How to troubleshoot MSIX app attach

MSIX app attach is an application delivery mechanism that takes dependencies on:

- Windows Virtual Desktop (WVD)
- Storage fabric for hosting MSIX images
- The MSIX packaging format. More [info](#).
- Tools ([MSIX packaging tool](#) and [MSIXMGR](#))

This article does not cover repackaging [non-MSIX to MSIX](#) and expanding an [MSIX package](#).

This article covers steps for troubleshooting all dependencies that MSIX app attach relies on. Areas covered including the certificate used to sign the MSIX applications, setting up MSIX share, permissions issue, error messages in WVD management UI for MSIX app attach, registration, staging, user experience error, tooling, and other.
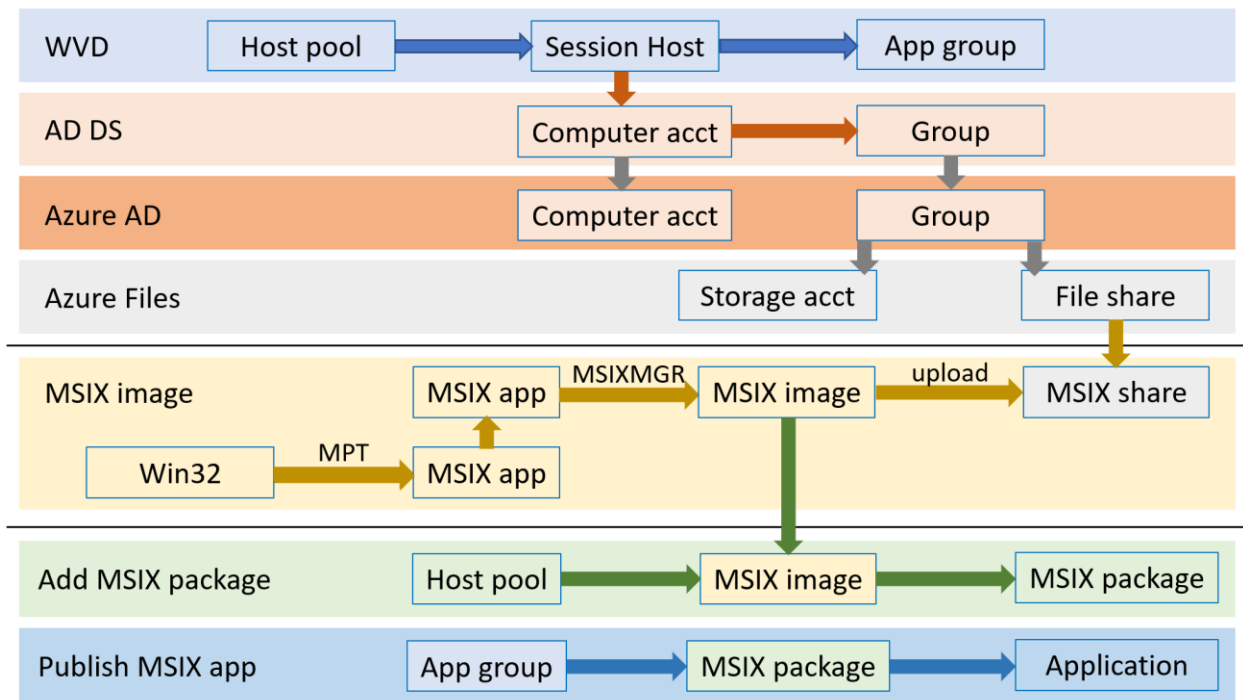
Overview of MSIX app attach setup process

The process of using MSIX app attach to deliver applications to users is:

