Dell Command | Endpoint Configure for Microsoft Intune User Guide

Dell Command | Endpoint Configure for Microsoft Intune User Guide

Notes, cautions, and warnings

ⓘ **NOTE**: A NOTE indicates important information that helps you make better use of your product.

⚠ **CAUTION**: **A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.**

⚠ **WARNING**: **A WARNING indicates a potential for property damage, personal injury, or death.**

Contents

Introduction to Dell Command | Endpoint Configure for Microsoft Intune (DCECMI)

Dell Command | Endpoint Configure for Microsoft Intune (DCECMI) enables you to manage and configure BIOS easily and securely with Microsoft Intune. The software uses Binary Large Objects (BLOBs) to store data, configure, and manage Dell system BIOS settings with zero touch, and set and maintain unique passwords. For more information on Microsoft Intune, see Endpoint management documentation in **Microsoft Learn**.

Accessing Dell Command | Endpoint Configure for Microsoft Intune installer

Prerequisites
The installation file is available as a Dell Update Package (DUP) at **Support | Dell**.

Steps

1. Go to **Support | Dell**.
2. Under **Which product can we help you with**, enter the **Service Tag** of your supported Dell device and click **Submit**, or click **Detect personal computer**.
3. On the Product Support page for your Dell device, click **Drivers & Downloads**.
4. Click **Manually find a specific** driver for your model.
5. Check **System Management** checkbox under **Category** drop-down.
6. Locate **Dell Command | Endpoint Configure for Microsoft Intune** in the list and select **Download** on the right side of the page.
7. Locate the downloaded file on your system (in Google Chrome, the file appears at the bottom of the Chrome

window), and run the executable file.

8. Follow the steps in **[Installing DCECMI using the installation wizard](#)**.

Prerequisites for Microsoft Intune Dell BIOS management

- You must have a Dell commercial client with Windows 10 or later operating system.
- The device must be enrolled into Intune mobile device management (MDM).
- NET 6.0 Runtime for Windows x64 must be installed on the device.
- Dell Command | Endpoint Configure for Microsoft Intune (DCECMI) must be installed.

Important notes

- Intune application deployment can also be used to deploy .NET 6.0 Runtime and DCECMI applications to the endpoints.
- Enter command dotnet –list-runtimes in command prompt to check if .**NET 6.0 runtime for Windows x64** is installed on the device.
- Only administrative users can install, modify, or uninstall the DCECMI application.

Supported Platforms

- OptiPlex
- Latitude
- XPS Notebook
- Precision

Supported operating systems for Windows

- Windows 10 (64-bit)
- Windows 11 (64-bit)

Installing DCECMI using the installation wizard

Steps

1. Download the DCECMI Dell update package from **[Support | Dell](#)**.
2. Double-click the downloaded installer file.
3. Click **Yes** when prompted to allow the application to make changes to your device.
4. Click **Install**.
5. Click **Next**.
6. Read and **Accept** the license agreement.
7. Click **Install**.

   The application starts to install on your device.
8. Click **Finish**.

   To verify the installation, go to **Control Panel** and see if **Dell Command | Endpoint Configure for Microsoft Intune** is displayed in the list of applications.

Installing DCECMI in silent mode

Steps

1. Go to the folder where you have downloaded DCECMI.
2. Open the command prompt as administrator.
3. Run the following command: Dell-Command-Endpoint-Configure-for-Microsoft-Intune_XXXXX_WIN_X.X.X_AXX.exe /s.

   ⓘ**NOTE**: For more information about using commands, enter the following command: Dell-Command-Endpoint-Configure-for-Microsoft-Intune_XXXXX_WIN_X.X.X_AXX.exe/?

Deploying an application package to Microsoft Intune

Prerequisites

To create and deploy a Dell Command | Endpoint Configure for Microsoft Intune Win32 application using Microsoft Intune, prepare the application package using **Microsoft Win32 Content Prep Tool** and upload it.

