV バージョン 2017.0 以降にのみ適用されます。

Amazon DCV ライセンス要件

トピック

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

Amazon EC2 での Amazon DCV

および AWS Local Zones で実行されているインスタンスを含め、EC2 インスタンスに Amazon DCV サーバーをインストール AWS Outposts して使用するライセンスサーバーは必要ありません。Amazon DCV サーバーは、Amazon EC2 インスタンスで動作し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な状態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。インスタンスからインターネットにアクセスできない場合は、アウトバウンドの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、HTTPS 経由で Amazon S3 にアクセスできるように VPC 用のゲートウェイエンドポイントを設定します。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 バケットへの接続で問題が発生した場合は、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、`#####`プレースホルダーを AWS リージョン (など) に置き換えます`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。

  JSON

```
{
    "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。このサービスへのアクセスは、Amazon DCV サーバーのライセンスを適切に取得できるようにするために必要です。インスタンスメタデータサービスの詳細については、「Amazon EC2 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。

  カスタム Windows AMI を使用している場合は、EC2Launch 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ます。詳細については、Amazon EC2 EC2Launch を使用した Windows インスタンスの設定」を参照してください。

Amazon EC2 インスタンスに Amazon DCV サーバーをインストールして使用している場合、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、Amazon 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。

Amazon DCV のその他のユースケース

それ以外のすべてのユースケースの場合は、Amazon 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、Amazon 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、そのサーバーで Amazon DCV セッションを作成してホストすることができなくなります。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、Amazon DCV サーバーでは自動評価ライセンスがデフォルトになります。

- 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、ケースバイケース AWS で によって決定されます。 case-by-case 拡張評価ライセンスは有効期限に達すると無効になり、その後は、サーバーで Amazon DCV セッションを作成してホストすることはできなくなります。延長評価ライセンスは、Amazon DCV ウェブサイトの「How to Buy」ページに記載されている Amazon DCV 販売代理店または再販業者に依頼する必要があります。ライセンスは、Amazon 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。

- 実稼働ライセンス — 実稼働ライセンスは、Amazon DCV から購入する完全ライセンスです。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 Amazon 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 Amazon DCV 同時セッション数を制限することもできます。Amazon DCV 同時セッションごとに 1 つのライセンスが必要です。実稼働ライセンスは Reprise License Manager (RLM) 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配布されます。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、お使いの Amazon DCV サーバーで Amazon DCV セッションを作成してホストすることができなくなります。

Amazon DCV の永久ライセンスやサブスクリプションの購入方法については、Amazon DCV のウェブサイトの「How to Buy」を参照してください。お住まいの地域の Amazon DCV 販売代理店または再販業者が掲載されています。

ライセンス要件

- Amazon DCV クライアントにはライセンスは不要です。
- Amazon DCV サーバーライセンスファイルは、旧バージョンの Amazon DCV サーバーとの下位互換性があります。例えば、Amazon DCV サーバーバージョン 2021 のライセンスは Amazon DCV サーバーバージョン 2019 で使用できます。
- Amazon DCV サーバーのバージョンには、少なくとも同じバージョンの Amazon DCV サーバーライセンスが必要です。例えば、Amazon DCV サーバーバージョン 2021 を使用する場合、ライセンスバージョン 2021 以降が必要です。Amazon DCV サーバーを最新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、Amazon DCV の販売代理店または再販業者にお問い合わせください。

> **Note**
>
> Amazon DCV サーバーの互換性については、「互換性に関する考慮事項」を参照してください。

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 のクライアントアクセスライセンス (CAL、Client Access License) に加えて、サーバーのグラフィカルユーザーインターフェイス (GUI、Graphical User Interface) にリモートアクセスするユーザーごとに、使用している Windows Server バージョンの Windows Server リモートデスクトップサービス (RDS、Remote Desktop Services) CAL が必要です。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、Amazon DCV を使用してリモートの Windows Server ホストの 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 Amazon DCV サーバーを実行し、Windows Server AMI を使用する場合は、Amazon が Windows Server CAL のライセンスコストを負担し、管理のみを目的とした 2 つの Windows Server RDS CAL 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

Amazon DCV から拡張評価ライセンスをリクエストすると、ライセンスを定義する `license.lic` ファイルが送信されます。

拡張評価ライセンスをインストールするには

`license.lic` ファイルをサーバー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/] キーに移動して、[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) を購入して使用する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: Amazon 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。ライセンスファイルは、Reprise License Manager (RLM) サーバーにインストールする必要があります。

RLM の詳細については、Reprise Software 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. RLM License Administration Bundle を Reprise Software ウェブサイトからダウンロードします。

> **Note**
>
> Amazon DCV バージョン 2022.1 以降では、RLM サーバー 14 以上が必要です。以前のバージョンでは RLM 12 以上が必要です。
> RLM License Administration Bundle の最新の安定版をインストールすることをお勧めします。

2. RLM License Administration Bundle を `C:\RLM` にインストールします。

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. RLM License Administration Bundle を Reprise Software ウェブサイトからダウンロードします。

> **Note**
>
> Amazon DCV バージョン 2022.1 以降では、RLM サーバー 14 以上が必要です。以前のバージョンでは RLM 12 以上が必要です。
> RLM License Administration Bundle の最新の安定版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/license` ディレクトリ `/opt/nice/rlm` とディレクトリを作成します。

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

4. RLM License Administration Bundle の内容を `/opt/nice/rlm/` に展開し、ファイルが `rlm` ユーザーに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
```

このコマンドから、RLM サーバーのホスト ID が次のように返されます。

```
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てください。

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

   このコマンドは、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 を次のように返します。
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
```

この ID は記録され、DCV のライセンスの購入に使用されます。

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てください。

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

   複数の ID が ID のリストとして返されます。
3. 以下のコマンドを実行してください。

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

   この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 のリストとして複数の ID 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
```

この ID は記録され、DCV のライセンスの購入に使用されます。

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

Amazon DCV の永久ライセンスやサブスクリプションの購入方法については、Amazon DCV のウェブサイトの「How to Buy」を参照してください。お住まいの地域の Amazon DCV 販売代理店または再販業者が掲載されています。

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、Amazon DCV 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

Amazon DCV の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、ライセンスを定義する `license.lic` ファイルが届きます。`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。
- ライセンスを購入したときに指定した RLM サーバーのホスト ID。
- RLM サーバーの TCP ポート番号。デフォルトは `5053` です。
- ISV ポート番号。これは、RLM サーバーが Amazon DCV ライセンスリクエストをリッスンするオプションポートです。指定しない場合は、起動時に RLM によってランダムにポートが選択されます。
- ライセンスによってカバーされる Amazon DCV 製品、および各製品の以下の詳細:
  - ライセンスの対象であるメジャーバージョン (例: 2017 年の Amazon DCV 製品の場合は `2017`)。
  - 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 と dcv-gl の 2 つの Amazon DCV 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="abcdefghijklmnopqrstuvwxyz1234567890abcdefghijklmnopqrstuvwxyz1234567890ab"
LICENSE nice dcv-gl 2017 permanent 10 share=hi _ck=123454323x
 sig="1234567890abcdefghijklmnopqrstuvwxyz1234567890abcdefghijklmnopqrstuvwxyz12"
```

`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. `HOST` で始まる RLM サーバーのホスト名とTCPポート番号をファイルの最初の行に追加します。

   > **Warning**
   >
   > *`RLM_server_host_id`* は、ライセンスを購入したときに指定したホスト ID です。*`RLM_server_host_id`* は編集できません。

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

   ISV ポートを指定しない場合は、`port=`*`port_number`* を省略します。ISV ポートを指定しない場合は、起動のたびに RLM 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を使用できないようにファイアウォールを設定している場合は、`HOST` 行で指定されている RLM ポートに加えてこのポートを指定し、それを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、そのファイルを RLM サーバーに配置し、RLM サービスを開始する必要があります。

トピック

- Windows で RLM サーバーを設定する

- Linux で RLM サーバーを設定する

Windows で RLM サーバーを設定する

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. Amazon 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\` フォルダに入れます。
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、「Reprise Software License Manager (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 サーバーが実行されてい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 サーバーのバージョンによって `rlm.log` ファイルの内容が多少異なる場合があります。

   b. 以下のコマンドを実行してください。

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

このコマンドから RLM サーバーに関する情報が返されます。

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. Amazon DCV サーバーから `/usr/share/dcv/license/nice.set` ファイルをコピーし、RLM サーバーの `/opt/nice/rlm` に入れます。
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ようにします。

   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 $pid >/dev/null 2>&1
        sleep 3
        if [ -d "/proc/$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 サーバーが実行中で、予期したとおりに機能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 サーバーのバージョンによって `rlm.log` ファイルの内容が多少異なる場合があります。

ステップ 6: Amazon DCV サーバーを設定する

RLM サーバーを使用するように Amazon DCV サーバーを設定します。この設定のために、Amazon 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、接続する RLM サーバーを指定して *RLM_server_port*@*RLM_server* の形式で設定する必要があります。RLM サーバーは、ホスト名または IP アドレスで指定できます。明示的に設定しない場合、RLM サーバーポートはデフォルトで `5053` です。

複数の RLM サーバーを使用している場合は、Linux の場合は `:`、Windows の場合は `;` で区切って、複数の RLM サーバーのリストを指定できます。その後サーバーは、対応する RLM サーバーと 1 つの接続が確立されるまで、各サーバーに対して順番に接続を試みます。これは、プライマリ RLM サーバーに到達できない場合に引き継ぐために RLM フェイルオーバーサーバーを使用している場合などに特に便利です。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*

> **Note**
>
> Amazon DCV サーバーが Windows にインストールされている場合は、指定のエントリを `;` で区切る必要があります。

トピック

- Windows Amazon DCV サーバーの設定
- Linux Amazon DCV サーバーの設定

Windows Amazon DCV サーバーの設定

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/] キーに移動して、[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。[値のデータ] に、RLM サーバーのポート番号とホスト名を *`RLM_server_port`*`@`*`RLM_server`* の形式で入力します。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 Amazon 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`* 形式で 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

Amazon DCV サーバーは、数分おきに RLM サーバ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ると、Amazon DCV サーバーは実行中のセッションで使用されるライセンスを自動的に更新します。次の手順では、RLM で DCV ライセンスを更新する方法を詳しく説明します。

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。
2. `C:\RLM\rlmutil.exe rlmreread` Windows または Linux `/opt/nice/rlm/rlmutil rlmreread` で を実行して、ライセンスファイルを強制的に再ロードします。

RLM サーバーでライセンスが更新されると、Amazon DCV サーバーが数分以内 (通常は 5 分以内) に新しいライセンスの使用状況を確認します。

Amazon DCV バージョン 2021.0 以降では、管理者が次のコマンドを使用して、ライセンスを直ちに強制更新できます。

```
$ dcv reload-licenses
```

ステップ 3: Amazon DCV サーバーのイメージングを設定する (オプション)

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 (AMI) としてキャプチャできます。この機能を使用すると、1 つの AMI から複数のインスタンスを起動できます。すべてのインスタンスは、必要に応じて同じ設定で起動できます。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、Amazon DCV をオペレーティングシステムに追加できます。Amazon 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 構成を設定したりできます。

例えば、単一の AMI から複数の Amazon EC2 インスタンスをデプロイする場合、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、Amazon DCV アクセス許可をエンドユーザーに委任

できます。または、Amazon DCV セッションマネージャーなどのブローカーを使用して、Amazon 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。

Amazon DCV AMI の作成は、次のいずれかの方法で実行できます。

Amazon DCV イメージの構築

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

または、Windows または Linux で Amazon DCV の前提条件が満たされている場合は、Amazon が管理する Image Builder Amazon DCV コンポーネントを実行して、Amazon DCV をインストールおよび設定できます。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. [フィルターの所有者] ドロップダウンメニューを選択し、[クイックスタート (Amazon 管理)] 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 `dcv-server-linux` を検索します。
4. コンポーネントのハイパーリンクを選択します。
5. Amazon 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の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およびエグゼキューター (AWSTOE) を使用して、インスタンスでコンポーネントをローカルで実行します。

> **Note**
>
> 詳細については、「AWSTOE の開始方法」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。

Amazon DCV をイメージパイプラインに追加する

EC2 Image Builder レシピでは、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

する一連のコンポーネントを定義します。このレシピ内で、`dcv-server-windows` または `dcv-server-linux` コンポーネントを選択して、パイプライン内の Amazon DCV のインストールを自動化します。これらのコンポーネントのいずれかを選択するときは、要件を満たすようにパラメータをファインチューニングできます。

> **Note**
>
> Linux では、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 または前述の Image Builder コンポーネントで行うことができます。

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、「Amazon 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 Amazon DCV アクセス許可を設定します。詳細については、「DCV 管理ガイド」の「アクセス許可ファイルでの作業」を参照してください。

リナックス

- `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、「Amazon 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 Amazon 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 Amazon DCV パッケージがインストールされます。詳細については、「Amazon DCV 管理ガイド」の「Linux に Amazon DCV サーバーをインストールする」を参照してください。

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。

Amazon DCV サーバーの管理

Amazon DCV は、クライアントがリモートデスクトップにアクセスするためのユーザーセッションを作成する専有サーバーで実行されます。管理者は、登録されている Amazon DCV サーバーとそのホストされたセッションを管理および制御できます。

この管理にアクセスするには、管理者認証情報が必要です。Amazon DCV サーバーの起動、停止、または設定を行うには、管理者 (Windows) またはルート (Linux) としてサインインする必要があります。

トピック

- Amazon DCV サーバーの起動
- Amazon DCV サーバーの停止
- Amazon DCV サーバーのアップグレード
- Amazon DCV サーバーのアンインストール
- QUIC UDP トランスポートプロトコルを無効にする
- Amazon DCV サーバーの 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 Amazon DCV サーバーでの GPU 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 Amazon DCV サーバーでのプリンターの設定
- Linux Amazon 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
- Amazon DCV 認証の設定

- Amazon DCV 認可の設定
- 仮想セッションで X Server へのリモート X 接続を有効にする
- Amazon DCV ウェブブラウザクライアントを iFrame 内に埋め込む

Amazon DCV サーバーの起動

セッションをホストするには、Amazon 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、Amazon DCV サーバーは、ホストされているサーバーのスタートアップ時には必ず起動します。Amazon DCV サーバーのインストール時に自動スタートアップを無効にした場合、サーバーを手動で起動するか、自動スタートアップを再び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 Amazon 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 Amazon DCV サーバーを手動で起動するには、以下の手順に従います。

Windows で Amazon DCV サーバーを起動する方法

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 Server] (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。

Amazon DCV サーバーが Windows で自動的に起動するように設定する方法

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 Server] (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 Amazon DCV server

コマンドラインを使用して Amazon DCV サーバーを手動で起動します。

Linux で Amazon DCV サーバーを起動する方法

次のコマンドを使用します。

- RHEL、CentOS、SUSE Linux Enterprise 12、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 Amazon DCV サーバーを設定します。

Amazon DCV サーバーが Linux で自動的に起動するように設定する方法

次のコマンドを使用します。

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

```
$ sudo systemctl enable dcvserver
```

Amazon DCV サーバーの停止

Amazon 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、アクティブな Amazon DCV セッションがすべて終了します。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 Amazon 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 Amazon DCV サーバーを手動で停止するには、以下の手順に従います。

Windows で Amazon DCV サーバーを停止する方法

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 Server] (DCV サーバー) を開きます。
3. [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 Amazon DCV サーバーが自動的に起動されないようにする方法

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 Server] (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 Amazon DCV server

コマンドラインを使用して Amazon DCV サーバーを停止します。

Linux で Amazon DCV サーバーを停止する方法

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する Amazon DCV サーバーの自動スタートアップを無効にします。

Linux で Amazon DCV サーバーが自動的に起動されないようにする方法

以下のコマンドを使用します。

```
$ sudo systemctl disable dcvserver
```

Amazon DCV サーバーのアップグレード

次のトピックでは、Amazon 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 Amazon DCV サーバーのアップグレード
- Linux での Amazon DCV サーバーのアップグレード

互換性に関する考慮事項

Amazon DCV サーバーバージョン 2017 以降は、Amazon 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**
>
> オンプレミスサーバーおよび EC2 ベース以外のサーバーに対する Amazon DCV サーバーのライセンス互換性要件については、「ライセンス要件」を参照してください。