1. WVD host pool with healthy VMs
2. Obtain application packaged as MSIX (aka as MSIX package)
3. Expand MSIX package in MSIX image via the [MSIXMGR](#)
4. (Optional) if MSIX package was signed with a no-public certificate install it in **Trusted People** on each VM in the host pool
5. Create new or use existing AD DS group in AD DS
6. Add all session hosts computer accounts to the AD DS group
7. VMs must be restarted to pick up membership to the AD DS group
8. Synch the group to Azure AD (this may take up to 30 minutes)
9. Create MSIX share
10. Grant permissions on the MSIX share
    a. If using Azure Files for the MSIX share
    b. Grant the AD DS group RBAC permission on the storage account
    c. Grant the AD DS group NTFS permissions on the file share
11. Upload MSIX image to MSIX share.
12. Open the WVD host pool and click **MSIX packages**
13. Click **Add** and complete **Add MSIX package**
14. Next open remote app (RAG) or desktop application (DAG) and publish an MSIX application
15. Assign users or user groups to the RAG or DAG

The below diagram shows overview of the entire MSIX app attach process including setup of the infrastructure.

The first four areas WVD, AD DS, Azure AD, and Azure Files are considered underlying infrastructure and prerequisites for MSIX app attach. They are not in scope for this document.

WVD	Host pool → Session Host → App group
AD DS	Computer acct → Group
Azure AD	Computer acct Group
Azure Files	Storage acct File share
MSIX image	MSIX app —MSIXMGR→ MSIX image —upload→ MSIX share; Win32 —MPT→ MSIX app
Add MSIX package	Host pool → MSIX image → MSIX package
Publish MSIX app	App group → MSIX package → Application

Troubleshooting MSIX app attach dependencies

WVD host pool VM is unhealthy

Before being able to add MSIX image to a WVD host pool the VMs in the host pool must be on and available. Generic troubleshooting for unhealthy session host is covered here.

If a machine goes unhealthy after an MSIX package has been added to the host pool please go here.

Troubleshooting MSIX package

Troubleshooting MSIX packages and MSIX applications is not in scope for MSIX app attach and this document. Please reference this article for troubleshooting steps.

All MSIX package error are listed here.

Errors when using MSIX packages

Error/Behavior	Explanation/Fix
0x80073D02 0x80073D02: Unable to install because the following apps need to be closed **<PackageName>**	Trying to register an application that is already running.
error 0x80070013: Failed to set access rights to resources.pri.	Malformed MSIX package. This usually happens when non-MSIX is repackaged to MSIX. Repackage again and confirm MSIX package install prior to expanding into MSIX image and delivering via MSIX app attach.

0x80070002 AppxManifest.xml(20,16): error 0x80070002: Cannot register the **<PackageName>** package because the following error was encountered while registering the windows.defaultPrograms extension: The system cannot find the file specified. . Try again and contact the package publisher if the problem persists.	Malformed MSIX package. This usually happens when non-MSIX is repackaged to MSIX. Repackage again and confirm MSIX package install before expanding into MSIX image and delivering via MSIX app attach.
0x80070002 AppxManifest.xml(17,4): error 0x80070002: Cannot register the **<PackageName>** package because the following error was encountered while registering the windows.defaultPrograms extension: The system cannot find the file specified. . Try again and contact the package publisher if the problem persists.	The RD Agent check for new on dynamically added MSIX packages every 5 minutes (by default). This error will occur when the user logs in and RD Agent hasn't staged the MSIX package
0x80073CF3 AppX Deployment operation failed for package **<PackageName>** with error 0x80073CF3. The specific error text for this failure is: Windows cannot install package **<PackageName>** because this package depends on a framework that could not be found. Provide the framework **<FrameworkName>** published by "CN=Microsoft Corporation, O=Microsoft Corporation, L=Redmond, S=Washington, C=US", with neutral or x64 processor architecture and minimum version **<Version>**, along	The package has failed a registration because and underlying dependency is missing.

with this package to install. The frameworks with name **<FrameworkName>** currently installed are: {}	
0x80073CF3 Deployment Register operation with target volume C: on Package **<PackageName>** from: (AppxManifest.xml) failed with error 0x80073CF3	The package has failed a registration because an underlying dependency is missing.

