



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 >



Learn

Discover ▾

Product documentation ▾

Development languages ▾

Topics ▾



Sign in

MSIX

Overview ▾

Admin & enterprise ▾

Developer ▾

Resources ▾

Filter by title

▾ MSIX

Overview

What is MSIX?

Features and supported platforms

MSIX appContainer apps

Flexible virtualization

▾ MSIX package signing

Overview

Sign an app package using SignTool

Sign an MSIX package with Device Guard signing

Known issues and troubleshooting

Unsigned MSIX for testing

> App package formats

> API and schema reference

MSIX resources

> Customer journeys

MSIX partners

> MSIX SDK

> Create an MSIX package from an existing installer

> Build an MSIX package from source code

> Manage your MSIX deployment

> MSIX validation, testing, and troubleshooting

[Learn](#) / [Windows](#) / [MSIX](#) /



## Create an unsigned MSIX package

Article • 05/26/2023 • [5 contributors](#)

[Feedback](#)

### In this article

[Create an unsigned package](#)

[Install an unsigned package](#)

[Related topics](#)

As of Windows 11, you can install your app via PowerShell without needing to sign your package. This feature is intended to make it easier for you to quickly test your app. Don't use this feature to distribute your app widely.

## Create an unsigned package

An unsigned package must include a special OID (organization ID) value in its [Identity](#) element in the manifest file, otherwise it won't be allowed to register. An unsigned package will never have the same identity as a package that's signed. That prevents unsigned packages from conflicting with, or spoofing the identity of, a signed package.

Here's an example.

XML

Copy

```
...
<Identity Name="NumberGuesserManifest"
  Publisher="CN=AppModelSamples, OID.2.25.31172936891398431765440773059495699772
  Version="1.0.0.0" />
...
```

## Install an unsigned package

### Important

In most scenarios, you'll need to run PowerShell as administrator. See the details below.

- To install an unsigned package, pass the `-AllowUnsigned` flag to the [Add-AppxPackage](#) command.

 Download PDF

- In most scenarios, the unsigned package will contain executable content; so you'll need to run PowerShell as administrator. That's because an unsigned package containing executable content must be installed for all users. Since that can affect more than just the current user, it requires administrator privilege.
- If the unsigned package contains only non-executable content (for example, when you need to load only images, assets and other content, or script files) then administrator privilege is *not* needed, and a non-admin can install the package.

Here's an example of the syntax.

PowerShell  Copy



```
Add-AppPackage -Path ".\MyEmployees.appx" -AllowUnsigned
```

When your app is ready to be distributed, you should ensure that the package is signed. Be sure to remove the special OID, and ensure that the publisher name is the same as the certificate subject name.

## Related topics

- [Create hosted apps](#)

## Feedback

Was this page helpful?  Yes  No

[Provide product feedback](#)  | [Get help at Microsoft Q&A](#)

## Additional resources

### Training

Module  
[Install and configure apps on a session host - Training](#)

MSIX app attach is a way to deliver MSIX applications to both physical and virtual machines. MSIX app attach is different from regular MSIX because it’s specifically for Azure Virtual Desktop.

Certification  
[Microsoft Certified: Power Platform Developer Associate - Certifications](#)

Demonstrate how to simplify, automate, and transform business tasks and processes using Microsoft Power Platform Developer.

### Events

Nov 20, 12 AM - Nov 22, 12 AM

Gain the competitive edge you need with powerful AI and Cloud solutions by attending Microsoft Ignite online.  
[Register now](#)