The **dtutil** command prompt utility is used to manage SQL Server Integration Services packages. The utility can copy, move, delete, or verify the existence of a package. These actions can be performed on any SSIS package that is stored in one of three locations: a Microsoft SQL Server database, the SSIS Package Store, and the file system. If the utility accesses a package that is stored in **msdb**, the command prompt may require a user name and a password. If the instance of SQL Server uses SQL Server Authentication, the command prompt requires both a user name and a password. If the user name is missing, **dtutil** tries to log on to SQL Server using Windows Authentication. The storage type of the package is identified by the /SQL, /FILE, and /DTS options.

The **dtutil** command prompt utility does not support the use of command files or redirection.

The **dtutil** command prompt utility includes the following features:

* Remarks in the command prompt, which makes the command prompt action self-documenting and easier to understand.
* Overwrite protection, to prompt for a confirmation before overwriting an existing package when you are copying or moving packages.
* Console help, to provide information about the command options for **dtutil**.

|  |
| --- |
| **NoteNote** |
| Many of the operations that are performed by dtutil can also be performed visually in SQL Server Management Studio when you are connected to an instance of Integration Services. For more information, see [Package Management (Legacy SSIS Service)](http://msdn.microsoft.com/en-us/library/ms137916.aspx). |

The options can be typed in any order. The pipe ("|") character is the OR operator and is used to show possible values. You must use one of the options that are delimited by the OR pipe.

All options must start with a slash (/) or a minus sign (-). However, do not include a space between the slash or minus sign and the text for the option; otherwise, the command will fail.

Arguments must be strings that are either enclosed in quotation marks or contain no white space.

Double quotation marks within strings that are enclosed in quotation marks represent escaped single quotation marks.

Options and arguments, except for passwords, are not case sensitive.

Installation Considerations on 64-bit Computers

On a 64-bit computer, Integration Services installs a 64-bit version of the **dtexec** utility (dtexec.exe) and the **dtutil** utility (dtutil.exe). To install 32-bit versions of these Integration Services tools, you must select either Client Tools or SQL Server Data Tools (SSDT) during setup.

By default, a 64-bit computer that has both the 64-bit and 32-bit versions of an Integration Services command prompt utility installed will run the 32-bit version at the command prompt. The 32-bit version runs because the directory path for the 32-bit version appears in the PATH environment variable before the directory path for the 64-bit version. (Typically, the 32-bit directory path is <drive>:\Program Files(x86)\Microsoft SQL Server\110\DTS\Binn, while the 64-bit directory path is <drive>:\Program Files\Microsoft SQL Server\110\DTS\Binn.)

|  |
| --- |
| **NoteNote** |
| If you use SQL Server Agent to run the utility, SQL Server Agent automatically uses the 64-bit version of the utility. SQL Server Agent uses the registry, not the PATH environment variable, to locate the correct executable for the utility. |

To ensure that you run the 64-bit version of the utility at the command prompt, you can take one of the following actions:

* Open a Command Prompt window, change to the directory that contains the 64-bit version of the utility (<drive>:\Program Files\Microsoft SQL Server\110\DTS\Binn), and then run the utility from that location.
* At the command prompt, run the utility by entering the full path (<drive>:\Program Files\Microsoft SQL Server\110\DTS\Binn) to the 64-bit version of the utility.
* Permanently change the order of the paths in the PATH environment variable by placing the 64-bit path (<drive>:\Program Files\Microsoft SQL Server\110\DTS\Binn) before the 32-bit path (<drive>:\ Program Files(x86)\Microsoft SQL Server\110\DTS\Binn) in the variable.

[Syntax](javascript:void(0))

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_1f82c020-6a5f-40c1-bb7a-68542e37f35b');" \o "Copy to clipboard.)

dtutil /option [value] [/option [value]]...