Windows での Amazon DCV サーバーのアップグレード

Windows で Amazon DCV サーバーをアップグレードする方法

1. RDP クライアントを使用して、管理者として Amazon DCV サーバーに接続します。
2. 実行中の Amazon DCV セッションがないことを確認します。実行中のセッションをチェックするには、`dcv list-sessions` Amazon DCV コマンドを使用します。実行中のセッションがある場合は、`dcv close-session` Amazon 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、Amazon DCV サーバーを停止します。詳細については、「Amazon DCV サーバーの停止」を参照してください。
4. Amazon 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. Amazon DCV サーバーの最新バージョンを NICE 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、その時点でも Amazon DCV サーバーの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、ステップ 4 でエクスポートした設定とパラメータを比較します。
8. 新しい Amazon DCV セッションを開始して、Amazon DCV サーバーをテストします。詳細については、「Amazon DCV セッションの開始」を参照してください。

Linux での Amazon DCV サーバーのアップグレード

Linux で Amazon DCV サーバーをアップグレードする方法

1. `root` ユーザーを使用して、SSH でサーバーにサインインします。
2. 実行中の Amazon DCV セッションがないことを確認します。実行中のセッションをチェックするには、`dcv list-sessions` Amazon DCV コマンドを使用します。実行中のセッションがある場合は、`dcv close-session` Amazon 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、Amazon DCV サーバーを停止します。詳細については、「Amazon DCV サーバーの停止」を参照してください。
4. Amazon 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. Amazon 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、その時点でも Amazon DCV サーバーの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 Amazon DCV セッションを開始して、Amazon DCV サーバーをテストします。詳細については、「Amazon DCV セッションの開始」を参照してください。

Amazon DCV サーバーのアンインストール

次のトピックでは、Amazon 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 Amazon DCV サーバーのアンインストール
- Linux での Amazon DCV サーバーのアンインストール

Windows での Amazon DCV サーバーのアンインストール

Windows で Amazon DCV サーバーをアンインストールする方法

1. RDP クライアントを使用して、管理者として Amazon DCV サーバーに接続します。
2. 実行中の Amazon DCV セッションがないことを確認します。実行中のセッションをチェックするには、`dcv list-sessions` Amazon DCV コマンドを使用します。実行中のセッションがある場合は、`dcv close session` Amazon 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、Amazon DCV サーバーを停止します。詳細については、「Amazon 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. Amazon 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) Amazon DCV サーバ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\] に移動して [log] (ログ) フォルダを削除します。

Linux での Amazon DCV サーバーのアンインストール

Amazon 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 パッケージまたは .deb パッケージを使用してインストールされます。

> **Note**
>
> Amazon 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 Amazon DCV サーバーをアンインストールする方法

1. 実行中の Amazon DCV セッションがないことを確認します。実行中のセッションをチェックするには、`dcv list-sessions` Amazon DCV コマンドを使用します。実行中のセッションがある場合は、`dcv close session` Amazon 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、Amazon DCV サーバーを停止します。詳細については、「Amazon DCV サーバーの停止」を参照してください。
3. Amazon 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、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 Amazon 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) Amazon DCV サーバ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。

QUIC UDP トランスポートプロトコルを無効にする

デフォルトでは、バージョン 2024.0 以降、Amazon DCV は TCP に基づく WebSocket プロトコルと、データ転送用の UDP に基づく QUIC プロトコルの両方をサポートしています。

この QUIC トランスポートプロトコルは UDP に基づいています。ネットワークでレイテンシーとパケット損失が頻発する場合、QUIC を使用するとパフォーマンスが向上する可能性があります。QUIC では、サーバーは認証トラフィックに WebSocket を使用し続けます。

> **Note**
> ネットワークとセキュリティの設定で UDP トラフィックが許可されている場合のみ、QUIC を使用できます。

QUIC を有効にすると、クライアントでは、Amazon DCV サーバーセッションへの接続時に QUIC プロトコルを使用してデータを転送できます。クライアントの接続時に QUIC プロトコルが使用されない場合は WebSocket が使用されます。QUIC プロトコルの詳細については、「Amazon DCV ユーザーガイド」の「Amazon DCV セッションへの接続」を参照してください。

Windows Amazon DCV server

Amazon DCV でのデータ転送に QUIC (UDP) の使用を無効にするには

1. Windows レジストリエディタを開き、HKEY_USERS/S-1-5-18/Software/GSettings/com/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 `0` を入力します。

> **Note**
> この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て `enable-quic-frontend` という名前を付けます。

3. Windows レジストリエディタを閉じます。
4. Amazon DCV サーバーを停止して再起動します。

Linux Amazon DCV server

Amazon DCV でのデータ転送に QUIC (UDP) の使用を無効にするには

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、`false` を指定します。

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

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

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

デフォルトでは、Amazon DCV サーバーは TCP ポート 8443 をリッスンし、実行中の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。

Amazon DCV サーバーをインストールした後、カスタム TCP ポートを指定できます。QUIC を有効にするように Amazon DCV サーバーを設定した場合は、QUIC トラフィック用のカスタム 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。

Amazon 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 のみを使用するか、IPv6 のみを使用するかを指定できます。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**
>
> Amazon 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てください。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として、Amazon DCV Connection Gateway または別のウェブプロキシや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方法があります。ゲートウェイ経由で Amazon DCV サーバーホストにアクセスすることで、サーバーに単一のアドレスを割り当てることができます。また、HTTPS の標準ポート番号である 443 を含む 1024 未満のポート番号を使用することもできます。
> 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

- Amazon DCV サーバーの TCP/UDP ポートの変更
- 特定のエンドポイントのリスニング

Amazon DCV サーバーの TCP/UDP ポートの変更

Windows Amazon DCV server

Amazon DCV サーバーにより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/GSettings/com/nicesoftware/dcv/connectivity/] キーに移動します。
3. TCP ポートを設定するには、web-port 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、[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 `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。[値のデータ] 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、Amazon DCV サーバーではデフォルトで TCP ポート 8443 が使用されます。

   > **Note**
   > TCP ポート番号は 1024 より大きくする必要があります。

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

   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、[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 `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。[値のデータ] 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しないで QUIC サポートを有効にした場合、Amazon DCV サーバーではデフォルトで UDP ポート 8443 が使用されます。

   > **Note**
   > UDP ポート番号は 1024 より大きくする必要があります。

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

Linux Amazon DCV server

Amazon DCV サーバーで使用されるポートを変更するには、`dcv.conf` ファイルの `web-port` および `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. Amazon 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、Amazon 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 `:` で区切られたポート番号を追加できます。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 上の任意のローカル IPv4 アドレスと任意のローカル IPv6 アドレスを表し、`'[::%1]:8443'` は Windows ホストのインデックス 1 のネットワークインターフェイスにバインドされている IPv6 アドレスを表し、`'[::%eth1]:8443'` は Linux ホストの `eth1` ネットワークインターフェイスにバインドされている IPv6 アドレスを表します。

> **Note**
>
> これらの設定パラメータは、Amazon DCV サーバー 2022.0 以降でのみ使用できます。

Windows Amazon 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 パラメーターを選択します。

   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. Amazon DCV サーバーを停止して再起動します。

Linux Amazon 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. Amazon DCV サーバーを停止して再起動します。

TLS 証明書の管理

Amazon DCV では、Amazon DCV クライアントと Amazon DCV サーバー間のトラフィックの保護に使用する自己署名証明書が自動的に生成されます。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、サーバーの証明書に関する警告が表示される場合があり、ユーザーは接続が確立される前に検証できます。

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て証明書の信頼性を確認するように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、特定の証明書のフィンガープリントを Amazon DCV サーバー管理者に確認するように勧められる場合があります。

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 Amazon DCV 証明書とキーは、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 Amazon DCV server

Windows でサーバーの TLS 証明書を変更する方法

- Windows Amazon DCV サーバーの以下の場所に証明書とキーを配置してください。

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

Linux Amazon DCV server

Linux でサーバーの TLS 証明書を変更する方法

1. Linux Amazon DCV サーバーの以下の場所に証明書とキーを配置してください。

```
/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
```

> **Note**
>
> Amazon DCV 2022.0 以降では、Amazon DCV サーバーの実行中に証明書ファイルを更新すると、新しい証明書が自動的に再ロードされます。それより前のバージョンの Amazon DCV では、Amazon 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が切断されるように Amazon DCV を設定することができます。具体的には、Amazon DCV サーバーにキーボードまたはポインタの入力を一定期間送信しなかったクライアントを切断できます。デフォルトでは、アイドル状態に入って 60 分間 (1 時間) が経過した Amazon DCV クライアントは Amazon DCV サーバーにより切断されます。

特定のアクションにより、アイドル状態の切断タイムアウト期間がリセットされます。次のいずれかのアクションを行うと、アイドルタイムアウト期間が設定された時間枠にリセットされます。

- マウスを移動する
- マウスボタンを押すか、マウスホイールを移動する

- 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)
- スタイラスを使用する (有効な場合)
- ゲームパッドを使用する (有効な場合)
- ウェブカメラを使用してストリーミングする (有効な場合)
- 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に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。

通知をアイドル状態のクライアントに送信するように Amazon DCV サーバーを設定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、Amazon DCV サーバーおよびクライアントのバージョン 2017.4 以降でのみサポートされます。

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 Amazon DCV server

Amazon 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/] キーに移動して、[idle-timeout] 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、[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 `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) タイムアウト通知をアイドル状態のクライアントに送信するように Amazon DCV サーバーを設定する方法

1. [HKEY_USERS/S-1-5-18/Software/GSettings/com/nicesoftware/dcv/connectivity/] キーに移動して、[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、[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 `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 Amazon DCV server

Amazon 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` ファイルの `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) タイムアウト通知をアイドル状態のクライアントに送信するように Amazon DCV サーバーを設定する方法

1. 任意のテキストエディタで /etc/dcv/dcv.conf を開きます。
2. 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 Amazon DCV サーバーでの GPU 共有の有効化

GPU 共有を使えば、複数の Amazon DCV 仮想セッション間で 1 つ以上の物理的な GPU を共有できるようになります。セッションの詳細については、「Amazon DCV セッションの管理」を参照してください。GPU 共有を使えば、単一の Amazon DCV サーバーを使用して、サーバーの物理的な GPU リソースを共有する複数の仮想セッションをホストすることができるようになります。

> **Note**
> GPU 共有は Linux Amazon DCV サーバーでのみサポートされます。

前提条件

開始する前に、次の前提条件を完了します。

- Linux サーバーに Amazon DCV サーバーをインストールします。
- サーバーに Amazon DCV dcv-gl および nice-Xdcv パッケージをインストールします。
- サーバーに少なくとも 1 つのサポートされている NVIDIA GPU があることを確認します。

- サーバーに NVIDIA GPU ドライバーをインストールします。公式の NVIDIA ドライバーが必要です。オープンソースの 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加速化 OpenGL をサポートしていることを確認してください。
- 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、自動的に `xorg.conf` ファイルを作成し、すべての利用可能な NVIDIA GPU 用に設定することができます。

- 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) `dcvgldiag` ツールを実行して、Amazon DCV サーバー設定を検証します。詳細については、「インストール後のチェックの実行」を参照してください。

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

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

このトピックで前述した前提条件を完了した後、`dcv.conf` ファイルの `gl-displays` パラメータで GPU を指定します。

Linux Amazon DCV サーバーで GPU 共有を有効にする方法

1. `/etc/dcv/` に移動し、任意のテキストエディタで `dcv.conf` 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 を以下の形式で指定します。

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

   ここで、*`xserver_port`* はサーバーで、*`screen_number`* は GPU と連携している画面に関連付けられている番号です。*`screen_number`* は `0` から始まります。

以下の例に示しているのは、デフォルトの X サーバーセッションで実行されている 2 つの GPU の `gl-displays` パラメータです。

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

3. ファイルを保存して閉じます。
4. Amazon 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**
>
> タッチスクリーンやスタイラスデバイスの USB リダイレクトは不要です。また、Amazon 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もありません。
> Amazon DCV は、ネイティブオペレーティングシステムの API 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。
> 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。Windows Amazon DCV サーバーでは、この機能はデフォルト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。Linux Amazon DCV サーバー上でホストされている仮想セッションでは、この機能はデフォルトで有効になっています。ただし、Linux Amazon 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**
>
> Amazon DCV でのタッチスクリーンとスタイラスの使用は、クライアントとサーバーの両方でサポートされ、サーバーでも有効になっている場合に有効になります。クライアントサポートについては、「Amazon DCV ユーザーガイド」の「クライアントの機能」を参照してください。

Linux Amazon 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"
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 スタイラスペン、DCV スタイラス消しゴム、および 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)]
   |   | DCV Touchscreen                           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

Amazon DCV サーバー 2022.0 以降、ゲームパッドデバイスは、サポート対象の Windows または Linux オペレーティングシステムのいずれかに接続されている場合に使用できます。

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 のコントローラー

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 ネイティブの Amazon 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。2022.0 以降の Amazon DCV クライアントを使用してください。

ゲームパッドのサポートを有効化するには、Amazon DCV サーバーの最新バージョンがインストールされていること、ゲームパッド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 Amazon DCV サーバーのインストール」を参照してください。ドライバーがインストールされると、その機能は Windows Amazon DCV サーバーでデフォルトで有効化されます。

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

Amazon DCV を使用すると、クライアントでさまざまな専用 USB デバイス (例: 3D ポインティングデバイス、認証デバイス) を使用できます。これらのデバイスは、Amazon DCV サーバーで実行されているアプリケーションとの相互通信を目的としてコンピュータに物理的に接続されます。

> **Important**
>
> Amazon 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。Amazon DCV クライアントでは、追加設定が必要になる場合があります。USB リモート化をクライアントにインストールする方法については、「Amazon 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」のオプション手順を参照してください。

Amazon DCV サーバーでは、許可リストを使用して、使用を許可する USB デバイスクライアントを決定します。デフォルトで、一般的に使用されている一部の USB デバイスが許可リストに追加されています。これにより、クライアントのコンピュータにこれらの USB デバイスを接続して、サーバーで使用できるようになります。追加設定は必要ありません。詳細については、「Amazon DCV ユーザーガイド」の「USB 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、クライアントに使用される前に、Amazon DCV サーバー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 Amazon DCV server

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

USB デバイスを Windows Amazon DCV サーバーの許可リストに追加する方法

1. Amazon DCV サーバーの最新バージョンがインストールされていること、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 Amazon 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 Amazon 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. Amazon DCV サーバーを停止して再起動します。

Linux Amazon DCV server

USB デバイスを許可リストに追加するには、USB デバイスのフィルター文字列を `usb-devices.conf` ファイルに追加する必要があります。

Linux Amazon DCV サーバーの許可リストへの USB デバイスの追加

1. 最新バージョンの Amazon DCV サーバーと DCV USB ドライバーがインストールされていることを確認してください。詳細については、「Linux への Amazon 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 Amazon 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. Amazon 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

スマートカードのキャッシュ機能を使用すると、Amazon DCV サーバーでスマートカード値をキャッシュできるようになります。この機能を有効にすると、Amazon DCV サーバーによって、クライアントのスマートカードへの最新の呼び出しの結果がキャッシュされます。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、「Amazon DCV ユーザーガイド」の「スマートカードの使用」を参照してください。または、Amazon DCV サーバーを設定して、DCV_PCSC_ENABLE_CACHE 環境変数に対して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化または無効化することができます。

Linux Amazon DCV server

Linux Amazon DCV サーバーでスマートカードのキャッシュを永続的に有効化または無効化する方法

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. 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. Amazon DCV サーバーを停止して再起動します。

Windows Amazon DCV server

Windows Amazon DCV サーバーでスマートカードのキャッシュを永続的に有効化または無効化する方法

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/smartcard/] キーに移動して、[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

Amazon DCV サーバー 2023.1 以降、ユーザーはサポート対象のブラウザでのリモートセッション中に、ウェブ認証 (WebAuthn) 標準を使用するウェブアプリケーションに対して認証できるようになりました。これは、Windows Hello や YubiKey などのローカルに接続された FIDO2 認証システムや、その他の FIDO2 準拠認証システムに認証プロンプトを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 リダイレクトから独立して動作します。Amazon DCV サーバーにベンダー固有のドライバーを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、ブラウザのネイティブ API によって行われます。

WebAuthn を使用する前に、「サポートされている機能」の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 リダイレクションは、`webauthn-redirection` 権限を使用して有効化または無効化できます。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。