Certificate related errors

MSIX package must be signed. The same applies to a win32/non-MSIX application that has been repackaged as am MSIX package. This requires certificates to be available during all adding MSIX package to host pool, staging and registering. While public certificates are recommended it is possible to use non-public or self-signed certificates.

Non-public and self-signed certificates must be:

- available on all VMs in a host pool
- Installed under Local Machine > Trusted People

The error that the UI reports where a certificate is missing are:

```
ActivityId: <ActivityId> Error: The MSIX Application metadata expand request
failed on all Session Hosts that it was sent to. Session Host: <hostname>,
Error: App contains untrusted signature.Session Host: <hostname>, Error: App
contains untrusted signature. (Code: 400)
```

The solution for this issue is to installing the certificate for this application via steps here.

The Azure portal UI where MSIX app attach packages are managed is under the **Windows Virtual Desktop** module.

AD DS group and sync to Azure AD

The process for granting access to Azure Files requires an AD DS group to have the computer accounts for the session host. Then that group will be synced to Azure.

When new members of that group are added it takes up to 30 minutes to synch. If needed sync can be forced via the PS code here.

When computer objects are added to AD DS group the machines those objects represent do not pick up membership in the new group until those machines have been restarted.

Troubleshooting MSIX file share

MSIX share is the term representing the network share where MSIX images are stored. For deployments with up to 2000 VMs Azure Files standard tier will provide enough performance.

Step by step guide on how to set up Azure Files Storage Account for MSIX app attach is available [here](#).

Additionally, the Azure Files team have excellent troubleshooting document available [here](#).

Verify storage account location
RD Agent must be able to access and mount an MSIX image from MSIX share in less than 400ms for this to happen the storage account must be in the same region as the session host.

Azure premium files do not have file share
If using Azure Files premium storage account must be created with **Account Kind** set to **FileStorage.**

Account kind ⓘ	FileStorage ⌄

ⓘ This account kind allows you to create Azure file shares with premium performance characteristics that can be tailored to your needs.
Learn more about file storage ↗

Confirm there is a file share created under the storage account.

Storage account permissions
Confirm via **Access control (IAM)** that the AD DS group has **Storage File Data SMB Share Contributor** permissions on the storage account.

File share aka MSIX share permissions
MSIX app attach uses MSIX images stored on a file share. Certain requirements must be met:

- all VMs computer/system accounts in the host pool must have access to the share
- share should be located as close as possible to the session host (same region, same resource group)

Note: While a regular user may have access and be able to mount the MSIX images from the store path that does not guarantee that the VMs computer/system accounts have permissions.

Confirm that the AD DS group has **Read & Execute** permissions on the file share. To do this use the steps here to mount the [file](#) share and grant or review permissions.

Steps to confirm storage account permissions
1. Navigate to the Azure portal
2. Open the storage account being used for storing the MSIX packages
3. Click on **Access Control (IAM)**
4. Click on **Role assignment**
5. Confirm that the group containing the machine accounts have **Storage File Data SMB Share Contributor**

Troubleshooting MSIX app attach

MSIX image

Creating an MSIX image
MSIX image is a VHD(x) or CIM file in which MSIX has been expanded (unzipped and ACLs applied). The expansion is done via the [MSIXMGR tool](#).

The steps to use the MSIXMGR are defined [here](#).

MSIXMGR is a command-line tool that relies heavily upon the correctness of its input however, MSIXMGR does not very that input. It is the responsibility of the IT professional to confirm validity.

Outside of bad input, the next most common error is running MSIXMGR without the needed permissions.