[Parameters](javascript:void(0))

|  |  |
| --- | --- |
| **Option** | **Description** |
| /? | Displays the command prompt options. |
| /C[opy] location;destinationPathandPackageName | Specifies a copy action on an SSIS package. Use of this parameter requires that you first specify the location of the package using the **/FI**, **/SQ**, or **/DT** option. Next, specify the destination location destination package name. The destinationPathandPackageName argument specifies where the SSIS package is copied to. If the destination location is SQL, the DestUser, DestPassword and DestServer arguments must also be specified in the command.  When the Copy action encounters an existing package at the destination, **dtutil** prompts the user to confirm package deletion. The Y reply overwrites the package and the N reply ends the program. When the command includes the Quiet argument, no prompt appears and any existing package is overwritten. |
| /Dec[rypt] password | (Optional). Sets the decryption password that is used when you load a package with password encryption. |
| /Del[ete] | Deletes the package specified by the SQL, DTS or FILE option. If **dtutil** cannot delete the package, the program ends. |
| /DestP[assword] password | Specifies the password that is used with the SQL option to connect to a destination SQL Server instance using SQL Server Authentication. An error is generated if DESTPASSWORD is specified in a command line that does not include the DTSUSER option.   |  | | --- | | **NoteNote** | | When possible, use Windows Authentication.. | |
| /DestS[erver] server\_instance | Specifies the server name that is used with any action that causes a destination to be saved to SQL Server. It is used to identify a non-local or non-default server when saving an SSIS package. It is an error to specify DESTSERVER in a command line that does not have an action associated with SQL Server. Actions such as SIGN SQL, COPY SQL, or MOVE SQL options would be appropriate commands to combine with this option.  A SQL Server instance name can be specified by adding a backslash and the instance name to the server name. |
| /DestU[ser] username | Specifies the user name that is used with the SIGN SQL, COPY SQL, and MOVE SQL options to connect to a SQL Server instance that uses SQL Server Authentication. It is an error to specify DESTUSER in a command line that does not include the SIGN SQL, COPY SQL, or MOVE SQL option. |
| /Dump process ID | (Optional) Causes the specified process, either the **dtexec** utility or the **dtsDebugHost.exe** process, to pause and create the debug dump files, .mdmp and .tmp.   |  | | --- | | **NoteNote** | | To use the **/Dump** option, you must be assigned the Debug Programs user right (SeDebugPrivilege). |   To find the process ID for the process that you want to pause, use Windows Task Manager.  By default, Integration Services stores the debug dump files in the folder, <drive>:\Program Files\Microsoft SQL Server\110\Shared\ErrorDumps.  For more information about the **dtexec** utility and the **dtsDebugHost.exe** process, see [dtexec Utility](http://msdn.microsoft.com/en-us/library/hh231187.aspx) and [Building, Deploying, and Debugging Custom Objects](http://msdn.microsoft.com/en-us/library/ms403356.aspx).  For more information about debug dump files, see [Generating Dump Files for Package Execution](http://msdn.microsoft.com/en-us/library/hh231095.aspx).   |  | | --- | | **NoteNote** | | Debug dump files may contain sensitive information. Use an access control list (ACL) to restrict access to the files, or copy the files to a folder with restricted access. | |
| /DT[S] filespec | Specifies that the SSIS package to be operated on is located in the SSIS Package Store. The filespec argument must include the folder path, starting with the root of the SSIS Package Store. By default, the names of the root folders in the configuration file are "MSDB" and "File System." Paths that contain a space must be delimited by using double quotation marks.  If the DT[S] option is specified on the same command line as any of the following options, a DTEXEC\_DTEXECERROR is returned:   * FILE * SQL * SOURCEUSER * SOURCEPASSWORD * SOURCESERVER |
| /En[crypt] {SQL | FILE}; Path;ProtectionLevel[;password] | (Optional). Encrypts the loaded package with the specified protection level and password, and saves it to the location specified in Path. The ProtectionLevel determines whether a password is required.   * SQL - Path is the destination package name. * FILE - Path is the fully-qualified path and file name for the package. * DTS - This option is not supported currently.   ProtectionLevel options:  Level 0: Strips sensitive information.  Level 1: Sensitive information is encrypted by using local user credentials.  Level 2: Sensitive information is encrypted by using the required password.  Level 3: Package is encrypted by using the required password.  Level 4: Package is encrypted by using local user credentials.  Level 5 Package uses SQL Server storage encryption. |
| /Ex[ists] | (Optional). Used to determine whether a package exists. **dtutil** tries to locate the package specified by either the SQL, DTS or FILE options. If **dtutil** cannot locate the package specified, a DTEXEC\_DTEXECERROR is returned. |
| /FC[reate] {SQL | DTS};ParentFolderPath;NewFolderName | (Optional). Create a new folder that has the name that you specified in NewFolderName. The location of the new folder is indicated by the ParentFolderPath. |
| /FDe[lete] {SQL | DTS}[;ParentFolderPath;FolderName] | (Optional). Deletes from SQL Server or SSIS the folder that was specified by the name in FolderName. The location of the folder to delete is indicated by the ParentFolderPath. |
| /FDi[rectory] {SQL | DTS};FolderPath[;S] | (Optional). Lists the contents, both folders and packages, in a folder on SSIS or SQL Server. The optional FolderPath parameter specifies the folder that you want to view the content of. The optional S parameter specifies that you want to view a listing of the contents of the subfolders for the folder specified in FolderPath. |
| /FE[xists ] {SQL | DTS};FolderPath | (Optional). Verifies if the specified folder exists on SSIS or SQL Server. The FolderPath parameter is the path and name of the folder to verify. |