Steps

1. Download **Microsoft Win32 Content Prep Tool** from **Github** and extract the tool.
2. Prepare the input file by following these steps:

   a. Follow the steps in **Accessing Dell Command | Endpoint Configure for Microsoft Intune installer** .

   b. Locate the .exe file and double-click it.

   c. Click **Extract** to extract the contents to a folder.

   d. Create a **source** folder and then copy the MSI file that you got from the previous step to the **source** folder.

   e. Create another folder called **output** to save the IntuneWinAppUtil output.

   f. Go to IntuneWinAppUtil.exe in Command prompt and run the application.

   g. When prompted, enter the following details:

Table 1. Win32 application details

Option	What to enter
Please specify the source folder	<specify the source folder location>
Please specify the setup file	DCECMI.msi
Please specify the output folder	<specify the output folder location>
Do you want to specify catalog folder (Y/N)?	N

Uploading an application package to Microsoft Intune

Steps

1. Log in to Microsoft Intune with a user who has the **Application Manager** role assigned.
2. Go to **Apps > Windows apps**.
3. Click **Add**.
4. In the **App type** dropdown, select **Windows app (Win32)**.
5. Click **Select**.

6. In **App information** tab, click **Select app package** file and select the IntuneWin file that is created using Win32 Content Prep Tool.

7. Click **OK**.

8. Review the rest of the details in the **App information** tab.

9. Enter the details that are not automatically populated:

Table 2. App information details

Options	What to enter
Publisher	Dell
Category	Computer management

10. Click **Next**.

    In the **Program** tab, the **Install commands** and **Uninstall commands** fields are automatically populated.

11. Click **Next**.

    In the **Requirements** tab, select **64-bit** from the **Operating system architecture** dropdown and the Windows operating system version that is based on your environment from the **Minimum operating system** dropdown.

12. Click **Next**.

    In the **Detection rule** tab, do the following:

    a. In the **Rules format** dropdown, select **Manually configure detection rules**.

    b. Click **+Add** and select **MSI** from the **Rule type** dropdown, which populates the **MSI product code** field.

    c. Click **OK**.

13. Click **Next**.

    In the **Dependencies** tab, click **+Add** and select **dotnet-runtime-6.x.x-win-x64.exe** as dependencies. See [Creating and deploying DotNet Runtime Win32 Application from Intune](#) for more information.

14. Click **Next**.

15. In **Supersedence** tab, select **No Supersedence** if you have not created any lower version of the application. Otherwise,select the lower version that must be superseded.

16. Click **Next**.

17. In the **Assignments** tab, click **+Add group** to select the device group for which the application is required. The required applications are installed automatically on enrolled devices.

    ⓘ**NOTE**: If you want to uninstall DCECMI, add the respective device group to the **Excluded** list.

18. Click **Next**.

19. In the **Review + create** tab, click **Create**.

Results
Once uploaded, the DCECMI application package is available in Microsoft Intune for deployment to managed devices.

Checking the deployment status of the application package

Steps

1. Go to the Microsoft Intune admin center and sign in with a user having the **Application Manager** role assigned.

2. Click **Apps** in the navigation menu on the left.

3. Select **All apps**.

4. Locate and open the **Dell Command | Endpoint Configure for Microsoft Intune Win32** application.

5. Open the details page.

6. On the details page, click the **Device install status** tab.

   You can see the installation status of the DCECMI application on different devices.

Creating and deploying DotNet Runtime Win32 Application from Intune

To create and deploy a DotNet Runtime Win32 application using Intune, do the following:

1. Download **Microsoft Win32 Content Prep Tool** from **Github** and extract the tool.

2. Prepare the input file by following these steps:

   a. Download the latest DotNet Runtime 6.x.x from **Microsoft .NET**.

   b. Create a folder that is called **Source** and then copy the .exe file to the **Source** folder.

   c. Create another folder called **output** to save the IntuneWinAppUtil output.

   d. Go to IntuneWinAppUtil.exe in Command prompt and run the application.

   e. When prompted, enter these details:

   Table 3. Input details

Options	What to enter
Please specify the source folder	<specify the source folder location>
Please specify the setup file	dotnet-runtime-6.x.x-win-x64.exe
Please specify the output folder	<specify the output folder location>
Do you want to specify catalog folder (Y/N)?	N

   A dotnet-runtime-6.x.x-win-x64.intunewin package is created in the **output** folder.

