



We use optional cookies to improve your experience on our websites, such as through social media connections, and to display personalized advertising based on your online activity. If you reject optional cookies, only cookies necessary to provide you the services will be used. You may change your selection by clicking "Manage Cookies" at the bottom of the page. [Privacy Statement](#) [Third-Party Cookies](#)

Accept

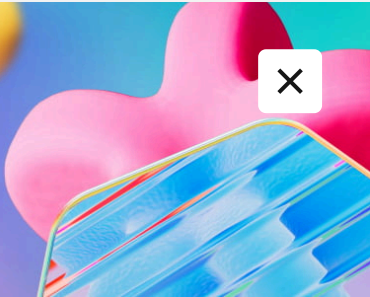
Reject

Manage cookies

# Microsoft Ignite

Nov 19–22, 2024

Register now >



## Export-PfxCertificate

Reference

Feedback

Module: [pki](#)

### In this article

- [Syntax](#)
- [Description](#)
- [Examples](#)
- [Parameters](#)
- [Show 3 more](#)

Exports a certificate or a PFXData object to a Personal Information Exchange (PFX) file.

## Syntax

```
Export-PfxCertificate
    [-NoProperties]
    [-NoClobber]
    [-Force]
    [-CryptoAlgorithmOption <CryptoAlgorithmOptions>]
    [-ChainOption <ExportChainOption>]
    [-ProtectTo <String[]>]
    [-Password <SecureString>]
    [-FilePath] <String>
    [-PFXData] <PfxData>
    [-WhatIf]
    [-Confirm]
    [<CommonParameters>]
```

```
Export-PfxCertificate
    [-NoProperties]
    [-NoClobber]
    [-Force]
    [-CryptoAlgorithmOption <CryptoAlgorithmOptions>]
    [-ChainOption <ExportChainOption>]
    [-ProtectTo <String[]>]
    [-Password <SecureString>]
    [-FilePath] <String>
    [-Cert] <Certificate>
    [-WhatIf]
    [-Confirm]
    [<CommonParameters>]
```

## Description

The `Export-PfxCertificate` cmdlet exports a certificate or a **PFXData** object to a Personal Information Exchange (PFX) file. By default, extended properties and the entire chain are exported.

Delegation may be required when using this cmdlet with Windows PowerShell remoting and changing user configuration.

## Examples

### EXAMPLE 1

```
$mypwd = ConvertTo-SecureString -String '1234' -Force -AsPlainText  
  
Get-ChildItem -Path Cert:\LocalMachine\My\5F98EBBF735CDDAE0  
Export-PfxCertificate -FilePath C:\mypfx.pfx -Password $mypwd
```

This example exports a certificate from the local machine store to a PFX file which includes the entire chain and all external properties.

### EXAMPLE 2

```
$mypwd = ConvertTo-SecureString -String '1234' -Force -AsPlainText  
  
Get-ChildItem -Path Cert:\LocalMachine\My |  
Export-PfxCertificate -FilePath C:\mypfx.pfx -Password $mypwd
```

This example exports all certificates under the My store for the machine account into one file named `mypfx.pfx`. In order for this cmdlet to succeed, all keys need to be exportable.

### EXAMPLE 3

```
powershell
$mypwd = ConvertTo-SecureString -String '1234' -Force -AsPlainText

$params = @{
    Cert = 'Cert:\CurrentUser\My\5F98EBBFE735CDDAE00E33E0FD6'
    FilePath = 'C:\myexport.pfx'
    ChainOption = 'EndEntityCertOnly'
    NoProperties = $true
    Password = $mypwd
}
Export-PfxCertificate @params
```

This example exports a certificate from the current user store with no chain and no external properties

## EXAMPLE 4

```
$a = Get-ChildItem -Path Cert:\LocalMachine\My

$params = @{
    Cert = $a[1]
    FilePath = 'C:\myexport.pfx'
    ProtectTo = 'billb99', 'johnj99'
}
Export-PfxCertificate @params
```

This example exports a certificate from the local machine store. Both user accounts, `billb99` and `johnj99`, can access this PFX with no password. A Windows Server 2012 or later domain controller is required for key distribution.

## EXAMPLE 5

```
$a = Get-ChildItem -Path Cert:\LocalMachine\My

$mypwd = ConvertTo-SecureString -String '1234' -Force -AsPlainText

$params = @{
    Cert = $a[1]
    FilePath = 'C:\myexport.pfx'
    ProtectTo = 'billb99', '\johnj99'
    Password = $mypwd
}
Export-PfxCertificate @params
```

This example exports a certificate from the local machine store. Both user accounts, `johnj99` and `billb99`, can access this PFX file with no password. For everyone else, they need to use 1234 as a password. A Windows Server 2012 or later domain controller is required for key distribution.

## EXAMPLE 6

```
$NewPwd = ConvertTo-SecureString -String 'abcd' -Force -AsPlainText

$mypfx = Get-PfxData -FilePath C:\mypfx.pfx -Password $OldPwd

Export-PfxCertificate -PFXData $mypfx -FilePath C:\mypfx2.pfx -Password $NewPwd
```

This example changes an existing password for a PFX file from `$OldPwd` to `$NewPwd`.

## Parameters

### -Cert

Specifies the path to the certificate to be exported.