| /Fi[le] filespec | This option specifies that the SSIS package to be operated on is located in the file system. The filespec value can be provided as either a Universal Naming Convention (UNC) path or local path.  If the File option is specified on the same command line as any of the following options, a DTEXEC\_DTEXECERROR is returned:   * DTS * SQL * SOURCEUSER * SOURCEPASSWORD * SOURCESERVER |
| /FR[ename] {SQL | DTS} [;ParentFolderPath; OldFolderName;NewFolderName] | (Optional). Renames a folder on the SSIS or SQL Server. The ParentFolderPath is the location of the folder to rename. The OldFolderName is the current name of the folder, and NewFolderName is the new name to give the folder. |
| /H[elp] option | Displays text extensive help that shows the **dtutil** options and describes their use. The option argument is optional. If the argument is included, the Help text includes detailed information about the specified option. The following example displays help for all options:  dtutil /H  The following two examples show how to use the /H option to display extended help for a specific option, the /Q [uiet] option, in this example:  dtutil /Help Quiet  dtutil /H Q |
| /I[DRegenerate] | Creates a new GUID for the package and updates the package ID property. When a package is copied, the package ID remains the same; therefore, the log files contain the same GUID for both packages. This action creates a new GUID for the newly-copied package to distinguish it from the original. |
| /M[ove] {SQL | File | DTS}; pathandname | Specifies a move action on an SSIS package. To use this parameter, first specify the location of the package using the **/FI**, **/SQ**, or **/DT** option. Next, specify the **Move** action. This action requires two arguments, which are separated by a semicolon:   * The destination argument can specify SQL, FILE, or DTS. A SQL destination can include the DESTUSER, DESTPASSWORD, and DESTSERVER options. * The pathandname argument specifies the package location: SQL uses the package path and package name, FILE uses a UNC or local path, and DTS uses a location that is relative to the root of the SSIS Package Store. When the destination is FILE or DTS, the path argument does not include the file name. Instead, it uses the package name at the specified location as the file name.   When the **MOVE** action encounters an existing package at the destination, **dtutil** prompts you to confirm that you want to overwrite the package. The Y reply overwrites the package and the N reply ends the program. When the command includes the QUIET option, no prompt appears and any existing package is overwritten. |
| /Q[uiet] | Stops the confirmation prompts that can appear when a command including the COPY, MOVE, or SIGN option is executed. These prompts appear if a package with the same name as the specified package already exists at the destination computer or if the specified package is already signed. |
| /R[emark] text | Adds a comment to the command line. The comment argument is optional. If the comment text includes spaces, the text must be enclosed in quotation marks. You can include multiple REM options in a command line. |
| /Si[gn] {SQL | File | DTS}; path; hash | Signs an SSIS package. This action uses three required arguments, which are separated by semicolons:   * The destination argument can specify SQL, FILE, or DTS. A SQL destination can include the DESTUSER, DESTPASSWORD and DESTSERVER options. * The path argument specifies the location of the package to take action on. * The hash argument specifies a certificate identifier expressed as a hexadecimal string of varying length.   For more information, see [Identify the Source of Packages with Digital Signatures](http://msdn.microsoft.com/en-us/library/ms141174.aspx).   |  | | --- | | **Important noteImportant** | | When configured to check the signature of the package, Integration Services only checks whether the digital signature is present, is valid, and is from a trusted source. Integration Services does not check whether the package has been changed. | |
| /SourceP[assword] password | Specifies the password that is used with the SQL and SOURCEUSER options to enable the retrieval of an SSIS package that is stored in a database on a SQL Server instance that uses SQL Server Authentication. It is an error to specify SOURCEPASSWORD in a command line that does not include the SOURCEUSER option.   |  | | --- | | **NoteNote** | | When possible, use Windows Authentication. | |
| /SourceS[erver] server\_instance | Specifies the server name that is used with the SQL option to enable the retrieval of an SSIS package that is stored in SQL Server. It is an error to specify SOURCESERVER in a command line that does not include the SIGN SQL, COPY SQL, or MOVE SQL option.  A SQL Server instance name can be specified by adding a backslash and the instance name to the server name. |
| /SourceU[ser] username | Specifies the user name that is used with the SOURCESERVER option to enable the retrieval of an SSIS package stored in SQL Server using SQL Server Authentication. It is an error to specify SOURCEUSER in a command line that does not include the SIGN SQL, COPY SQL, or MOVE SQL option.   |  | | --- | | **NoteNote** | | When possible, use Windows Authentication. | |
| /SQ[L] package\_path | Specifies the location of an SSIS package. This option indicates that the package is stored in the **msdb** database. The package\_path argument specifies the path and name of the SSIS package. Folder names are terminated with back slashes.  If the SQL option is specified on the same command line as any of the following options, a DTEXEC\_DTEXECERROR is returned:   * DTS * FILE   The SQL option may be accompanied by zero or one instance of the following options:   * SOURCEUSER * SOURCEPASSWORD * SOURCESERVER   If SOURCEUSERNAME is not included, Windows Authentication is used to access the package. SOURCEPASSWORD is allowed only if SOURCEUSER is present. If SOURCEPASSWORD is not included, a blank password is used.   |  | | --- | | **Important noteImportant** | | Do not use a blank password. Use a strong password. | |