WebAuthn リダイレクトを使用するには、リモートサーバーにブラウザ拡張機能をインストールする必要があります。機能を有効にしてブラウザ拡張機能をインストールすると、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セッション内で開始された 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 YubiKey などのデバイスを使用して認証を完了できます。

> **Note**
> この機能により、リモートセッション中にブラウザ内で WebAuthn を使用できますが、WebAuthn 認証システムを使用した DCV セッション認証はサポートされません。

WebAuthn リダイレクトのブラウザ拡張機能の設定

初回ブラウザ起動時の自動プロンプト

WebAuthn リダイレクトを有効にして Amazon DCV サーバー 2023.1 をインストールすると、ユーザーは初めてブラウザを起動した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か、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インストール

拡張機能をより広範囲に展開したいと考えている組織は、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 (gpmc.msc) を起動します。
3. [フォレスト] > [ドメイン] > [FQDN (例:example.com)] > [グループポリシーオブジェクト] の順に移動します。
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。

```
{"mmiioagbgnbojdbcjoddlefhmcocfpmn":
{ "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. Amazon 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に移動します (上記のリンク)。
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す。

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 リダイレクト拡張機能の実行を具体的に許可する必要があります。許可しない場合、WebAuthn リダイレクトは実行されません。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、[シークレットモードでの実行を許可する] を見つけます。
3. スイッチを [オン] にします。

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 Amazon DCV セッションに接続されているときにアクセスできる、Amazon DCV サーバーにあるフォルダです。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 Amazon DCV サーバーでのセッションストレージの有効化
- Linux Amazon DCV サーバーでのセッションストレージの有効化

Windows Amazon 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、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 キーに移動して、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、session-management/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. Amazon 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、「Amazon DCV セッションの開始」を参照してください。

Linux Amazon 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. Amazon 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、「Amazon DCV セッションの開始」を参照してください。

Linux Amazon DCV サーバーでのプリンターの設定

Amazon DCV では、ローカルにリダイレクトされたプリンターまたは仮想 Amazon DCV プリンターに印刷できます。

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 Amazon DCV サーバーを設定する必要があります。

Windows Amazon 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 Amazon DCV サーバーでプリンターのリダイレクトを有効にする方法

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、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が cups ローカル証明書を読み取る許可を取得してい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. Amazon 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 と RHEL 8 がプリンターソケットへの接続を妨げる可能性があります。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 Amazon 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、ポインター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 Amazon DCV サーバーでは、クライアントとサーバー間でコピーと貼り付け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. Amazon 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。

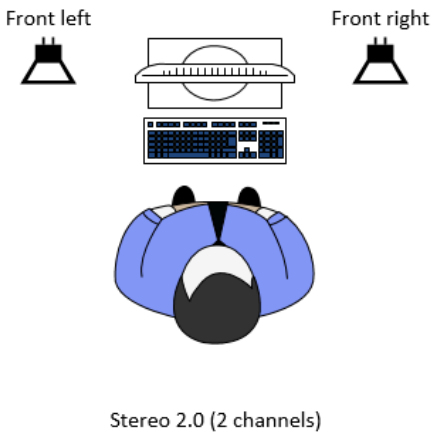
   ```
   [clipboard]
   primary-selection-copy=true
   ```
3. ファイルを保存して閉じます。
4. Amazon DCV セッションを停止して再起動します。

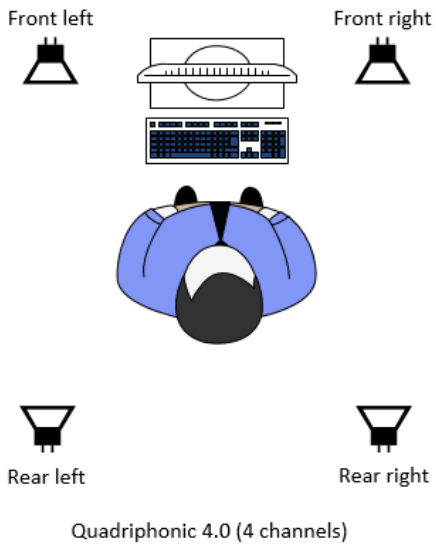
マルチチャンネルオーディオの設定

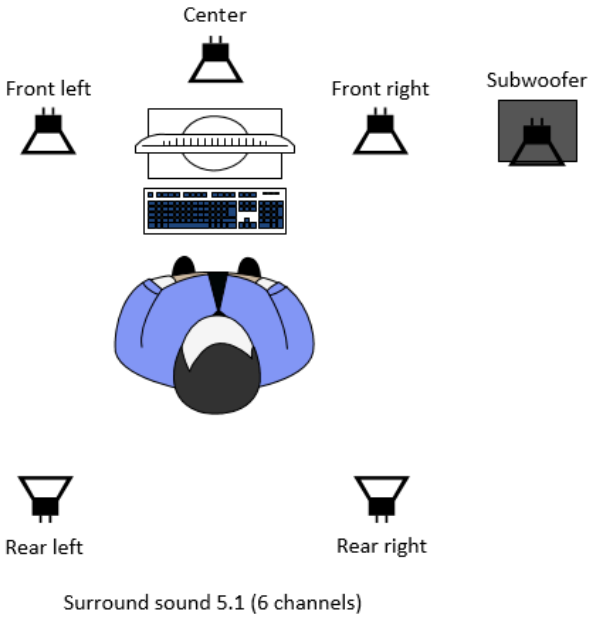
Amazon DCV では、Amazon DCV ネイティブクライアントを使用している場合、最大 7.1 のオーディオチャンネルがサポートされます。ウェブブラウザクライアントでは、ステレオ 2.0 オーディオチャンネルのみがサポートされます。

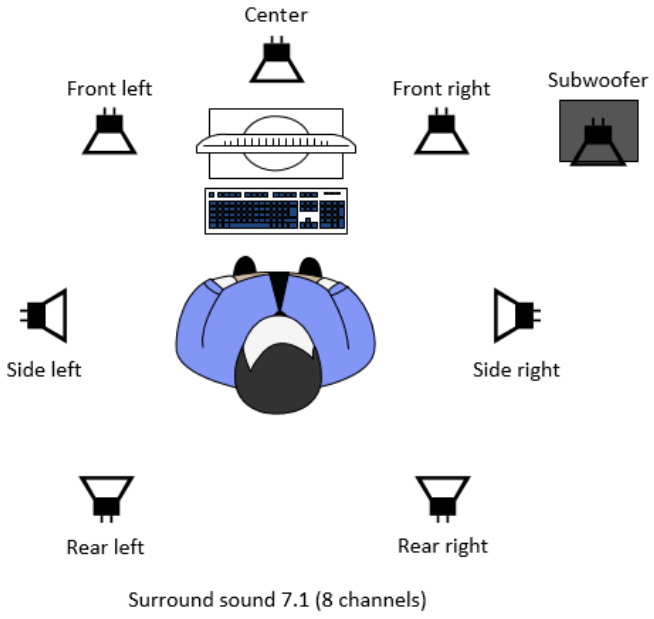
Amazon DCV では、次のマルチチャンネルオーディオ設定がサポートされています。

- 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 チャネル) — Windows Amazon DCV サーバーのみ









サーバーで提供されるチャネル数よりも少ない数のオーディオチャ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。これに該当する場合は、オーディオチャンネルが失われるのを防ぐため、オーディオチャンネルがダウンミックスされるように Amazon DCV サーバーを設定してください。

トピック

- Windows Amazon DCV サーバーでのオーディオチャンネルの設定
- Linux Amazon DCV サーバーでのオーディオチャンネルの設定

Windows Amazon 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、Amazon 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] を選択してください。

Linux Amazon 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 Amazon 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方法

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 サウンドサーバーからオーディオがキャプチャされるように Amazon DCV サーバーを設定する必要があります。

PulseAudio デバイスが使用されるように Amazon DCV サーバーを設定します。

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 サウンドサーバーからオーディオをキャプチャするように Amazon DCV サーバーを設定したら、使用するチャネル数を指定できます。

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. Amazon 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クライアントを使用してセッションに接続するときに、追加の HTTP レスポンスヘッダーが Amazon DCV クライアントに送信されるように Amazon DCV サーバーを設定できます。レスポンスヘッダーで、ユーザーが接続している Amazon DCV サーバーに関する追加情報を提供できます。

トピック

- Windows Amazon DCV サーバーでの HTTP ヘッダーの設定
- Linux Amazon DCV サーバーでの HTTP ヘッダーの設定

Windows Amazon DCV サーバーでの 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。[Value data] (値のデータ) に、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. Amazon DCV サーバーを停止して再起動します。

Linux Amazon 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` ファイルの `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 ヘッダーの名前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. Amazon DCV サーバーを停止して再起動します。

Amazon DCV 認証の設定

デフォルトでは、クライアントは、Amazon DCV セッションに接続する前に Amazon DCV がホストされているサーバーに対して認証を行うよう求められます。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。

Amazon DCV では、次の認証方法がサポートされています。

- `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 Amazon DCV サーバーの場合、認証は WinLogon に委任されます。Linux Amazon DCV サーバーの場合、認証は PAM に委任されます。クライアントが Amazon DCV セッションに接続されるときに、クライアントからシステム認証情報が提供されます。クライアントに Amazon DCV サーバーの適切なサインイン認証情報があることを確認してください。
- none — Amazon DCV セッションへの接続時にクライアント認証は不要です。Amazon DCV サーバーから、セッションに接続するすべてのクライアントに対して、自動的にアクセス権が付与されます。

Amazon DCV サーバーによって使用される認証方法がクライアントに認識されていることを確認してください。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
- 外部認証を使用する

Windows での認証の設定

Amazon 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/] キーに移動して、[authentication] 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、ユーザーが DCV サーバーに対して認証できた場合に、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、DCV `authentication` パラメータが `system` に設定されている場合にのみサポートされます。DCV `authentication` パラメータが none に設定されている場合、ユーザーは DCV サーバに対して自動的に認証された後、Windows に手動でサインインする必要があります。

Windows 認証情報プロバイダーは、デフォルトで、Amazon DCV サーバーのインストール時に有効になります。

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

Amazon DCV サーバーの認証方法を変更するには、dcv.conf ファイルの 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、Amazon DCV authentication パラメータが system に設定されている場合、PAM サービスを実行することによって認証が実施されます。

デフォルトでは、Amazon DCV サーバーによって実行される特権的アクセス管理 (PAM) サービスは /etc/pam.d/dcv になります。

Amazon DCV を通じてユーザー認証を行う際に PAM で実行されるステップに変更を加えたい場合、dcv.conf の authentication セクションで pam-service パラメータを設定できます。

PAM サービスを変更する方法

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、お使いのシステムのマニュアル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、/etc/pam.d で作成したファイルの名前と一致していなければなりません。

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、外部認証システム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。

外部認証を使用する

デフォルトでは、Amazon 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 Amazon DCV サーバーの場合、認証は WinLogon に委譲されます。Linux Amazon DCV サーバーの場合、認証は Linux 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 Amazon 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、Amazon 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 Amazon 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 Amazon DCV サーバーから受信し、必要な認証を実行して、応答を Amazon 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。
- Amazon DCV サーバー設定 — Amazon 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、「Amazon DCV サーバー設定」を参照してください。

トピック

- Amazon DCV サーバー設定
- トークンの使用
- 認証サービス要件

Amazon DCV サーバー設定

外部認証サービスが使用されるように Amazon DCV サーバーを設定する必要があります。

Linux Amazon 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、DcvSimpleExternalAuthenticator を使用している場合は *`http://127.0.0.1`*`:`*`8444`* を指定します。

   `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 Amazon 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. セキュリティパラメータで auth-token-Verifier パラメータを探します。
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、DcvSimpleExternalAuthenticator を使用している場合は *http://127.0.0.1*:*8444* を指定します。
   - セキュリティセクションに auth-token-verifier パラメータが存在しない場合は、PowerShell に追加してください。「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. Amazon 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 Amazon 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、Amazon 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、Amazon 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。

Amazon 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が判断されます。

トークンの検証後、認証サーバーから Amazon 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

Amazon 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の作成に開始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、Amazon DCV サーバーがインストールされているサーバーと同じサーバーで実行される必要があります。デフォルトでは、DcvSimpleExternalAuthenticator はポート 8444 のリクエストをリッスンします。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。

DcvSimpleExternalAuthenticator を使用するには、`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、「Amazon 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。

   ```
   echo "123456" | sudo dcvsimpleextauth add-user --session session-123 --
   auth-dir /var/run/dcvsimpleextauth/ --user username --append
   ```

3. サーバーを起動します。

   ```
   sudo dcvsimpleextauth --port 8444 --auth-dir /var/run/dcvsimpleextauth/
   start-server
   ```

4. サーバーが起動したら、構成をテストして検証します。

   Example

   ここでも、この例を使用した場合、テストは次のように実行されます。

   ```
   curl -k http://localhost:8444 -d sessionId=session-123 -d
   authenticationToken=123456
   ```

   成功した場合は yes の認証結果が返されます。

Amazon DCV 認可の設定

認可は、特定の Amazon DCV 機能に対して、Amazon 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。Amazon DCV では、認可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、セッションに接続するときに特定のユーザーが利用できる特定の Amazon DCV 機能を定義するものです。

Amazon DCV では、次の 2 種類のアクセス許可ファイルがサポートされています。

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について

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 サーバー上の `C:\Program Files\NICE\DCV\Server\conf\default.perm`、および Linux NICE DCV サーバー上の `/etc/dcv/default.perm` 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について」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すると、特定のユーザーまたはグループが Amazon DCV セッションに接続するときにアクセスす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、「Amazon 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について」を参照してください。

アクセス許可ファイルについて

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。

トピック

- アクセス許可ファイルのインポート
- グループの作成
- エイリアスの作成
- アクセス許可を追加

アクセス許可ファイルのインポート

`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。これにより、以前に定義された Amazon 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ができます。

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 Amazon DCV サーバー:

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

  - Linux Amazon 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。これは Windows Amazon DCV サーバーの Amazon DCV インストールフォルダ内にあります。

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

グループの作成

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ユーザーグループをグループに追加する方法 (Linux Amazon 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
```

エイリアスの作成

アクセス許可ファイルの [aliases] セクションでは、Amazon DCV 機能のセットを作成することができます。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 Amazon DCV 機能やその他のエイリアスです。

```
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

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。Amazon DCV には、セッション所有者の参照に使用される組み込み %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 Amazon DCV 機能、エイリアス、または両方の組み合わせが含まれる場合があります。機能のリストはスペースで区切る必要があります。Amazon DCV には、すべての Amazon DCV 機能を含む組み込み builtin エイリアスが含まれます。

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` — Amazon DCV サーバーにクライアントからオーディオを入力します。
- `audio-out` — クライアントで Amazon 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` — Amazon 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のデータを Amazon DCV サーバーに貼り付けます。
- `display` — Amazon DCV サーバーからビジュアルデータを受信します。
- `extensions-client` — Amazon DCV クライアントにインストールされた拡張機能を起動できます。
- `extensions-server` — Amazon DCV サーバーにインストールされた拡張機能を起動できます。
- `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 Amazon DCV 2022.0 以降でサポートされています。
- `keyboard` — Amazon DCV サーバーにクライアントキーボードから入力します。
- `keyboard-sas` — セキュア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 Amazon DCV 2017.3 以降でサポートされています。
- `mouse` — Amazon DCV サーバーにクライアントポインタから入力します。
- `pointer` — Amazon DCV サーバーのマウス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 Amazon DCV 2017.3 以降でサポートされています。
- `printer` — Amazon DCV サーバーからクライアントに対して PDF ファイルまたは 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 Amazon DCV 2021.2 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、Amazon DCV クライアントウィンドウで Windows Snipping Tool 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` — 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) から入力します。
- `touch` — ネイティブタッチイベントを使用します。バージョン DCV 2017.3 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 Amazon DCV 2021.0 以降でサポートされています。
- `webauthn-redirection`— 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 Amazon DCV 2023.1 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
```

仮想セッションで X Server 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 Amazon DCV に継承されます。Amazon 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 Server 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 Server 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. Amazon DCV セッションを接続します。
2. Amazon DCV セッションの X サーバーが 6000-6063 の範囲のポートでリッスンし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 X サーバーホストアクセスリストに追加します。

```
$ xhost +remote_server
```

4. Amazon 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、Amazon 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 Amazon DCV サーバーのデスクトップ環境に表示されます。