[Expand table](#)

Type:	Microsoft.CertificateServices.Commands.Certificate
Aliases:	PsPath
Position:	0
Default value:	None
Required:	True
Accept pipeline input:	True
Accept wildcard characters:	False

**-ChainOption**

Specifies the options for building a chain when exporting certificates. The acceptable values for this parameter are:

- **BuildChain**: Certificate chain for all end entity certificates will be built and included in the export. This option is valid for both **PfxData** and **Cert** parameters. In the case of **PfxData** parameter, the collection of all PFX certificates will be used as an additional store.
- **EndEntityCertOnly**: Only end entity certificates are exported without any chain. This option is valid for both the **PfxData** and the **Cert** parameters.
- **PfxDataOnly**: Certificates contained in **PFXData** objects will be exported with no chain building. This option is only valid when the **PfxData** parameter is used.


 Expand table

Type:	Microsoft.CertificateServices.Commands.ExportChainOption
Accepted values:	BuildChain, EndEntityCertOnly, PfxDataOnly
Position:	Named

Default value:	None
Required:	False
Accept pipeline input:	False
Accept wildcard characters:	False

**-Confirm**

Prompts you for confirmation before running the cmdlet.

 Expand table

Type:	SwitchParameter
Aliases:	cf
Position:	Named
Default value:	False
Required:	False
Accept pipeline input:	False
Accept wildcard characters:	False

**-CryptoAlgorithmOption**

Specifies the algorithm for encrypting private keys within the PFX file. If this parameter is not specified, the default is TripleDES\_SHA1. The acceptable values for this parameter are:

- TripleDES\_SHA1: Private keys will be encrypted in the PFX file using Triple DES encryption.


- **AES256\_SHA256**: Private keys will be encrypted in the PFX file using AES-256 encryption.

 Expand table

Type:	Microsoft.CertificateServices.Commands.CryptoAlgorithmOptions
Accepted values:	TripleDES_SHA1, AES256_SHA256
Position:	Named
Default value:	None
Required:	False
Accept pipeline input:	False
Accept wildcard characters:	False

**-FilePath**

Specifies the path for the PFX file to be exported.

 Expand table

Type:	String
Position:	1
Default value:	None
Required:	True
Accept pipeline input:	False
Accept wildcard characters:	False



**-Force**

Specifies that the provided PFX file should be overwritten, even if the Read-only attribute is set on the file. By default, this cmdlet overwrites existing PFX files without warning, unless the Read-only or hidden attribute is set or the **NoClobber** parameter is used in the cmdlet.

 Expand table

Type:	Microsoft.CertificateServices.Commands.CryptoAlgorithmOptions
Position:	Named
Default value:	None
Required:	False
Accept pipeline input:	False
Accept wildcard characters:	False

**-NoClobber**

Specifies that if the PFX file already exists, it should not be overwritten. This parameter takes precedence over the **Force** parameter, which permits this cmdlet to overwrite a PFX file even if it has the Read-only attribute set.

 Expand table

Type:	Microsoft.CertificateServices.Commands.CryptoAlgorithmOptions
Position:	Named

Default value:	None
Required:	False
Accept pipeline input:	False
Accept wildcard characters:	False

**-NoProperties**

Specifies whether the extended properties for a certificate are exported. If this parameter is specified, then extended properties are not included with the export. By default, all extended properties are included in the exported file.

 Expand table

Type:	Microsoft.CertificateServices.Commands.CryptoAlgorithmOptions
Position:	Named
Default value:	None
Required:	False
Accept pipeline input:	False
Accept wildcard characters:	False

**-Password**


Specifies the password used to protect the exported PFX file. The password should be in the form of secure string. Either the **ProtectTo** or this parameter must be specified, or an error will be displayed.

 Expand table

Type:	System.SecureString
Position:	Named
Default value:	None
Required:	False
Accept pipeline input:	False
Accept wildcard characters:	False

**-PFXData**

Specifies a **PFXData** object that contains one or more certificates from a PFX file.

 Expand table

Type:	PfxData
Position:	0
Default value:	None
Required:	True
Accept pipeline input:	True
Accept wildcard characters:	False

**-ProtectTo**


Specifies an array of strings for the username or group name that can access the private key of PFX file without any password. This requires a Windows Server 2012 or later domain controller. Either the **Password** or this parameter must be specified, or an error will be displayed.

 Expand table

Type:	String[]
Position:	Named
Default value:	None
Required:	False
Accept pipeline input:	False
Accept wildcard characters:	False

**-WhatIf**

Shows what would happen if the cmdlet runs. The cmdlet is not run.

 Expand table

Type:	Microsoft.CertificateServices.Commands.CryptoAlgorithmOptions
Aliases:	wi
Position:	Named
Default value:	False
Required:	False
Accept pipeline input:	False

Accept wildcard characters:	False
-----------------------------	-------

# Inputs

## X509Certificate2[]

The X509Certificate2[] object is an array of certificate objects.

# Outputs

## FileInfo

The FileInfo object contains the information about the PFX file.

# Related Links

- [ConvertTo-SecureString](#)
- [Get-ChildItem](#)
- [Get-PfxData](#)
- [Import-PfxCertificate](#)

# Feedback

Was this page helpful? 

Yes

No

[Provide product feedback](#)

© Microsoft 2024