[dtutil Exit Codes](javascript:void(0))

**dtutil** sets an exit code that alerts you when syntax errors are detected, incorrect arguments are used, or invalid combinations of options are specified. Otherwise, the utility reports "The operation completed successfully".The following table lists the values that the **dtutil** utility can set when exiting.

|  |  |
| --- | --- |
| **Value** | **Description** |
| 0 | The utility executed successfully. |
| 1 | The utility failed. |
| 4 | The utility cannot locate the requested package. |
| 5 | The utility cannot load the requested package |
| 6 | The utility cannot resolve the command line because it contains either syntactic or semantic errors. |

[Remarks](javascript:void(0))

You cannot use command files or redirection with **dtutil**.

The order of the options within the command line is not significant.

[Examples](javascript:void(0))

The following examples detail typical command line usage scenarios.

**Copy Examples**

To copy a package that is stored in the **msdb** database on a local instance of SQL Server using Windows Authentication to the SSIS Package Store, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_e9198226-7efa-4037-9e4a-2840aa7dfc05');)

dtutil /SQL srcPackage /COPY DTS;destFolder\destPackage

To copy a package from a location on the File system to another location and give the copy a different name, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_dc82420c-538e-442e-b1d3-561bab8ddb2a');)

dtutil /FILE c:\myPackages\mypackage.dtsx /COPY FILE;c:\myTestPackages\mynewpackage.dtsx

To copy a package on the local file system to an instance of SQL Server hosted on another computer, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_8be28938-dc38-4bf2-8c76-1e01d2f28f94');)

dtutil /FILE c:\sourcepkg.dtsx /DestServer <servername> /COPY SQL;destpkgname

Because the /DestU[ser] and /DestP[assword] options were not used, Windows Authentication is assumed.

To create a new ID for a package after it is copied, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_07d3eb2b-c222-4084-80c1-29a300107bd0');)

dtutil /I /FILE copiedpkg.dtsx

To create a new ID for all the packages in a specific folder, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_f6c6b482-68c0-4de6-9ef8-bed165eecbd7');)

for %%f in (C:\test\SSISPackages\\*.dtsx) do dtutil.exe /I /FILE %%f

Use a single percent sign (%) when typing the command at the command prompt. Use a double percent sign (%%) if the command is used inside a batch file.