Amazon 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、Amazon 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 Amazon 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**
>
> Amazon DCV Connection Gateway 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。これは、ゲートウェイ設定の「`[web-resources]` セクション」内の `local-resources-http-headers` パラメータを使用して行われます。

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")]` 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-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. Amazon 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. Amazon 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>
```

Amazon DCV セッションの管理

Amazon DCV サーバーが設定され、すべてのアプリケーションがインストールされると、クライアントは安全なセッションを通じてアクセスします。クライアントでこれらのセッションを管理すると、アクセスが許可され、各セッションのパラメータが設定されます。

クライアントを Amazon DCV セッションに接続できるようにするには、Amazon 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 Amazon DCV サーバーに接続できます。

各 Amazon DCV セッションには次の属性があります。

- セッション ID — Amazon 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 Amazon 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、Amazon DCV クライアントはこの情報を必要とします。

トピック

- Amazon DCV セッションについて
- コマンドラインツールを使用したセッションの管理
- Amazon DCV セッションの開始
- Amazon DCV セッションの停止
- Amazon DCV セッションの表示
- アクティブな Amazon DCV セッションの管理
- セッションのタイムゾーンの設定
- Linux でのスクリーンブランキングの管理
- Amazon DCV セッションでのスクリーンショットの作成

Amazon DCV セッションについて

Amazon 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 Amazon 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ユーザーのみ	可	可。追加ソフトウェアなし
仮想	Linux Amazon 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 Amazon 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 Amazon DCV サーバーでサポートされています。Windows Amazon 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。

Amazon 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 Amazon DCV サーバーの管理者および Linux Amazon 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、Amazon 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、Amazon 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 Amazon DCV サーバーでのみサポートされます。

同一の Amazon 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 Amazon DCV ユーザーが行います。Amazon 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 Amazon 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、Amazon DCV では、X サーバーインスタンス `Xdcv` が起動され、X サーバー内でデスクトップ環境が実行されます。Amazon 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**
>
> Amazon DCV では各仮想セッションで独立した `Xdcv` 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 Amazon 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。

`dcv-gl` 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、Amazon 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、`dcv-gl.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セッションの管理

Amazon DCV サーバーには、Amazon 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 Amazon DCV サーバーでのコマンドラインツールの使用

Windows Amazon DCV サーバーでコマンドラインツールを使用するには、Amazon 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、Amazon 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。Amazon 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。

Amazon 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
```

Amazon 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. [新規] を選択し、Amazon 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 Amazon DCV サーバーでのコマンドラインツールの使用

Linux Amazon 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、使用できる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、`dcv` を呼び出すときに `--help` を使用して取得できます。各コマンドの使用方法の詳細については、使用状況の情報を希望するコマンドの後に `--help` を渡します。例: `dcv create-session --help`。

コマンド	説明
`create-session`	新しい DCV セッションを作成します。
`close-session`	アクティブな DCV セッションを閉じます。
`describe-session`	DCV セッションを説明します。
`list-sessions`	アクティブな DCV セッションを一覧表示します。
`list-connections`	DCV セッションのクライアント接続を一覧表示します。
`close-connection`	アクティブなクライアント接続を閉じます。
`get-screenshot`	DCV コンソールのスクリーンショットを取得します。
`set-display-layout`	アクティブな DCV セッションの表示レイアウトを設定します。
`set-name`	DCV セッションの名前を設定します。
`set-permissions`	アクティブな DCV セッションのアクセス許可を設定します。
`set-storage-root`	アクティブな DCV セッションのストレージルートを設定します。
`reload-licenses`	

コマンド	説明
	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します。
`get-config`	サーバー設定を取得します。
`list-endpoints`	DCV エンドポイントを一覧表示します。
`set-config`	サーバー構成を設定します。
`version`	DCV のバージョンを表示します。
`help`	ヘルプを表示します。

Amazon DCV セッションの開始

デフォルトを使用して Windows Amazon 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。

Amazon 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。Amazon 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 Amazon 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超えたとします。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、Amazon 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、「Amazon 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

Amazon 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 Amazon 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 Amazon 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 Amazon 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

- 構文
- オプション
- 例

構文

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 Amazon 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。

タイプ: 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 Amazon 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 Amazon 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。

タイプ: 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 Amazon 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 Amazon 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。

タイプ: 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 Amazon DCV サーバー

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

- Linux Amazon 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 Amazon 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 Amazon 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、Amazon 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 Amazon 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**
>
> Amazon DCV では自動仮想セッションはサポートされません。

Windows Amazon DCV server

Windows Amazon 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] の順に選択します。

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. Amazon DCV サーバーを停止して再起動します。

Linux Amazon DCV server

Linux Amazon 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. Amazon DCV サーバーを停止して再起動します。

Amazon 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 Amazon DCV サーバーでは管理者、Linux Amazon DCV サーバーではルートユーザーだけです。Linux Amazon 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 Amazon 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 Amazon 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
```

Amazon DCV セッションの表示

Windows Amazon DCV サーバーの管理者または Linux Amazon 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。Amazon 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の表示
- 特定のアクティブなセッションの表示

すべてのアクティブなセッションの表示

Windows または Linux の Amazon 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

- 構文
- Output

構文

```
dcv list-sessions
```

Output

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')
```

特定のアクティブなセッションの表示

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

- 構文
- Output

構文

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

Output

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
    }
  ]
}
```

アクティブな Amazon DCV セッションの管理

クライアントがログインして DCV セッションの使用を開始すると、セッションはアクティブと見なされます。セッションがアクティブである間も、特定の管理タスクを実行できます。このセクションでは、使用中のセッションでクライアントがパラメータを管理するのに必要な情報を提供します。

トピック

- Amazon DCV セッションストレージの管理
- Amazon DCV セッション認可の管理
- Amazon 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
- アイドル状態のセッションの検出と停止

Amazon 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 Amazon DCV サーバーにあるディレクトリで、Amazon 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。

Amazon 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サーバーでディレクトリを指定できます。Amazon 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

- 構文
- オプション
- 例

構文

```
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。

タイプ: 文字列

必須: はい

例

Windows Amazon 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 Amazon 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/
```

Amazon DCV セッション認可の管理

認可は、特定の Amazon DCV 機能に対して、Amazon 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 Amazon 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す。Amazon DCV 認可の詳細については、「Amazon DCV 認可の設定」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

- 構文
- オプション
- 例

構文

```
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について」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:\dcv\` ディレクトリにあります。

```
C:\> dcv set-permissions --session my-session --file c:\dcv\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
```

Amazon DCV セッション表示レイアウトの管理

実行中の Amazon 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に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、Amazon 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、Amazon 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。

Amazon DCV 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解像度は、デフォルトで 1920x1080 (web-client-max-head-resolution 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 4096x2160 (max-head-resolution から) に制限されています。

使用可能な解像度とモニターの数はサーバーの設定に依存す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、システム環境とドライバーを適切にセットアップするための「前提条件ガイド」に従ってください。

> **Note**
>
> ネイティブクライアントでは、最大 4 つのモニターを使用できます。
> ウェブクライアントでは、最大 2 つのモニターを使用できます。
> 高解像度または最大モニター数を超えるモニターは、どの設定でもサポートされていません。

トピック

- 表示レイアウトへのアクセス
- 表示レイアウトの設定
- 表示レイアウトの表示

表示レイアウトへのアクセス

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 Amazon 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 Amazon 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。

表示レイアウトの設定

実行中の Amazon 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。

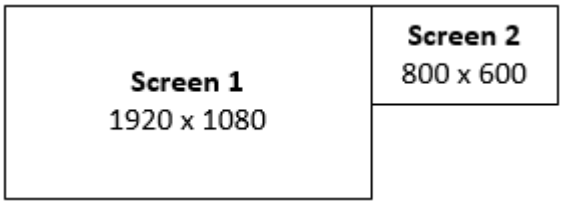
```
widthxheight+|-x-position-offset+|-y-position-offset
```

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。

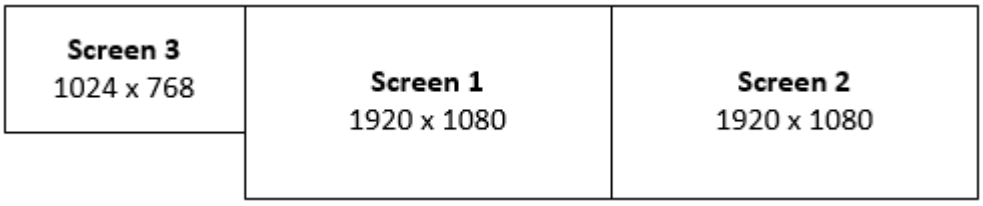
- 画面 1: 解像度 1920x1080、オフセット x=0、y=0
- 画面 2: 解像度 800x600、オフセット x=1920、y=0、画面 1の右側に表示されます。




```
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、「Amazon 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

- 構文

- オプション
- 例

構文

```
$ 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。

タイプ: 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
```

アイドル状態のセッションの検出と停止

アイドル状態の Amazon DCV セッションを識別するには、-j コマンドオプションがある dcv describe-sessions CLI コマンドを使用します。-j オプションを指定すると、コマンドが JSON 形式で出力を返すように設定され、セッションに関する追加の詳細情報が提供されます。

たとえば、次のコマンドは、my-session という名前のセッションに関する情報を返します。

```
$ dcv describe-session my-session -j
```

出力:

```
{
    "id" : "my-session",
    "owner" : "dcvuser",
    "x11-display" : ":1",
    "x11-authority" : "/run/user/1009/dcv/test3.xauth",
    "num-of-connections" : 1,
    "creation-time" : "2019-05-13T13:21:19.262883Z",
    "last-disconnection-time" : "2019-05-14T12:32:14.357567Z",
    "licensing-mode" : "DEMO",
    "licenses" : [
        {
            "product" : "dcv",
            "status" : "LICENSED",
            "check-timestamp" : "2019-05-14T12:35:40Z",
            "expiration-date" : "2019-05-29T00:00:00Z"
        },
        {
            "product" : "dcv-gl",
            "status" : "LICENSED",
```

```
            "check-timestamp" : "2019-05-14T12:35:40Z",
            "expiration-date" : "2019-05-29T00:00:00Z"
        }
    ]
}
```

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。

セッションのタイムゾーンの設定

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-redirection 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 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。

Linux でのスクリーンブランキングの管理

Linux サーバー上のコンソールセッションの場合、DCV は少なくとも 1 人のリモートユーザーがサーバーに接続されているときにデフォルトでローカル画面をブランクにし、最後のリモートユーザーの切断時に出力を復元 (画面もロック) します。

これにより、サーバーが物理的に近接しているユーザーが画面を表示し、ホストに接続された入力デバイスを使用してリモートセッションとやり取りすることができなくなります。これは、コンソールセッションをリモートで作業する場合、プライバシー上の理由や現地の法律のコンプライアンスのために望ましくない場合があります。ローカル入力は、キーボードやマウスなどの物理的に接続されたデバイスを無効にすることで防止されます。入力デバイスは、プラグインされるとすぐに、または有効であることがわかったときに無効になります。同様に、リモートユーザーが接続されていない場合、入力デバイスは再び有効になり、ローカルのログインとインタラクションが可能になります。

スクリーンブランキングと入力ブロックの無効化

スクリーンブランキングは、次の手順を使用して無効にできます。

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]` セクションで `disable-local-console parameter` を見つけます。ローカルに接続されたディスプレイをアクティブのままにして進行中のリモートセッションを

表示し、ローカルに接続されたデバイスを介したインタラクションを許可するには、disable-local-console=false を設定します。デフォルト値は true (スクリーンブランキングと入力ブロックがアクティブなど) です。[display/linux] セクションに disable-local-console 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。

```
[display/linux]
disable-local-console=false|true
```

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

選択した入力デバイスの無効化の防止

特定のデバイスを有効にしたままに設定できます。

名前が DCV で始まる入力デバイス (名前の後のスペースに注意してください) は、display-local-console 設定の値に関係なく無効にされることはありません。入力デバイスの名前を変更するには、「スタイラスを有効にする」のガイドを参照してください。

Amazon DCV セッションでのスクリーンショットの作成

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作成できます。

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、Amazon DCV で JPEG イメージファイルが作成されます。

タイプ: 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、Amazon 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
```

Amazon DCV のトラブルシューティング

この章では、Amazon 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
- Amazon 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。

- Amazon 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、Amazon DCV リセラーにお問い合わせください。
- Amazon EC2 で Amazon 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 サポートプランがない場合は、AWS re:Post に質問を投稿することで、Amazon DCV コミュニティに支援を求めることができます。

ログファイルの使用

Amazon DCV ログファイルは、Amazon 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。Amazon DCV ログファイルは、Amazon DCV サーバーの次の場所にあります。

- Windows Server

```
C:\ProgramData\NICE\dcv\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/
```

同様の名前のファイルが複数存在する場合があります。数値サフィックスは、ファイルの経過時間を示します。ファイルが古くなると、数値は高くなります。

接続の問題をトラブルシューティングするとき、ファイルは最も関連性の高いserver.logファイルです。サポートリクエストを送信するときは、フォルダ全体をアタッチすることをお勧めしますが、古いファイルは除外してアタッチサイズを制限できます。

Amazon 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。

トピック

- ログファイルの詳細度の変更

ログファイルの詳細度の変更

詳細度を変更すると、ログの詳細レベルを調整できるため、問題をより効果的に診断して解決できます。ログの詳細度を上げると、システムに関するより詳細な情報がキャプチャされます。ログの詳細度を下げると、ストレージが最適化され、パフォーマンスが向上します。

Windows の場合

2023.0 リリース以降の Amazon DCV バージョンでは、コマンドラインを使用してログファイルの詳細を設定することをお勧めします。古いバージョンでは、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。

リリース 2023.0 以降

1. コマンドプロンプトを開き、管理者として実行します。
2. インストールディレクトリに移動します。

```
C:\Program Files\NICE\DCV\Server\bin
```

3. 次のコマンドを入力してレベルを設定します。

```
dcv set-config --section log --key level "'LEVEL'"
```

4. デバッグを有効にするには、次のコマンドを入力します。

```
dcv set-config --section log --key level "'debug'"
```

> **Note**
> 二重引用符と一重引用符の両方を使用していることを確認してください。

リリース 2022.2 以前

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 の場合

2023.0 リリース以降の Amazon DCV バージョンでは、コマンドラインを使用してログファイルの詳細を設定することをお勧めします。古いバージョンでは、 `dcv.conf` ファイルで `level`パラメータを設定する必要があります。

リリース 2023.0 以降

1. ターミナルを開きます。
2. 次のコマンドを入力してレベルを設定します。

```
sudo dcv set-config --section log --key level "'LEVEL'"
```

3. デバッグを有効にするには、次のコマンドを入力します。

```
sudo dcv set-config --section log --key level "'debug'"
```

> **Note**
> 二重引用符と一重引用符の両方を使用していることを確認してください。

リリース 2022.2 以前

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

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。

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

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もあります。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 Amazon 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-gltest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 Amazon 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。

- Amazon 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 Amazon 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

Windows Server 2016 または Windows 10 以降で実行されている Amazon DCV サーバーでは、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 Amazon 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、Amazon 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。Amazon 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、Amazon 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、Amazon 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 Amazon 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. Amazon 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 Amazon 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 Amazon DCV サーバーとの直接かつ安全な接続を確立しようとします。
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。

Amazon DCV のパフォーマンスと統計のモニタリング

Amazon 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、Amazon 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。

Amazon DCV のパフォーマンスカウンターセット

パフォーマンスカウンターは、Amazon DCV の動作と使用率に関するインサイトを提供するメトリクスです。パフォーマンスカウンターのデータを経時的に収集して分析することで、パフォーマンスのボトルネックを特定し、リソースの使用を最適化し、問題をデバッグし、Amazon DCV の仕組みをより深く理解できます。

DCV パフォーマンスカウンターは 6 つのカウンターセットにグループ化されています。

- Amazon DCV サーバー
- Amazon DCV サーバープロセス
- Amazon DCV サーバーセッション
- Amazon DCV サーバー接続
- Amazon DCV サーバーチャネル
- Amazon DCV サーバーイメージング

Amazon 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アントも含まれており、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	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	カウント	
Active Connections	サーバーへのアクティブな接続の数	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)	カウント	
Idle Disconnections	操作がないために切断された接続の数	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	
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	

カウンター名	説明	単位	メモ
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。

Amazon DCV サーバープロセス

このカウンターセットには、個々の Amazon 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

カウンター名	説明	単位	メモ
			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)	-	

Amazon 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)	[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