MSIXMGR errors

Error	Explanation/Fix
0x8bad0003	Target VHD(x) or CIM file is too small to fit the expanded MSIX package. Increase the size of the VHD(x) or CIM file.
0x8007007b	Bad input parameter. The destination folder does not exist.
0x8bad0001	Bad input parameter. Source MSIX package cannot be found.
0x8bad0071	MSIXMGR must be run in an elevated mode as it must have the permissions to apply ACLs.
0x800B010A	The certificate used to repackage/compile the application is not present on the session host or its not in Trusted People
0x80070522	MSIXMGR must be ran in elevated mode as it must have the permissions to apply ACLs
0x80070050	Creating the VHD(X) file c:\temp**<PackageName>**.vhdx failed with HRESULT 0x80070050 The destination file already exists.
0x80070003	MSIXMGR cannot find the specified path. MSIXMGR does not create folders to host the CIM.
0x80070057	[WARNING] The following packages from C:\temp**<PackageName>**.vhdx failed to get unpacked. Please try again: Failed with HRESULT 0x80070057 when trying to unpack C:\temp**<PackageName>**.vhdx Please confirm the given package path is an .appx, .appxbundle, .msix, or .msixbundle file This is by design MSIXMGR can only use .appx, .appxbundle, .msix, or .msixbundle as input.
0x80070057	Creating the CIM file **<PackagePath.cim>** failed with HRESULT 0x80070057 This error happens when the CIM file size is too small to host the payload.

Upload MSIX image
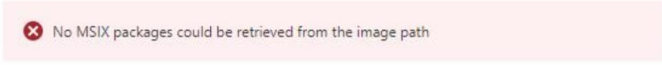
Uploading MSIX image to the MSIX share can be done via:

- Azure Portal
- PowerShell
- Azure CLI
- Storage Explorer
- 3rd party tools

Troubleshooting issue with uploading to the share is mainly confirming that user has permissions to upload content or setup.

UI for managing MSIX app attach

The Azure portal UI for managing MSIX app attach relies on the VMs in the session host to:

- validates integrity and structure of the MSIX package
- resolves that path to the MSIX image
- validate permission to the MSIX network share
- confirms a valid certificate is present
- checks for MSIX application dependencies

Error	Explanation/Fix
ActivityId: 00000000-0000-0000-0000-000000000000 Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: **\<HostName\>**, Error: Object reference not set to an instance of an object	MSIX app attach is not enabled on the subscription. ❌ No MSIX packages could be retrieved from the image path Adding MSIX package notifies WVD that a new application has been made available for publishing and assigning. MSIX image path * ⓘ　\\WVD-AD-001\AppAttachPackages\GoogleChrome_... ✓ MSIX image path error ERROR TYPE　🗐 ActivityId: 68593262-5e31-46b2-b186-ab1b6c82e3ec Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: WVDApp, Error: Object reference not set to an instance of an object.. (Code: 400)
ActivityId: 00000000-0000-0000-0000-000000000000 Error: This functionality is not supported. It will be included in a feature release.	MSIX app attach has been enabled but the session host is not in the validation environment. (public preview) This is applicable for the public preview. ❗ Add MSIX package　7:22 PM ✕ ActivityId: 84120720-6431-462e-8aab-653210be7b93 Error: This functionality is not supported. It will be included in a future release. Go to: https://go.microsoft.com/fwlink/?linkid=2146741
ActivityId 00000000-0000-0000-0000-000000000000 Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: **\<HostName\>**, Error: Object reference not set to an instance of an object..Session Host: **\<HostName\>**, Error:	This error occurs when the RD Agent version does not support MSIX app attach. VMs running guest OS pre-Win 10 2004, Win 7, or WS. (public preview) During the public preview, this will happen if the host pool is not in a validation environment.