**Delete Examples**

To delete a package that is stored in the **msdb** database on an instance of SQL Server that uses Windows Authentication, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_6cc9026d-6185-4836-919b-7d1e22615ab8');)

dtutil /SQL delPackage /DELETE

To delete a package that is stored in the **msdb** database on an instance of SQL Server that uses SQL Server Authentication, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_dff0ee3d-3bee-436c-b5d0-247ba4e6d8ee');)

dtutil /SQL delPackage /SOURCEUSER srcUserName /SOURCEPASSWORD #8nGs\*w7F /DELETE

|  |
| --- |
| **NoteNote** |
| To delete a package from a named server, include the SOURCESERVER option and its argument. You can only specify a server by using the SQL option. |

To delete a package that is stored in the SSIS Package Store, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_646895fe-30b2-4a65-9231-1bbcf560f848');)

dtutil /DTS delPackage.dtsx /DELETE

To delete a package that is stored in the file system, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_953033b6-483b-4436-a1cb-7bd9614a5484');)

dtutil /FILE c:\delPackage.dtsx /DELETE

**Exists Examples**

To determine whether a package exists in the **msdb** database on a local instance of SQL Server that uses Windows Authentication, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_2e128ce8-1400-442a-9abe-916e21d07ad2');)

dtutil /SQL srcPackage /EXISTS

To determine whether a package exists in the **msdb** database on a local instance of SQL Server that uses SQL Server Authentication, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_9d26448a-8a47-42b2-971d-3d01b2fabd7c');)

dtutil SQL srcPackage /SOURCEUSER srcUserName /SOURCEPASSWORD \*hY$d56b /EXISTS

|  |
| --- |
| **NoteNote** |
| To determine whether a package exists on a named server, include the SOURCESERVER option and its argument. You can only specify a server by using the SQL option. |

To determine whether a package exists in the local package store, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_0413e005-cab6-4cef-bfc8-65696b621b7a');)

dtutil /DTS srcPackage.dtsx /EXISTS

To determine whether a package exists in the local file system, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_b6740284-7cdb-49bb-ac4a-96331011414c');)

dtutil /FILE c:\srcPackage.dtsx /EXISTS

**Move Examples**

To move a package that is stored in the SSIS Package Store to the **msdb** database on a local instance of SQL Server that uses Windows Authentication, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_fed0f3e9-e545-4b58-bc95-40d26d744ca5');)

dtutil /DTS srcPackage.dtsx /MOVE SQL;destPackage

To move a package that is stored in the **msdb** database on a local instance of SQL Server that uses SQL Server Authentication to the **msdb** database on another local instance of SQL Server that uses SQL Server Authentication, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_ff1f4286-e75a-458a-9b1d-44f7989c3ffe');)

dtutil /SQL srcPackage /SOURCEUSER srcUserName /SOURCEPASSWORD $Hj45jhd@X /MOVE SQL;destPackage /DESTUSER destUserName /DESTPASSWORD !38dsFH@v

|  |
| --- |
| **NoteNote** |
| To move a package from one named server to another, include the SOURCES and the DESTS option and their arguments. You can only specify servers by using the SQL option. |

To move a package that is stored in the SSIS Package Store, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_1176cf1e-6a58-45b9-bcff-7ecd0d1cc746');)

dtutil /DTS srcPackage.dtsx /MOVE DTS;destPackage.dtsx

To move a package that is stored in the file system, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_821b3e56-415d-482b-9e79-5bd4ca1711ba');)

dtutil /FILE c:\srcPackage.dtsx /MOVE FILE;c:\destPackage.dtsx

**Sign Examples**

To sign a package that is stored in a SQL Server database on a local instance of SQL Server that uses Windows Authentication, use the following syntax:

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_b9ca6c8e-1036-4c4f-a610-ccb6c7b3efa8');)

dtutil /FILE srcPackage.dtsx /SIGN FILE;destpkg.dtsx;1767832648918a9d989fdac9819873a91f919

To locate information about your certificate, use **CertMgr**. The hash code can be viewed in the **CertMgr** utility by selecting the certificate, and then clicking View to view the properties. The Details tab provides more information about the certificate. The Thumbprint property is used as the hash value, with spaces removed.

|  |
| --- |
| **NoteNote** |
| The hash used in this example is not a real hash. |

For more information, see the CertMgr section in [Signing and Checking Code with Authenticode](http://go.microsoft.com/fwlink/?LinkId=78100).

**Encrypt Examples**

The following sample encrypts the file-based PackageToEncrypt.dtsx to the file-based EncryptedPackage.dts using full package encryption, with a password. The password that is used for the encryption is EncPswd.

[Copy](javascript:if%20(window.epx.codeSnippet)window.epx.codeSnippet.copyCode('CodeSnippetContainerCode_21bc0282-a27f-4dcb-ab2d-a5975bcfaa87');)

dtutil /FILE PackageToEncrypt.dtsx /ENCRYPT file;EncryptedPackage.dtsx;3;EncPswd