カウンター名	説明	単位
Display Count	セッション内のディスプレイ数	カウント

以下のカウンターは Amazon 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)
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

カウンター名	説明
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)

Amazon 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)。	[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)
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

カウンター名	説明
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)

Amazon 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	チャネル経由でデータを受信する速度 (ビット/秒)

カウンター名	説明
Received Bytes	チャネル経由で受信した総バイト数
Send Rate bits/sec	チャネル経由でデータが送信される速度 (ビット/秒)
Sent Bytes	チャネル経由で送信された総バイト数

Amazon 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、Amazon 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、Amazon 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
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	カウント	セッション

カウンター名	説明	単位	インスタンス
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
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

Amazon DCV サーバーパラメータリファレンス

以下の表に、Amazon DCV サーバー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ー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、サーバー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す。
> - `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 Amazon 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 Amazon 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	コンテキストの再ロード	デフォルト値	説明
avsync-support	文字列	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 2021.1-10557 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
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ネル数 — Linux のスピーカーデバイスのチャ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 2020.0-8428 から使用可能。

clipboard パラメータ

以下の表に、Linux Amazon 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 Amazon 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	コンテキストの再ロード	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	コンテキストの再ロード	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 2020.0-8428 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 2017.0-4334 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 2017.0-4334 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-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 2017.0-4334 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 2019.0-7318 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
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 2019.0-7318 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 2020.1-8942 から使用可能。

connectivity パラメータ

以下の表に、Linux Amazon 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 Amazon 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
disconnect-on-lock	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。
disconnect-on-logout	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。
enable-quic-frontend	true または false - DWORD (32 ビット)	server	Linux: true - Windows: 1	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 2020.2-9508 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 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
idle-timeout-warning	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 2017.4-6898 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	文字列	server	['0.0.0.0', '::']	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')。—

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 2022.0-11954 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 サーバーによりリッスンされる 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は、'quic-listen-endpoints' の設定を参照してください。— バージョン 2020.2-9508 から使用可能。
web-extra-http-headers	文字列	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 2017.2-6182 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	文字列	server	['0.0.0.0', '::']	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

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 2022.0-11954 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は、'web-listen-endpoints' の設定を参照してください。— バージョン 2017.0-4100 から使用可能。
web-root	文字列	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を指定します。— バージョン 2017.0-4100 から使用可能。
web-url-path	文字列	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 2017.0-4100 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-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 2017.0-4100 から使用可能。
web-x-frame-options	文字列	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 2017.1-5870 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 2017.0-4100 から使用可能。

display パラメータ

以下の表に、Linux Amazon 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 Amazon 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	文字列	セッション	[]	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':<800>, 'h':<600>, 'x':<0>, 'y': <0>}, {'w':<1024>, 'h':<768>, 'x':<800>,'y':<0>}] — バージョン 2017.0-5600 から使用可能。
cuda-devices	文字列	connection	[]	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

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
				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'] 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 2017.2-6182 から使用可能。
enable-client-resize	true または false - DWORD (32 ビット)	セッション	Linux: true - Windows: 1	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 2017.0-4100 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
enable-qu	true または false - DWORD (32 ビット)	セッション	Linux: true - Windows: 1	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 2017.0-4100 から使用可能。
enable-yuv444-encoding	文字列	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 2022.0-11954 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	フレームグラバーの 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 2017.1-5870 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 2017.0-4100 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-head-resolution	文字列	custom	(4096、2160)	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。サポートされる最大解像度は (4096, 4096) です。— バージョン 2017.0-4100 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-layout-area	integer - DWORD (32 ビット)	custom	0	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 2019.1-7423 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-num-heads	integer - DWORD (32 ビット)	custom	4	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 2017.0-4100 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
min-head-resolution	文字列	custom	(640、480)	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 2017.0-4100 から使用可能。
target-fps	integer - DWORD (32 ビット)	セッション	-1	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 2017.0-4100 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
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 2017.0-4100 から使用可能。
web-client-max-head-resolution	文字列	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。最大ヘッド解像度の制限オプションは、このオプションで設定された最大幅と最大高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 2020.0-8428 から使用可能。

display/linux パラメータ

以下の表に、Linux Amazon 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と、Windows Amazon DCV サーバーの display/linux 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
disable-local-console	true または false - DWORD (32 ビット)	セッション	Linux: true - Windows: 1	ローカル画面を空白にし、ローカル入力を禁止します (Xorg の Linux コンソールセッションのみ) — ローカルに接続されたモニターを空白にして、リモートクライアントが接続されている間は進行中の DCV コンソールセッションが表示されないようにし、ローカル入力デバイスを無効にします。モニターと入力デバイスは、最後のクライアントが切断されると再び有効になります。これにより、ローカルユーザーがリモートコンソールセッションに介入できなくなり、プライバシーが向上します。名前が "DCV " (スペースを含む) または "Xdcv " (スペースを含む) で始まる入力デバイスは、設定では無効になっていません。— バージョン 2024.0 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
enable-console-scaling	true または false - DWORD (32 ビット)	セッション	Linux: false - Windows: 0	フレームバッファのスケーリングを有効にして任意の解像度をサポート (Xorg の Linux コンソールセッションのみ) — フレームバッファからスキャンアウトバッファへの XRandR 変換を適切に設定することで、コンソールセッションの任意の解像度を許可します。"false" (デフォルト) に設定した場合は、レガシー動作が予想されます (ローカルに接続されたディスプレイで直接サポートされている解像度のみが許可されます)。"true" に設定した場合、クライアントは、物理的に接続されたディスプレイでサポートされているかどうかに関係なく、サーバーで任意の解像度をリクエストできます。— バージョン 2024.0 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
gl-displays	文字列	セッション	[':0.0']	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す。— バージョン 2017.0-4100 から使用可能。

extensions パラメータ

以下の表に、Linux Amazon 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 Amazon 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	コンテキストの再ロード	デフォルト値	説明
enabled	true または false - DWORD	connection	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

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
	(32 ビット)			ティー拡張機能を使用することはできません。— バージョン 2023.0-14852 から使用可能。

input パラメータ

以下の表に、Linux Amazon 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 Amazon 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	コンテキストの再ロード	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 での自動リピートを許可するかどうか — 単一のキーに対して自動リピートを許可するかどうかを指定します。— バージョン 2017.2-6182 から使用可能。
enable-gamepad	true または false - DWORD (32 ビット)	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 — ゲームパッドを有効にするかどうかを指定します。— バージョン 2022.0-11954 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
enable-relative-mouse	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 2017.0-5121 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 2019.0-7318 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 2017.3-6698 から使用可能。

license パラメータ

以下の表に、Linux Amazon 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 Amazon 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	文字列	セッション	"	ライセンス — EC2 以外のインスタンスで実行する場合に、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

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
				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 2017.0-4100 から使用可能。

log パラメータ

以下の表に、Linux Amazon 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 Amazon 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	コンテキストの再ロード	デフォルト値	説明
directory	文字列	server	"	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、デフォルトは Windows では「C:\ProgramData\NICE\DCV\log\」、Linux では「/var/log/dcv/」です。— バージョン 2017.0-4100 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
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 2023.0-14852 から使用可能。
level	文字列	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'、'info'、'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 2017.0-4100 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	ローテーション前の最大ログファイルサイズ (メガバイト) — ローテーションがトリガーされる前の最大ログファイル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 2022.1-13067 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 2017.0-4100 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
rotation-interval	文字列	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。他に指定できる値は、'every-minute'、'every-twenty-minutes'、'every-hour'、'every-day' です。— バージョン 2022.1-13067 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
rotation-suffix	文字列	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す。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 2022.1-13067 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
transfer-audit	文字列	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」、「all」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 2017.0-4100 から使用可能。

`printer` パラメータ

以下の表に、Linux Amazon 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 Amazon 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	コンテキストの再ロード	デフォルト値	説明
file-printer-name	文字列	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	コンテキストの再ロード	デフォルト値	説明
				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 2022.0-11954 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
use-default-printer	文字列	カスタム	'client-decides'	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、'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プリンタが送信されない場合、'printer' セクションの 'file-printer-name' で指定されているプリンタが設定されます。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 2022.2-13907 から使用可能。

redirection パラメータ

以下の表に、Linux Amazon 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 Amazon 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	コンテキストの再ロード	デフォルト値	説明
enable-timezone-redirection	文字列	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	コンテキストの再ロード	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 2022.2-13907 から使用可能。

security パラメータ

以下の表に、Linux Amazon 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 Amazon 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	コンテキストの再ロード	デフォルト値	説明
allowed-http-host-regex	文字列	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
				。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: '^(www\.)?example\.com$' — バージョン 2017.0-4100 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
allowed-ws-origin-regex	文字列	server	'^https://.+$'	許可されたオリジン — この DCV サーバーが受け入れるオリジンを表す正規表現パターンを指定します。WebSocket 接続を確立する際、クライアントハンドシェイクのオリジンヘッダーフィールドは、接続を確立するスクリプトのオリジンを示します。着信 HTTP リクエストのオリジンヘッダーがこのパターンに一致していない場合、リクエストそのものが、HTTP 403 (禁止) ステータスコードで失敗します。これは、クロスサイトの WebSocket ハイジャック (CSWSH) 攻撃を回避するためのセキュリティ対策です。Javascript のように有効な正規表現のパターンである必要があります。パターンの文字は、大文字と小文字の両方に一致します。オリジンヘッダーの形式は <scheme> "://" <host> [":" <port>] です。例: '^https://(www\.)?example\.com(:443)?$'. — バージョン 2017.0-4100 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
auth-connection-setup-timeout	integer - DWORD (32 ビット)	server	120	認証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認証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、認証チャ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 2017.0-4100 から使用可能。
auth-token-verifier	文字列	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。空でない場合、エンドポイントへの有効な変更はすぐに適用されます。サーバーを再起動する必要はありません (再起動せずに空に変更することはできません)。— バージョン 2017.0-4100 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
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 2023.0-14852 から使用可能。
認証	文字列	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 2017.0-4100 から使用可能。
authentication-threshold	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 2017.0-4100 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
ca-file	文字列	カスタム	"	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 2017.0-4100 から使用可能。
certificate-to-user-file	文字列	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 2022.0-11954 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
ciphers	文字列	server	'ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES128-SHA256:ECDHE-RSA-AES256-SHA384'	TLS 接続で使用される暗号リスト — TLS 接続で使用す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 2017.0-4100 から使用可能。
connection-estab-timeout	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 2017.0-4100 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
connection-setup-timeout	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 2017.0-4100 から使用可能。
crl-file	文字列	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 2022.0-11954 から使用可能。
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 2017.3-6698 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 2017.0-4100 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
no-tls-strict	true または false - DWORD (32 ビット)	カスタム	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。このパラメータの変更はすぐに適用されます。サーバーを再起動する必要はありません。— バージョン 2017.0-4100 から使用可能。
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 2017.1-5777 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
pam-service-name	文字列	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 2017.0-4100 から使用可能。
passwd-file	文字列	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、${XDG_CONFIG_HOME}/NICE/dcv/passwd (Linux の場合)、%CSIDL_LOCAL_APPDATA%\NICE\dcv\passwd (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 2017.0-4100 から使用可能。
server-fqdn	文字列	server	''	サーバー FQDN — サーバーの完全修飾ドメイン名。空は gethostname() を意味します。— バージョン 2017.3-6698 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
service-name	文字列	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 2020.0-8428 から使用可能。
supervision-control	文字列	カスタム	無効	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 2021.3-11591 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
user-realm	文字列	server	"	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 2017.3-6698 から使用可能。

session-management パラメータ

以下の表に、Linux Amazon 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 Amazon 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	コンテキストの再ロード	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 2017.0-4100 から使用可能。
enable-gl-in-virtual-sessions	文字列	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

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
				ンス)。使用できる値: 'always-on'、'always-off'、'default-on'、'default-off'。— バージョン 2017.0-4100 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。自動セッションにのみ制限を設定するには、「session-management/automatic-console-session」セクションの「max-concurrent-clients」を使用します。— バージョン 2017.0-4100 から使用可能。
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 2019.0-7318 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-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 2021.0-10242 から使用可能。
performance-profile	文字列	カスタム	'none'	一部の機能の初期化中に使用されるプロファイルを指定 — インスタンスのパフォーマンスプロファイルに従って、一部の機能のデフォルト設定を調整します。プロファイルが選択されていない場合、プロファイルは CPU の数および GPU が存在するかどうかに応じて自動的に推定されます。使用できる値は、「low」、「medium」、「high」、「none」です。— バージョン 2024.0 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	文字列	セッション	[]	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。デフォルトレイアウトの値の例: [{'w': <800>, 'h': <600>, 'x <0>':, 'y': <0>}, {'w <1024>': <768>, 'h': <800>, 'x':, 'y': <0>}] 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。このキーを設定すると、有効になっているモニターの数 (virtual-session-monitors 設定で指定) は無視されます。— バージョン 2017.0-5600 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-font-path	文字列	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 2017.0-4100 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 2021.3-11591 から使用可能。
virtual-session-xdcv-args	文字列	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 2017.0-4334 から使用可能。

session-management/automatic-console-session パラメータ

以下の表に、Linux Amazon 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 Amazon 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	文字列	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。許可される値は 'reject-new-connection' (着信接続は拒否される) と 'same-user-oldest-connection' (サーバーは、同じユーザーの接続のうちセッションと最も長い時間通信していない接続か、その情報がない場合は接続時間が最も古いものを終了する) です。— バージョン 2022.1-13067 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	コンテキストの再ロード	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 2017.0-5600 から使用可能。
所有者	文字列	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 2017.0-5600 から使用可能。
permissions-file	文字列	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 2017.0-5600 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
storage-root	文字列	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 2017.0-5600 から使用可能。

session-management/defaults パラメータ

以下の表に、Linux Amazon 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 Amazon 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	コンテキストの再ロード	デフォルト値	説明
permissions-file	文字列	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	コンテキストの再ロード	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 2017.0-5600 から使用可能。

smartcard パラメータ

以下の表に、Linux Amazon 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 Amazon 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	コンテキストの再ロード	デフォルト値	説明
enable-cache	文字列	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、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
				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 2017.2-6182 から使用可能。

webauthn パラメータ

以下の表に、Linux Amazon 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 Amazon 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	コンテキストの再ロード	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 YubiKey 、Windows Hello など、ローカル認証ツール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利用できません。— バージョン 2023.1-16220 から使用可能。

webcam パラメータ

以下の表に、Linux Amazon 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 Amazon 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	コンテキストの再ロード	デフォルト値	説明
max-resolution	文字列	connection	(0, 0)	ウェブカメラの最大解像度 — クライアントが提供する解像度の中から、選択可能な、ア

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
				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を指定します。この値が欠落している、または (0、0) の場合は、デフォルト値が使用されます。デフォルト値: 高中程度のパフォーマンスプロファイルの場合は 1280x720、低パフォーマンスプロファイルの場合は 640x480。— バージョン 2021.0-10242 から使用可能。
preferred-resolution	文字列	connection	(0, 0)	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ます。この値が欠落している、または (0, 0) の場合は、デフォルト値が使用されます。デフォルト値: 高中程度のパフォーマンスプロファイルの場合は 640x360、低パフォーマンスプロファイルの場合は 424x240。— バージョン 2021.0-10242 から使用可能。

windows パラメータ

以下の表に、Linux Amazon DCV サーバーの /etc/dcv/dcv.conf ファイルの [windows] セクションの設定パラメータと、Windows Amazon 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	コンテキストの再ロード	デフォルト値	説明
disable-display-sleep	true または false - DWORD (32 ビット)	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ないようにするかどうかを指定します。— バージョン 2017.0-4100 から使用可能。
printer	文字列	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: 'プリンター' セクションの 'file-printer-name' を使用。— バージョン 2017.0-4100 から使用可能。

設定パラメータの変更

このセクションでは、Amazon 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、「Amazon 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 Amazon DCV サーバー
- Linux Amazon DCV サーバー

Windows Amazon DCV サーバー

Windows Amazon 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、「Amazon 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、「Amazon 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。

PowerShell を使用して設定パラメータを変更するには

1. 管理者として PowerShell を実行します。
2. 「Amazon 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. 「Amazon 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. 「Amazon 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 Amazon DCV サーバー

Linux Amazon 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、「Amazon 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、「Amazon 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