3. Upload the DotNet intunewin package to Intune by following these steps:

   a. Log in to Microsoft Intune with a user who has the **Application Manager** role assigned.

   b. Go to **Apps > Windows apps** .

   c. Click **Add**.

   d. In the **App type** dropdown, select **Windows app (Win32)**.

   e. Click **Select**.

   f. In **App information** tab, click **Select app package file** and select the IntuneWin file that is created using Win32 Content Prep Tool.

   g. Click **OK**.

   h. Review the rest of the details in the **App information** tab.

   i. Enter the details, which are not automatically populated:

   Table 4. Input details

Options	What to enter
Publisher	Microsoft
App version	6.x.x

   j. Click **Next**.

   Program tab opens where you must add the **Install commands** and **Uninstall commands**:

   ● **Install commands**: powershell.exe -executionpolicy bypass .\dotnet-runtime-6.x.x-win- x64.exe /install /quiet /norestart

   ● **Uninstall commands**: powershell.exe -executionpolicy bypass .\dotnet-runtime-6.x.x-win- x64.exe /uninstall

/quiet /norestart

k. Click **Next**.

Requirements tab opens where you must select **64-bit** from the **Operating system architecture** dropdown and the Windows operating system version that is based on your environment from the **Minimum operating system** dropdown.

l. Click **Next**.

Detection rule tab opens where you must do the following:

- In **Rules format** dropdown, select **Manually configure detection rules**.
- Click **+Add**.
- Under **Detection rules**, select **File** as **Rule type**.
- Under **Path**, enter the complete path of the folder: **C:\Program theFiles\dotnet\shared\Microsoft.NETCore.App\6.x.x.**.
- Under **File or folder**, enter the folder name to detect.
- Under **Detection method**, select **File or folder exists**.
- Click **OK**.

m. Click **Next**.

Dependencies tab opens where you can select **No dependencies.**

n. Click **Next**.

In **Supersedence** tab, select **No Supersedence** if you have not created any lower version of the application. Otherwise, select the lower version that must be superseded.

o. Click **Next**.

Assignments tab opens where you must click **+Add group** to select the device group for which the application is required. The required applications are installed automatically on enrolled devices.

p. Click **Next**.

Review + create tab opens where you must click **Create**.

Once uploaded, the DotNet Runtime application package is available in Microsoft Intune for deployment to managed devices.

Checking the deployment status of the application package

To check the deployment status of the application package, do the following:

1. Go to the Microsoft Intune admin center and sign in with a user who has the **Application Manager** role assigned.
2. Click **Apps** in the navigation menu on the left.
3. Select **All apps**.
4. Locate the DotNet Runtime Win32 application, and click its name to open the details page.
5. On the details page, click the **Device install status** tab.

You can see the installation status of DotNet Runtime Win32 on different devices.

Uninstalling Dell Command | Endpoint Configure for Microsoft Intune for systems running on Windows

1. Go to **Start > Settings > Apps > Apps and Features.**
2. Select **Add/Remove Programs.**

ⓘ **NOTE**: You can also uninstall DCECMI from Intune. If you want to uninstall DCECMI, add the respective device group to the **Excluded** list, which can be found in the **Assignments** tab of Microsoft Intune. See Uploading an application package to Microsoft Intune for more details.

Contacting Dell

Prerequisites

ⓘ **NOTE**: If you do not have an active internet connection, you can find contact information on your purchase invoice, packing slip, bill, or Dell product catalog.
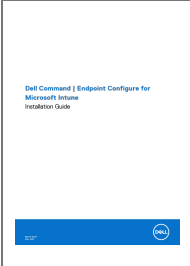
About this task

Dell provides several online and telephone-based support and service options. Availability varies by country and product, and some services may not be available in your area. To contact Dell sales, technical support, or customer service issues:

Steps

1. Go to Support | Dell.
2. Select your support category.
3. Verify your country or region in the **Choose a Country/Region** drop-down list at the bottom of the page.
4. Select the appropriate service or support link based on your need.

Documents / Resources

Dell Command | Endpoint Configure for Microsoft Intune Installation Guide	**DELL Dell Command | Endpoint Configure for Microsoft Intune** [pdf] User Guide Endpoint Configure For Microsoft Intune, Endpoint Configure For Microsoft Intune, Configure For Microsoft Intune, Microsoft Intune, Intune

References

- **.NET | Build. Test. Deploy.**
- **GitHub: Let's build from here · GitHub**
- **Microsoft Learn: Build skills that open doors in your career**
- **User Manual**