「Amazon 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'
```

Amazon DCV のサポート終了

Amazon DCV のサポート終了 (EOSL、End of Support Life) は、Amazon 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。

Amazon 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、Amazon DCV サーバーと Amazon 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 Amazon DCV バージョンにアップグレードする必要があります。

下位互換性は保持され、同じ EOSL ルールが適用されます。つまり、サーバーとクライアントの両方がサポートされている場合、Amazon DCV クライアントは Amazon DCV サーバーに接続でき、DCV クライアント/ビューワーは Amazon DCV サーバーに接続できます。

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、Amazon DCV メジャーバージョンの EOSL タイムラインを示します。

Amazon DCV メジャーバージョン	初回リリース日	EOSL 日
Amazon DCV 2016.x	2015 年 12 月 31 日	2021 年 3 月 31 日

Amazon DCV メジャーバージョン	初回リリース日	EOSL 日
Amazon DCV 2017.x	2017 年 12 月 18 日	2021 年 12 月 31 日
Amazon DCV 2019.x	2019 年 8 月 5 日	2022 年 12 月 31 日
Amazon DCV 2020.x	2020 年 4 月 16 日	2023 年 12 月 31 日
Amazon DCV 2021.x	2021 年 4 月 12 日	2024 年 12 月 31 日
Amazon DCV 2022.x	2022 年 2 月 23 日	2025 年 12 月 31 日
Amazon DCV 2023.x	2023 年 5 月 3 日	2026 年 12 月 31 日
Amazon DCV 2024.x	2024 年 10 月 2 日	2027 年 12 月 31 日

Note

2025 年 3 月 31 日以降、次の Amazon DCV バージョンはサポートされなくなります。

- Windows クライアントバージョン 2023.1.8993 以前

- Linux/MacOS クライアント 2023.1.6203 以前

お客様向けの EOSL パス

で Amazon DCV を実行している場合は AWS、Amazon DCV のライセンスは必要ありません。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。EOSL の日付を過ぎた Amazon DCV バージョンを現在使用している場合は、Amazon DCV ダウンロードページを使用して最新の Amazon DCV バージョンにアップグレードするか、 から Amazon DCV AMI を使用してサポート AWS Marketplace を続行します。

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 Amazon DCV を実行しており、現在使用している Amazon 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、Amazon DCV の最新バージョンに無償でアップグレードできます。Amazon 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 Amazon 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、Amazon DCV サポート契約の条件により、Amazon 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 Amazon 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、Amazon 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 Amazon 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、Amazon 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 Amazon DCV サーバーで EOSL に達した Amazon 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 とお客様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 — AWS 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 AWS 責任があります。 AWS また、 は、お客様が安全に使用できるサービスも提供します。AWS コンプライアンス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。Amazon DCV 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、「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。
- クラウド内のセキュリティ – 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、ユーザーは、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、Amazon DCV 使用時の責任共有モデルの適用方法を理解するうえで役立ちます。次のトピックでは、セキュリティおよびコンプライアンスの目標を達成するために Amazon DCV を構成する方法について説明します。また、Amazon DCV リソースのモニタリングや保護に役立つ他の AWS サービスの使用方法についても説明します。

内容

- Amazon DCV のデータ保護
- Amazon DCV のコンプライアンス検証

Amazon DCV のデータ保護

Amazon DCV でのデータ保護には、 AWS の責任共有モデルが適用されます。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。ユーザーは、このインフラストラクチャでホストされ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 のサービス 」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

に関するよくある質問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に投稿された AWS 責任共有モデルおよび GDPR の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 (IAM) 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。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。TLS 1.2 が必須で、TLS 1.3 をお勧めします。
- で 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。CloudTrail 証跡を使用して AWS アクティビティをキャプチャする方法については、「 AWS CloudTrail ユーザーガイド」のCloudTrail 証跡の使用」を参照してください。
- AWS 暗号化ソリューションと、その中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 のサービス。
- Amazon Macie などの高度な管理された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 API AWS を介して にアクセスするときに FIPS 140-3 検証済み暗号化モジュールが必要な場合は、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を、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含めないことを強くお勧めします。これは、コンソール、API、または SDK を使用して Amazon DCV AWS CLIまたは他の AWS のサービス を使用する場合も同様です。 AWS SDKs タグ、または名前に使用される自由記述のテキストフィールドに入力したデータは、請求または診断ログに使用される場合があります。外部サーバーに URL を提供する場合、そのサーバーへのリクエストを検証できるように、認証情報を URL に含めないことを強くお勧めします。

データの暗号化

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

Amazon DCV 自体には顧客データは保存されません。Amazon DCV サーバーホスト上のデータは保管中に暗号化できます。で Amazon DCV 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

Amazon DCV クライアントおよび Amazon 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。

Amazon DCV のコンプライアンス検証

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。Amazon 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わりません。

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コンプライアンス」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。

Amazon DCV を使用する際のお客様のコンプライアンス責任は、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デプロイする手順を示します 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。

Amazon 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、Amazon 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

- Amazon DCV リリースノート
- ドキュメント履歴

Amazon DCV リリースノート

このセクションでは、Amazon 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 2024.0-19030 — 2025 年 6 月 10 日
- DCV 2024.0-19030 — 2025 年 5 月 16 日
- DCV 2024.0-19030 — 2025 年 3 月 31 日
- DCV 2024.0-18131 — 2024 年 10 月 31 日
- DCV 2024.0-17979 — 2024 年 10 月 1 日
- DCV 2023.1-17701 — 2024 年 9 月 10 日
- DCV 2023.1-17701 — 2024 年 8 月 20 日
- 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 œ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 2024.0-19030 — 2025 年 6 月 10 日

ビルド番号	変更とバグ修正	
• nice-dcv-server: 19030 • nice-dcv-client(Windows): 9431 • nice-dcv-viewer (macOS): 8004 • nice-dcv-viewer (Linux): 8004 • nice-dcv-web-viewer: 19030 • nice-xdcv: 654	• ブランドを変更した SM エージェントインストーラ。 • 有効期限が切れる前に証明書を更新しました。 • Linux および MacOS クライアントのデッドキーを修正しました。 • アクセスコンソールを更新して、クレームスコープとは異なる UUID を使用します。 • JDK 17 への移行。	

ビルド番号	変更とバグ修正	
• nice-dcv-gl: 1096 • nice-dcv-gltest: 352 • nice-dcv-simple-external-authenticator: 266		

DCV 2024.0-19030 — 2025 年 5 月 16 日

ビルド番号	変更とバグ修正	
• nice-dcv-server: 19030 • nice-dcv-client(Windows): 9431 • nice-dcv-viewer (macOS): 8004 • nice-dcv-viewer (Linux): 8004 • nice-dcv-web-viewer: 19030	• 拡張機能データをパブリック CLI パラメータにしました。 • が接続ファイルからログレベルを設定できるようにします。 • Linux および MacOS クライアントのデッドキーを修正しました。 • WebAuthn の修正。 • MacOS クライアントのディスプレイスケール係数、相対マウス、および複数のウェブカメラを修正しました。 • Windows および MacOS クライアントのクリップボードイメージ処理が改善されました。 • その他の修正とパフォーマンスの向上。	

ビルド番号	変更とバグ修正	
• nice-xdcv: 654 • nice-dcv-gl: 1096 • nice-dcv-gltest: 352 • nice-dcv-simple-external-authenticator: 266		

DCV 2024.0-19030 — 2025 年 3 月 31 日

ビルド番号	変更とバグ修正	
• nice-dcv-server: 19030 • nice-dcv-client(Windows): 9254 • nice-dcv-viewer (macOS): 7209 • nice-dcv-viewer (Linux): 7209	• Amazon Linux 2023 のサポート。 • パフォーマンス向上とバグ修正が行われています。	

ビルド番号	変更とバグ修正	
• nice-dcv-web-viewer: 19030 • nice-xdcv: 654 • nice-dcv-gl: 1096 • nice-dcv-gltest: 352 • nice-dcv-simple-external-authenticator: 266		

DCV 2024.0-18131 — 2024 年 10 月 31 日

ビルド番号	変更とバグ修正	
• nice-dcv-server: 18131 • nice-dcv-client(Windows): 9254 • nice-dcv-viewer (macOS): 7209 • nice-dcv-viewer	• Windows 上の Amazon DCV サーバーのインストーラで、アンistallation が失敗する問題を修正しました。 • Linux の Amazon DCV サーバーで、物理ディスプレイが接続されているときにマウスカーソルが非表示になる問題を修正しました。 • USB リダイレクト機能の使用時にローカル USB デバイスが切断される可能性がある Windows ネイティブクライアントの問題を修正しました。 • クリップボードに関連する macOS のクラッシュを修正しました。 • 複数の DCV セッションを持つマシンに接続するときの macOS および Linux クライアントのクラッシュを修正しました。	

ビルド番号	変更とバグ修正	
(Linux): 7209 • nice-xdcv: 631 • nice-dcv-gl: 1078 • nice-dcv-gltest: 344 • nice-dcv-simple-external-authenticator: 259	• macOS および Linux クライアントで DCV PDF プリンターを介して大きなドキュメントを印刷できない問題を修正しました。 • Firefox 130 以降の Web クライアントでのレンダリングの問題を修正しました。	

DCV 2024.0-17979 — 2024 年 10 月 1 日

ビルド番号	新機能	変更とバグ修正
• nice-dcv-server: 17979 • nice-dcv-client (Windows): 9206 • nice-dcv-viewer (macOS) 6995	Amazon DCV に次の機能を追加しました。 • NICE DCV 製品の名前が Amazon DCV に変更されました。 • Ubuntu 24 サーバーとクライアントのサポート。 • Linux ネイティブクライアントの Wayland サポート。 • Linux サーバー上のコンソールセッションの場合、DCV はローカル画面を空白にして入力をブロックします。 • macOS 上のウェブクライアントで高ピクセル密度を実現。	• QUIC UDP は、Amazon DCV サーバーでデフォルトで有効になっています。 • RHEL 7 と CentOS 7 および 8 のサポートは終了しました。 • Amazon DCV サーバーとクライアントの Windows バージョンは、Visual Studio 2017 用の Microsoft Visual C++ 再頒布可能パッケージではなく、Visual Studio 2022 用の Microsoft Visual C++ 再頒布可能パッケージを要件としています。

ビルド番号	新機能	変更とバグ修正
• nice-dcv-viewer (Linux): 6995 • nice-xdcv: 627 • nice-dcv-gl: 1073 • nice-dcv-gltest: 340 • nice-dcv-simple-external-authenticator: 256	• Linux および macOS クライアントにエコーエコーキャンセレーションを追加。	• ウェブクライアントのアップロード速度制限を 10 Mbps と 100 Mbps に引き上げました。 • ウェブクライアント内の複数のディスプレイ間でウィンドウをドラッグアンドドロップする機能を追加しました。 • ネットワークパフォーマンスが低下すると、ウェブクライアントにバナーが表示されます。 • セキュリティとパフォーマンスの向上

DCV 2023.1-17701 — 2024 年 9 月 10 日

ビルド番号	変更とバグ修正	
• nice-dcv-	• Windows クライアントの DCV 拡張機能で Window ハンドルの取得をサポート。	

ビルド番号	変更とバグ修正	
server: 17701 • nice-dcv-client (Wi: 9210 • nice-dcv-viewer (macOS) 6809 • nice-dcv-viewer (Linux): 6809 • nice-xdcv: 565 • nice-dcv-gl: 1047 • nice-dcv-gltest: 325 • nice-dcv-simple-external-au		

ビルド番号	変更とバグ修正	
thenticator: 228		

DCV 2023.1-17701 — 2024 年 8 月 20 日

ビルド番号	変更とバグ修正	
• nice-dcv-server: 17701 • nice-dcv-client (Wi: 9187 • nice-dcv-viewer (macOS) 6809 • nice-dcv-viewer (Linux): 6809 • nice-xdcv: 565 • nice-dcv-gl: 1047	• パフォーマンス向上とバグ修正が行われています。	

ビルド番号	変更とバグ修正	
• nice-dcv-gltest: 325 • nice-dcv-simple-external-authenticator: 228		

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: 1(• nice-dcv-client (Wi: 9127 • nice-dcv-viewer (macOS) 6703 • nice-dcv-	• ネイティブクライアントのパフォーマンス向上とバグ修正が行われています。	

ビルド番号	変更とバグ修正	
viewer (Linux): 6703 • nice-xdcv: 565 • nice-dcv-gl: 1047 • nice-dcv-gltest: 325 • nice-dcv-simple-external-authenticator: 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: 1(• nice-dcv-	• ディスプレイのスケーリングが 100% 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。	

ビルド番号	変更とバグ修正	
client (Wi : 8993 • nice-dcv-viewer (macOS) 6203 • nice-dcv-viewer (Linux): 6203 • nice-xdcv: 565 • nice-dcv-gl: 1047 • nice-dcv-gltest: 325 • nice-dcv-simple-external-authenticator: 228	• 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。	

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: 1(• nice-dcv-client (Wi : 8934 • nice-dcv-viewer (macOS) 6203 • nice-dcv-viewer (Linux): 6203 • nice-xdcv: 565 • nice-dcv-gl: 1047 • nice-dcv-gltest: 325 • nice-dcv-simple-	• 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。 • 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 `dcv list-connections` で報告される、最後の対応時間を修正しました。 • 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。 • 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。 • 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。 • 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。 • 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。 • Linux および macOS クライアントでの WebAuthN のリダイレクトに関する問題を修正しました。	

ビルド番号	変更とバグ修正	
ex ternal- au thenticat or: 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
• nice-dcv-server: 1(• nice-dcv-client (Wi : 8908 • nice-dcv-viewer (macOS) 6125 • nice-dcv-viewer (Linux): 6125 • nice-xdcv: 565	Amazon DCV に次の機能を追加しました。 • リモートの 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、セッション中の WebAuthn リクエストのリダイレクトをサポートします。リダイレクトされたリクエストはクライアントに転送され、YubiKey や Windows Hello などの FIDO2 準拠の認証システムでユーザー ID を検証できるようになります。 • Windows ホスト用の新しい間接ディスプレイドライバー (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。 • 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。	• 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。 • 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い問題を修正しました。 • Windows でモニターのスケーリング係数が 100% Amazon DCV サーバーにリセットされる問題を修正しました。 • 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。 • 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。

ビルド番号	新機能	変更とバグ修正
• nice-dcv-gl: 1047 • nice-dcv-gltest: 325 • nice-dcv-simple-external-authenticator: 228		• 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。 • 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。 • macOS ネイティブクライアントでのスマートカードサポートに関する問題が修正されました。 • 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。 • 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。 • 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。 • 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。 • Xdcv が XServer のバージョン 21.1.9 に更新されました。 • Windows Server 2012R2、Ubuntu 18.04、Suse Enterprise Linux 15SP4 のサポートを削除しました。

ビルド番号	新機能	変更とバグ修正
		• バグを修正してパフォーマンスを改善しました。

DCV 2023.0-15487 — 2023 年 6 月 29 日

ビルド番号	変更とバグ修正	
• nice-dcv-server: 1 • nice-dcv-client (Wi : 8771 • nice-dcv-viewer (macOS) 5629 • nice-dcv-viewer (Linux): 5629 • nice-xdcv: 551 • nice-dcv-gl: 1039	• 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。 • アンインストール時のエラーを防ぐために、Amazon DCV サーバーと Xdcv の el7 rpm パッケージを修正しました。 • 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。 • 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。 • 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 Amazon DCV サーバーパッケージが必要になりました。 • 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。 • 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。	

ビルド番号	変更とバグ修正	
• nice-dcv-gltest: 318 • nice-dcv-simple-external-authenticator: 208	• 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。 • 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。 • OpenSSL サードパーティーライブラリを更新しました。	

DCV 2023.0-15065 — 2023 œ年 5 月 3 日

ビルド番号	変更とバグ修正	
• nice-dcv-server: 1 • nice-dcv-client (Wi: 8671 • nice-dcv-viewer (macOS) 5483 • nice-dcv-	• 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。 • BigSur 上の 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。	

ビルド番号	変更とバグ修正	
viewer (Linux): 5483 • nice-xdcv: 547 • nice-dcv-gl: 1027 • nice-dcv-gltest: 318 • nice-dcv-simple-external-authenticator: 208		

DCV 2023.0-15022 — 2023 年 4 月 21 日

ビルド番号	変更とバグ修正	
• nice-dcv-server: 1 • nice-dcv-	• 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。 • QUIC 接続で障害を引き起こす Amazon DCV サーバーの競合状態を修正しました。	

ビルド番号	変更とバグ修正	
client (Wi : 8671 • nice-dcv-viewer (macOS) 5456 • nice-dcv-viewer (Linux): 5456 • nice-xdcv: 547 • nice-dcv-gl: 1027 • nice-dcv-gltest: 318 • nice-dcv-simple-external-authenticator: 206	• 隠しカーソルがあるアプリケーションに関連する Amazon DCV サーバーのクラッシュを修正しました。 • 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。 • 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。 • ICU と libxml2 のサードパーティライブラリを更新しました。 • 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。 • 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。 • 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。	

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
• nice-dcv-server: 1 • nice-dcv-client (Wi: 8655 • nice-dcv-viewer (macOS) 5388 • nice-dcv-viewer (Linux): 5388 • nice-xdcv: 527 • nice-dcv-gl: 1022 • nice-dcv-gltest: 318 • nice-dcv-simple-	Amazon DCV に次の機能を追加しました。 • macOS および Linux 上の Amazon 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。 • 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。 • Added Red Hat Enterprise Linux 9、Rocky Linux 9、CentOS Stream 9 を追加しました。 • Linux の Amazon 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。	• 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。 • 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。 • 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 Amazon DCV が使用されるようになりました。 • 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。 • 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。

ビルド番号	新機能	変更とバグ修正
external-authenticator: 206		

DCV 2022.2-14521 — 2023 年 2 月 17 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 14521 • nice-dcv-client (Windows): 8570 • nice-dcv-viewer (macOS): 5125 • nice-dcv-viewer (Linux): 4804 • nice-xdcv: 519 • nice-dcv-gl: 1012 • nice-dcv-gltest: 307 • nice-dcv-simple-external-authenticator: 198	• 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。 • Windows Amazon DCV サーバーのテンキーに関する問題を修正しました。 • QUIC 接続でのメモリリークを修正しました。 • 古いビデオドライバーを使用する場合の Windows Amazon DCV クライアントの安定性が向上しました。 • OpenSSL とlibsoup のサードパーティーライブラリを更新しました。 • Xdcv が XServer のバージョン 21.1.7 に更新されました。

DCV 2022.2-14357 — 2023 年 1 月 18 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 14357 • nice-dcv-client (Windows): 8522 • nice-dcv-viewer (macOS): 4804 • nice-dcv-viewer (Linux): 4804 • nice-xdcv: 487 • nice-dcv-gl: 1012 • nice-dcv-gltest: 307 • nice-dcv-simple-external-authenticator: 198	• 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。 • 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。 • 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。 • 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)。 • Windows Amazon 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。 • Windows Amazon DCV クライアント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。

DCV 2022.2-14175 — 2022 年 12 月 21 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 14175 • nice-dcv-client (Windows): 8472 • nice-dcv-viewer (macOS): 4804 • nice-dcv-viewer (Linux): 4804	• WebSocket 接続を使用する際の、サーバー内のファイル記述子のリークを修正しました。 • Xdcv が XServer のバージョン 21.1.6 に更新されました。

ビルド番号	変更とバグ修正
• nice-xdcv: 487 • nice-dcv-gl: 983 • nice-dcv-gltest: 307 • nice-dcv-simple-external-authenticator: 198	

DCV 2022.2-14126 — 2022 年 12 月 9 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 14126 • nice-dcv-client (Windows): 8472 • nice-dcv-viewer (macOS): 4804 • nice-dcv-viewer (Linux): 4804 • nice-xdcv: 481 • nice-dcv-gl: 983 • nice-dcv-gltest: 301 • nice-dcv-simple-external-authenticator: 198	• 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。 • 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。 • '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。 • 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。 • 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。 • 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。 • 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。 • 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。 • 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。

ビルド番号	変更とバグ修正
	• 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。 • LibTIFF と MIT-Kerberos のオープンソース依存関係を更新しました。

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
• nice-dcv-server: 1 • nice-dcv-client (Wi: 8427 • nice-dcv-viewer (macOS) 4653 • nice-dcv-viewer (Linux): 4653 • nice-xdcv: 481 • nice-dcv-gl: 983	Amazon DCV に次の機能を追加しました。 • Windows の Amazon 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。 • 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。 • macOS および Linux で Amazon 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。 • Windows の Amazon 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。 • 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。 • オープンソースドライバーを使用する際に、AMD GPU に VAAPI ベースのエンコーダーを追加しました。	• 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。 • 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。 • 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。 • 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。 • 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。 • 数字用キーパッドの Enter キーの問題を修正しました。

ビルド番号	新機能	変更とバグ修正
• nice-dcv-gltest: 301 • nice-dcv-simple-external-authenticator: 198		

DCV 2022.1-13300 — 2022 年 8 月 4 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 13300 • nice-dcv-client (Windows): 8261 • nice-dcv-viewer (macOS): 4279 • nice-dcv-viewer (Linux): 4251 • nice-xdcv: 433 • nice-dcv-gl: 973 • nice-dcv-gltest: 295 • nice-dcv-simple-external-authenticator: 193	• 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。 • 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。 • 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。

DCV 2022.1-13216 — 2022 年 7 月 21 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 13216 • nice-dcv-client (Windows): 8261 • nice-dcv-viewer (macOS): 4251 • nice-dcv-viewer (Linux): 4251 • nice-xdcv: 433 • nice-dcv-gl: 966 • nice-dcv-gltest: 295 • nice-dcv-simple-external-authenticator: 193	• 2019.1 以前の Amazon 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。 • Windows サーバーでのスマートカードリダイレクトに関する問題を修正しました。 • GPU を搭載するホストの Amazon 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。

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
• nice-dcv-server: 1 • nice-dcv-client (Wi : 8248 • nice-dcv-viewer (macOS) 4241 • nice-dcv-viewer	Amazon DCV に次の機能を追加しました。 • 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。 • 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。 • 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。	• 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。 • 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。 • 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。 • 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

ビルド番号	新機能	変更とバグ修正
(Linux): 4241 • nice-xdcv: 433 • nice-dcv-gl: 966 • nice-dcv-gltest: 295 • nice-dcv-simple-external-authenticator: 193		しないことがある問題を修正しました。

DCV 2022.0-12760 — 2022 年 5 月 23 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 12760 • nice-dcv-client (Windows): 8145 • nice-dcv-viewer (macOS): 4131 • nice-dcv-viewer (Linux): 4131 • nice-xdcv: 424 • nice-dcv-gl: 961	変更: web-url-path オプションを指定すると Web クライアントが正常に接続されない問題を修正しました。

ビルド番号	変更とバグ修正
• nice-dcv-gltest: 291 • nice-dcv-simple-external-authenticator: 188	

DCV 2022.0-12627 — 2022 年 5 月 19 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 12627 • nice-dcv-client (Windows): 8145 • nice-dcv-viewer (macOS): 4131 • nice-dcv-viewer (Linux): 4131 • nice-xdcv: 424 • nice-dcv-gl: 961 • nice-dcv-gltest: 291 • nice-dcv-simple-external-authenticator: 188	変更: • 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。 • 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。 • 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。 • 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。 • OpenSSL、zlib、gdk-pixbuf のサードパーティライブラリを更新しました。

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
• nice-dcv-server: 12123 • nice-dcv-client (Windows): 7920	Amazon DCV に次の機能を追加しました。 • macOS および Linux クライアントで高い色精度を有効	変更: • 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。

ビルド番号	新機能	変更とバグ修正
• nice-dcv-viewer (macOS): 3973 • nice-dcv-viewer (Linux): 3973 • nice-xdcv: 424 • nice-dcv-gl: 961 • nice-dcv-gltest: 291 • nice-dcv-simple-external-authenticator: 188	にするオプションを追加しました。	修正内容: • 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。 • Windows ネイティブクライアントで Bluetooth 経由で接続される DualShock 4 コントローラーの問題を修正しました。 • 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。

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
• nice-dcv-server: 11954 • nice-dcv-client (Windows): 7866 • nice-dcv-viewer (macOS): 3929 • nice-dcv-viewer (Linux): 3929 • nice-xdcv: 424 • nice-dcv-gl: 961 • nice-dcv-gltest: 291	Amazon DCV に次の機能を追加しました。 • 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。 • Amazon DCV ウェブクライアントは、サポートされるブラウザで WebCodecs を利用できるようになりました。	変更: • Amazon 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。 • 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 アドレスでリッスンするように Amazon DCV サーバーを構成できるようになりました。

ビルド番号	新機能	変更とバグ修正
• nice-dcv-simple-external-authenticator: 188	• 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。 • 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。 • 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。	• 「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。 • Windows の Amazon 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。 修正内容: • 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。 • 一部の USB デバイスが Amazon 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。 • 認可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。 • Windows クライアントの相対マウスモードを改善しました。 • CapsLock キーの同期に関する問題を修正しました。

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
• nice-dcv-server: 11591 • nice-dcv-client (Windows): 7801 • nice-dcv-viewer (macOS): 3829 • nice-dcv-viewer (Linux): 3829 • nice-xdcv: 415 • nice-dcv-gl: 952 • nice-dcv-gltest: 284 • nice-dcv-simple-external-authenticator: 176	Amazon DCV に次の機能を追加しました。 • 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。 • EC2 G5 と G5G インスタンスがサポートされました。 • 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。	• 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。 • nice-dcv-ext-Authenticator に Python 3 が必要になりました。

DCV 2021.2-11445 — 2021 年 11 月 18 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 11445 • nice-dcv-client (Windows): 7792 • nice-dcv-viewer (MacOS): 3797 • nice-dcv-viewer (Linux): 3797 • nice-xdcv: 411 • nice-dcv-gl: 946 • nice-dcv-gltest: 279 • nice-dcv-simple-external-authenticator: 160	修正内容: • 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。 • Windows におけるサーバーのセキュリティが強化されました。 • 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。 • `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。

ビルド番号	変更とバグ修正
	• 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。

DCV 2021.2-11190 — 2021 年 10 月 11 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 11190 • nice-dcv-client (Windows): 7788 • nice-dcv-viewer (macOS): 3776 • nice-dcv-viewer (Linux): 3776 • nice-xdcv: 411 • nice-dcv-gl: 946 • nice-dcv-gltest: 279 • nice-dcv-simple-external-authenticator: 160	修正内容: • 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。 • 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。 • 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。 • 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。

DCV 2021.2-11135 — 2021 年 9 月 24 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 11135 • nice-dcv-client (Windows): 7781	修正内容:

ビルド番号	変更とバグ修正
• nice-dcv-viewer (macOS): 3740 • nice-dcv-viewer (Linux): 3740 • nice-xdcv: 408 • nice-dcv-gl: 944 • nice-dcv-gltest: 279 • nice-dcv-simple-external-authenticator: 160	• 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。 • 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。 • 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。 • 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。 • 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。 • 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。 • 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。

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
• nice-dcv-server: 11048 • nice-dcv-client (Windows): 7774	Amazon DCV に次の機能を追加しました。	変更: • Amazon DCV ウェブクライアントは、Linux の個別

ビルド番号	新機能	変更とバグ修正
• nice-dcv-viewer (macOS): 3690 • nice-dcv-viewer (Linux): 3690 • nice-xdcv: 406 • nice-dcv-gl: 944 • nice-dcv-gltest: 279 • nice-dcv-simple-external-authenticator: 160	• 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 Amazon 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。 • 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 Amazon 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。 • ストリーミングの質の強化。特に、QUIC プロトコルを使用した場合の「ロスレス構築」のパフォーマンスを向上させることで、ストリーミングの質を強化しました。 • クライアントの動作を指定する `certificate-validation-policy` オプションが追加されました。これは、自己署名証	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。 • 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。 • Amazon 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。 • Amazon 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。 • OpenSSL がバージョン 1.1.1 に更新されました。 • Xdcv が XServer のバージョン 1.20.13 に更新されました。 修正内容:

ビルド番号	新機能	変更とバグ修正
	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。 • ランタイム時にオーディオドライバーで設定されたチャネル数は変更できます。 • `dcvinput` Xorg モジュールに Pressure2K 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。 • Google Chrome と Microsoft Edge の実験的な WebCodecs API に対する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、Amazon 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。	• 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。 • 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。 • 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。 • 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。

DCV 2021.1-10851 — 2021 年 7 月 30 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 10851 • nice-dcv-client (Windows): 7744 • nice-dcv-viewer (macOS): 3590 • nice-dcv-viewer (Linux): 3560 • nice-xdcv: 392 • nice-dcv-gl: 937 • nice-dcv-gltest: 275 • nice-dcv-simple-external-authenticator: 154	変更: • 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。 修正内容: • 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。 • 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。 • 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。 • 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。 • 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。

DCV 2021.1-10598 — 2021 年 6 月 10 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 10598 • nice-dcv-client (Windows): 7713 • nice-dcv-viewer (macOS): 3473	• 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。

ビルド番号	変更とバグ修正
• nice-dcv-viewer (Linux): 3473 • nice-xdcv: 392 • nice-dcv-gl: 937 • nice-dcv-gltest: 275 • nice-dcv-simple-external-authenticator: 154	• 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。

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
• nice-dcv-server: 10557 • nice-dcv-client (Windows): 7713 • nice-dcv-viewer (macOS): 3450 • nice-dcv-viewer (Linux): 3454 • nice-xdcv: 392 • nice-dcv-gl: 937 • nice-dcv-gltest: 275 • nice-dcv-simple-external-authenticator: 154	• Amazon 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。 • Amazon 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。	• 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。 • 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。 • 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。 • 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。

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
• nice-dcv-server: 10242 • nice-dcv-client (Windows): 7643 • nice-dcv-viewer (macOS): 3186 • nice-dcv-viewer (Linux): 3294 • nice-xdcv: 380 • nice-dcv-gl: 912 • nice-dcv-gltest: 266 • nice-dcv-simple-external-authenticator: 134	• Windows Amazon 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。 • Linux Amazon 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。 • 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。 • 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。	• 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。 • Linux Amazon 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。 • 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。 • 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。 • 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。 • `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

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す。

DCV 2020.2-9662 — 2020 年 12 月 4 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 9662 • nice-dcv-client (Windows): 7490 • nice-dcv-viewer (macOS): 2117 • nice-dcv-viewer (Linux): 3007 • nice-xdcv: 359 • nice-dcv-gl: 881 • nice-dcv-gltest: 259 • nice-dcv-simple-external-authenticator: 125	• 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。 • 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。 • 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。

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
• nice-dcv-server: 9508 • nice-dcv-client (Windows): 7459 • nice-dcv-viewer (macOS): 2078 • nice-dcv-viewer (Linux): 1737 • nice-xdcv: 359 • nice-dcv-gl: 881 • nice-dcv-gltest: 259	• 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。 • SLES 15 と Ubuntu 20.4 のサポートが追加されました。 • Windows Amazon 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。	• 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 Amazon DCV フレームレートリミッターを 60 FPS に変更しました。 • NVIDIA GPU がある EC2 インスタンスでホストされている Windows Amazon DCV サーバーで使用される GPU

ビルド番号	新機能	変更とバグ修正
• nice-dcv-simple-external-authenticator: 125		および CPU リソースを最適化しました。 • `list-endpoints` Amazon 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。 • `version` Amazon DCV CLI コマンドは `--json` オプションに対応しています。 • Linux サーバーにおいて、 `create-session` Amazon 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。 • Linux Amazon 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。 • 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。 • 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。

DCV 2020.1-9012 — 2020 年 9 月 30 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 9012 • nice-dcv-client (Windows): 7342 • nice-dcv-viewer (macOS): 1986 • nice-dcv-viewer (Linux): 1545 • nice-xdcv: 338 • nice-dcv-gl: 840 • nice-dcv-gltest: 246 • nice-dcv-simple-external-authenticator: 111	• 欠落している macOS クライアントアイコンを追加しました。

DCV 2020.1-9012 — 2020 年 8 月 24 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 9012 • nice-dcv-client (Windows): 7342 • nice-dcv-viewer (macOS): 1910 • nice-dcv-viewer (Linux): 1545 • nice-xdcv: 338 • nice-dcv-gl: 840 • nice-dcv-gltest: 246 • nice-dcv-simple-external-authenticator: 111	• AWS GovCloud リージョンでの Amazon S3 アクセスを修正しました • ウェブベースのクライアントの改善

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
• nice-dcv-server: 8942	• Linux Amazon DCV サーバーは、M6g、C6g、R6g	• GPU がない Amazon EC2 インスタンスにおいて新し

ビルド番号	新機能	変更とバグ修正
• nice-dcv-client (Windows): 7342 • nice-dcv-viewer (macOS): 1910 • nice-dcv-viewer (Linux): 1545 • nice-xdcv: 338 • nice-dcv-gl: 840 • nice-dcv-gltest: 246 • nice-dcv-simple-external-authenticator: 111	などの 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。 M6g, C6g 詳細については、「AWS Graviton プロセッサ」を参照してください。 • Linux Amazon 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。 • Windows Amazon DCV サーバーと Windows Amazon 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。 • macOS と Linux のネイティブ Amazon 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。 • Linux Amazon DCV サーバーと Linux Amazon 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。 • Linux Amazon 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。 • カスタム名を Amazon DCV セッションに関連付けるこ	い Amazon DCV 仮想ディスプレイドライバーのサポートを追加しました。 • 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。 • コンソールセッションが存在する場合は常に `dcv list-sessions` コマンドに含まれるようになりました。 • 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。 • `dcv://` 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。 • 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。 • DCV-GL での視覚要素と fbconfig の選択を改善し、

ビルド番号	新機能	変更とバグ修正
	とができるようになりました。 • macOS ネイティブ Amazon DCV クライアントにおけるハードウェアアクセラレーションデコードおよびレンダリングのサポート。	一部のアプリケーションのサポートを改善しました。 • ファイル転送中の CPU 使用率を低下させました。 • 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。

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
• nice-dcv-server: 8428 • nice-dcv-client (Windows): 7238 • nice-dcv-viewer (macOS): 1716 • nice-dcv-viewer (Linux): 1358 • nice-xdcv: 296 • nice-dcv-gl: 759 • nice-dcv-gltest: 229 • nice-dcv-simple-external-authenticator: 87	• 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。 • 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。 • 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。 • 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。 • 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	• 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。 • 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。 • 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。 • レガシー NVIDIA NVIFR ライブラリのサポートを削除しました。 • 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。

ビルド番号	新機能	変更とバグ修正
	ライアントのディスプレイサポートを追加しました。	• Amazon EC2 のサポートを追加しました。 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。 • 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 `on-client-connected /disconnected` コマンドが用意されています。 • 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。 • 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。

DCV 2019.1-7644 — 2019 年 10 月 24 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 7644 • nice-dcv-client (Windows): 7114 • nice-dcv-viewer (macOS): 1535 • nice-dcv-viewer (Linux): 1124 • nice-xdcv: 226	• NICE EnginFrame と他のセッションマネージャーが使用する統合 API の問題が修正されました。 • 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。

ビルド番号	変更とバグ修正
• nice-dcv-gl: 544 • nice-dcv-gltest: 220 • nice-dcv-simple-external-authenticator: 77	

DCV 2019.1-7423 — 2019 年 9 月 10 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 7423 • nice-dcv-client (Windows): 7087 • nice-dcv-viewer (macOS): 1535 • nice-dcv-viewer (Linux): 1124 • nice-xdcv: 226 • nice-dcv-gl: 544 • nice-dcv-gltest: 220 • nice-dcv-simple-external-authenticator: 77	• Windows 上の DCV サーバーのセキュリティが強化されました。 • 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。 • 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。

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
• nice-dcv-server: 7318 • nice-dcv-client (Windows): 7059 • nice-dcv-viewer (macOS): 1530 • nice-dcv-viewer (Linux): 968 • nice-xdcv: 224 • nice-dcv-gl: 529 • nice-dcv-gltest: 218	• ウェブクライアントでのマルチモニターをサポート。 • Windows Server 2019 でのスタイラス入力をサポート。 • 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。	• 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。 • 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。 • 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

ビルド番号	新機能	変更とバグ修正
• nice-dcv-simple-external-authenticator: 72	• 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)。	クから Wi-Fi ネットワークへの変更時の対応など)。 • 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。 • 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。 • 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。 • 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。 • 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。 • 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。 • 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。 • 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。

ビルド番号	新機能	変更とバグ修正
		• 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。

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
• nice-dcv-server: 6898 • nice-dcv-client (Windows): 6969 • nice-dcv-viewer (macOS): 1376 • nice-dcv-viewer (Linux): 804 • nice-xdcv: 210 • nice-dcv-gl: 490 • nice-dcv-gltest: 216 • nice-dcv-simple-external-authenticator: 70	• macOS 用の新しいネイティブクライアント。	• 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。 • `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。 • `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。 • 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。 • 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。 • 通信障害に対する Reprise ライセンスマネージャーと

ビルド番号	新機能	変更とバグ修正
		の対話の堅牢性が向上しました。 • 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。 • 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。 • 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。

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
• nice-dcv-server: 6698 • nice-dcv-client: 5946 • nice-dcv-viewer (Linux): 683 • nice-xdcv: 207 • nice-dcv-gl: 471 • nice-dcv-gltest: 210 • nice-dcv-simple-external-authenticator: 66	• Kerberos (GSSAPI) 認証のサポートが追加されました。 • 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。 • 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	• Y'UV444 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。 • EL6 RPM には NVENC エンコーダーモジュールが搭載されました。 • 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。

ビルド番号	新機能	変更とバグ修正
	セッションを自動的にロック解除します。	• Yubikey USB デバイスが許可リストに追加されました。 • 日本語キーボードのサポートが向上しました。 • 入力の認可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 `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。 • 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。 • 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。 • 同時ファイルストレージ転送の処理が向上しました。

ビルド番号	新機能	変更とバグ修正
		• 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す。 • 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。 • 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。 • 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。 • 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。 • NVIDIA ドライバ 410.xx と DCV-GL の非互換性が修正されました。

ビルド番号	新機能	変更とバグ修正
		• 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。

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
• nice-dcv-server: 6182 • nice-dcv-client: 5890 • nice-dcv-viewer (Linux): 503 • nice-xdcv: 180 • nice-dcv-gl: 427 • nice-dcv-gltest: 201 • nice-dcv-simple-external-authenticator: 58	• 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。 • スマートカードのパフォーマンスが向上しました。 • 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。	• キーボードの処理に関する改善とバグ修正。 • 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。 • 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。 • 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。 • 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

ビルド番号	新機能	変更とバグ修正
		実行するオプションが追加されました。 • `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 エンコーディングを配信するようにサーバーが設定されます。 • 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。 • 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。 • Windows クライアントはレガシー `.dcv` 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)。 • 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。 • 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。

ビルド番号	新機能	変更とバグ修正
		• 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。

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
• nice-dcv-server: 5870 • nice-dcv-client: 5813 • nice-dcv-viewer (Linux): 450 • nice-xdcv: 170 • nice-dcv-gl: 366 • nice-dcv-gltest: 198 • nice-dcv-simple-external-authenticator: 53	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。	• 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。 • 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。 • エラーの可能性がある状態での口グ記録が削減されました。

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
• nice-dcv-server: 5777 • nice-dcv-client: 5777 • nice-dcv-viewer (Linux): 438 • nice-xdcv: 166 • nice-dcv-gl: 366 • nice-dcv-gltest: 189 • nice-dcv-simple-external-authenticator: 51	• Linux ネイティブクライアントが追加されました。 • 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。 • 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。	• Linux 版でのパフォーマンスの向上。 • 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。 • Linux でのスマートカードのサポートが向上しました。 • 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。

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
• nice-dcv-server: 5600 • nice-dcv-client: 5600 • nice-xdcv: 160 • nice-dcv-gl: 279 • nice-dcv-gltest: 184 • nice-dcv-simple-external-authenticator: 48	• Linux でのマルチモニターのサポートが追加されました。 • Windows クライアントのパフォーマンスが向上しました。 • Chrome 66 以降で新しいクリップボード API が使用されます。	• 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。 • 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。

ビルド番号	新機能	変更とバグ修正
	• Windows 用 NVENC エンコーダーが追加されました。	• NumPad とブロックされた修飾子に関連するキーボードの問題が修正されました。 • 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。 • 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。

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
• nice-dcv-server: 5121 • nice-dcv-client: 5121 • nice-xdcv: 146 • nice-dcv-gl: 270 • nice-dcv-gltest: 184 • nice-dcv-simple-external-authenticator: 46	• Windows ネイティブクライアントが DPI 対応になりました。 • 相対マウス移動モードのサポートが追加されました。	• Linux の Ansys cfx5solve でのハングを防止しました。 • 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。 • 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。 • 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。 • RHEL6 の外部認証システムが修正されました。

DCV 2017.0-4334 — 2018 年 1 月 24 日

ビルド番号	変更とバグ修正
• nice-dcv-server: 4334 • nice-dcv-client: 4334 • nice-xdcv: 137 • nice-dcv-gl: 254 • nice-dcv-gltest: 184 • nice-dcv-simple-external-authenticator: 45	• キーボードの処理が向上しました。 • 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。 • 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。 • 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。 • 仮想セッションでのフォントサポートの向上。

DCV 2017.0-4100 — 2017 年 12 月 18 日

ビルド番号
• nice-dcv-server: 4100 • nice-dcv-client: 4100 • nice-xdcv: 118 • nice-dcv-gl: 229 • nice-dcv-gltest: 158 • nice-dcv-simple-external-authenticator: 35

ドキュメント履歴

以下の表は、Amazon DCV の今回のリリースのドキュメント内容をまとめたものです。

変更	説明	日付
Amazon DCV バージョン 2024.0	2024.0 の Amazon DCV 修正。詳細については、「???」を参照してください。	2025 年 6 月 10 日
Amazon DCV バージョン 2024.0	2024.0 の Amazon DCV 修正。詳細については、「DCV 2024.0-19030 — 2025 年 6 月 10 日」を参照してください。	2025 年 5 月 16 日
Amazon DCV バージョン 2024.0	2024.0 の Amazon DCV 修正。詳細については、「DCV 2024.0-19030 — 2025 年 3 月 31 日」を参照してください。	2025 年 3 月 31 日
Amazon DCV バージョン 2024.0	2024.0 の Amazon DCV 修正。詳細については、「DCV 2024.0-18131 — 2024 年 10 月 31 日」を参照してください。	2024 年 10 月 31 日
Amazon DCV バージョン 2024.0	Amazon DCV 2024.0 が利用可能になりました。詳細については、「DCV 2024.0-17979 — 2024 年 10 月 1 日」を参照してください。	2024 年 10 月 1 日
Amazon DCV バージョン 2023.1	Amazon DCV の 2023.1 で修正が行われました。詳細については、「DCV 2023.1-17701 — 2024 年 9 月 10 日」を参照してください。	2024 年 9 月 10 日
Amazon DCV バージョン 2023.1	Amazon DCV の 2023.1 で修正が行われました。詳細については、「DCV 2023.1-17701	2024 年 8 月 20 日

変更	説明	日付
	— 2024 年 8 月 20 日」を参照してください。	
Amazon DCV バージョン 2023.1	Amazon DCV の Windows、macOS、Linux クライアント 2023.1 で修正が行われました。詳細については、「DCV 2023.1-16388 — 2024 年 7 月 3 日」を参照してください。	2024 年 7 月 3 日
Amazon DCV バージョン 2023.1	Amazon DCV の Windows クライアント 2023.1 で修正が行われました。詳細については、「DCV 2023.1-16388 — 2024 年 3 月 5 日」を参照してください。	2024 年 3 月 5 日
Amazon DCV バージョン 2023.1	Amazon DCV の 2023.1 で修正が行われました。詳細については、「DCV 2023.1-16388 – 2023 年 12 月 19 日」を参照してください。	2023 年 12 月 19 日
Amazon DCV バージョン 2023.1	Amazon DCV 2023.1 が利用可能になりました。詳細については、「DCV 2023.1-16220 — 2023 年 11 月 9 日」を参照してください。	2023 年 11 月 9 日
Amazon DCV バージョン 2023.0	Amazon 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

変更	説明	日付
Amazon DCV バージョン 2023.0	Amazon DCV の 2023.0 で修正が行われました。詳細については、「DCV 2023.0-15487 — 2023 年 6 月 29 日」を参照してください。	2023 年 6 月 29 日
Amazon DCV バージョン 2023.0	Amazon DCV の 2023.0 で修正が行われました。詳細については、「DCV 2023.0-15065 — 2023 œ年 5 月 3 日」を参照してください。	2023 年 5 月 3 日
Amazon DCV バージョン 2023.0	Amazon DCV の 2023.0 で更新とで修正が行われました。詳細については、「DCV 2023.0-15022 — 2023 年 4 月 21 日」を参照してください。	2023 年 4 月 21 日
Amazon DCV バージョン 2023.0	Amazon DCV 2023.0 が利用可能になりました。詳細については、「DCV 2023.0-14852 — 2023 年 3 月 28 日」を参照してください。	2023 年 3 月 28 日
Amazon DCV バージョン 2022.2	Amazon DCV 2022.2 が利用可能になりました。詳細については、「DCV 2022.2-13907 — 2022 年 11 月 11 日」を参照してください。	2022 年 11 月 11 日
Amazon DCV バージョン 2022.1	Amazon DCV 2022.1 が利用可能になりました。詳細については、「DCV 2022.1-13067 — 2022 年 6 月 29 日」を参照してください。	2022 年 1 月 29 日

変更	説明	日付
Amazon DCV バージョン 2022.0	Amazon DCV 2022.0 が利用可能になりました。詳細については、「DCV 2022.0-11954 — 2022 年 2 月 23 日」を参照してください。	2022 年 2 月 23 日
Amazon DCV バージョン 2021.3	Amazon DCV 2021.3 が利用可能になりました。詳細については、「DCV 2021.3-11591 — 2021 年 12 月 20 日」を参照してください。	2021 年 12 月 20 日
Amazon DCV バージョン 2021.2	Amazon DCV 2021.2 が利用可能になりました。詳細については、「DCV 2021.2-11048 — 2021 年 9 月 1 日」を参照してください。	2021 年 9 月 1 日
Amazon DCV バージョン 2021.1	Amazon DCV 2021.1 が利用可能になりました。詳細については、「DCV 2021.1-10557 — 2021 年 5 月 31 日」を参照してください。	2021 年 5 月 31 日
Amazon DCV バージョン 2021.0	Amazon DCV 2021.0 が利用可能になりました。詳細については、「DCV 2021.0-10242 — 2021 年 4 月 12 日」を参照してください。	2021 年 4 月 12 日

変更	説明	日付
Amazon DCV ウェブクライアント SDK	Amazon DCV ウェブクライアント SDK が利用可能になりました。Amazon DCV ウェブクライアント SDK とは、独自の Amazon DCV ウェブブラウザクライアントアプリケーションの開発に使用できる JavaScript ライブラリです。エンドユーザーはこのアプリケーションを使用して、実行中の Amazon DCV セッションへの接続と操作を行うことができます。詳細については、「Amazon DCV ウェブクライアント SDK デベロッパーガイド」を参照してください。	2021 年 3 月 24 日
Amazon DCV バージョン 2020.2	Amazon DCV 2020.2 が利用可能になりました。詳細については、「DCV 2020.2-9508 — 2020 年 11 月 11 日」を参照してください。	2020 年 11 月 11 日
Amazon DCV バージョン 2020.1	Amazon DCV 2020.1 が利用可能になりました。詳細については、「DCV 2020.1-8942 — 2020 年 8 月 3 日」を参照してください。	2020 年 8 月 3 日

変更	説明	日付
Amazon DCV バージョン 2020.0	Amazon DCV 2020.0 には、サラウンドサウンド 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、「Amazon DCV 管理者ガイド」の「Amazon DCV サーバーのインストール」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	Amazon DCV サーバーは、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す。	2019 年 8 月 26 日
macOS クライアント	Amazon 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、「Amazon DCV ユーザーガイド」の「macOS クライアント」を参照してください。	2019 年 4 月 18 日
スマートカードのキャッシュ	Amazon 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、「Amazon DCV 管理者ガイド」の「スマートカードのキャッシュの設定」を参照してください。	2018 年 10 月 8 日

変更	説明	日付
Linux クライアント	Amazon 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、「Amazon DCV ユーザーガイド」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、「Amazon DCV 管理者ガイド」の「Amazon 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。	2018 年 8 月 7 日
USB リモート化	Amazon 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または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、「Amazon DCV 管理者ガイド」の「USB リモート化の有効化」を参照してください。	2018 年 8 月 7 日
Amazon DCV の初期リリース	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。