Object reference not set to an instance of an object.. (Code: 400)	❌ No MSIX packages could be retrieved from the image path Adding MSIX package notifies WVD that a new application has been made available for publishing and assigning. MSIX image path * ⓘ `\\stgeorg1215-0\temp\GoogleChrome_68.46.66.0_x6...` ✓ **MSIX image path error** ERROR TYPE 📋 ActivityId: cc943539-2ab4-422a-b6ee-81cce241602a Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: stgeorg1215-0, Error: Object reference not set to an instance of an object..Session Host: stgeorg1215-1, Error: Object reference not set to an instance of an object.. (Code: 400)
The host pool **\<HostName\>** does not contain available Virtual Machines.	There are no session hosts or they are all unavailable. ❌ The host pool stgeorg1028-hp does not contain available Virtual Machines. MSIX packages are conceptually similar to installation media for Win32 applications. Learn more ⧉
The value must not be empty.	Empty path was specified in the **MSIX image path** field. Please provide a valid file path. MSIX image path * ⓘ [ ] ❌ The value must not be empty.
Enter a Windows rooted absolute path or a network shared path. Files names can't contain any of the following characters : * ? " < > | / and only VHD, VHDx, and CIM extensions are supported.	The path specified does not point to a VHD(x) or CIM file. MSIX image path * ⓘ `\\s` ❌ Enter a Windows rooted absolute path or a network shared path. Files names can't contain any of the following characters : * ? " < > | / and only VHD, VHDx and CIM extensions are supported
No MSIX packages could be retrieved from the image path.	The path cannot be resolved by the VMs machine account. ❌ No MSIX packages could be retrieved from the image path Adding MSIX package notifies WVD that a new application has been made available for publishing and assigning. MSIX image path * ⓘ `C:\temp\MicrosoftEdgeDev_87.0.658.0_x64__81q6ced...` ✓ **MSIX image path error** ERROR TYPE 📋 ActivityId: 0b793b82-05a8-4309-919d-62498e95bbc5 Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: stgeorg1026-1, Error: Virtual disk not found at ≤C:\temp\MicrosoftEdgeDev_87.0.658.0_x64__81q6ced8g4aa.vhdx≥.Session Host: stgeorg1026-0, Error: Virtual disk not found at ≤C:\temp\MicrosoftEdgeDev_87.0.658.0_x64__81q6ced8g4aa.vhdx≥. (Code: 400)

	Raw error: {"code":"400","message":"ActivityId: 0b793b82-05a8-4309-919d-62498e95bbc5 Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: stgeorg1026-1, Error: Virtual disk not found at ≤C:\\temp\\MicrosoftEdgeDev_87.0.658.0_x64__81q6ced8g4aa.vhdx≥.Session Host: stgeorg1026-0, Error: Virtual disk not found at ≤C:\\temp\\MicrosoftEdgeDev_87.0.658.0_x64__81q6ced8g4aa.vhdx≥."}
No MSIX packages could be retrieved from the image path.	The VMs machine/system accounts do not have permission to access the path. ❌ No MSIX packages could be retrieved from the image path Adding MSIX package notifies WVD that a new application has been made available for publishing and assigning. MSIX image path * ⓘ \\stgeorgi1027.file.core.windows.net\test\edge_dev_8... ✓ **MSIX image path error** ERROR TYPE ActivityId: 11b2c8ef-cd26-4638-b287-65b4e2cc1a05 Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: stgeorg1026-1, Error: The user name or password is incorrect. .Session Host: stgeorg1026-0, Error: The user name or password is incorrect. . (Code: 400) Raw error: ActivityId: 11b2c8ef-cd26-4638-b287-65b4e2cc1a05 Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: stgeorg1026-1, Error: The user name or password is incorrect. .Session Host: stgeorg1026-0, Error: The user name or password is incorrect. . (Code: 400)
Error subscription is not registered for MSIX app attach.	Prior to being publicly available this MSIX app attach in closed private preview. Please reach out to us via TBD. ❗ Add MSIX package 4:41 PM ✕ ActivityId: fb976335-7794-461b-9700-f50d00e33eb8 Error Subscription is not registered for use of MSIX
Error subscription is not registered for MSIX app attach or the host pool is not in evaluation environment.	Please follow process here aka.ms/msixandwvd to get the feature enabled. This is applicable only during public preview. ❗ Add MSIX package 9:48 PM ✕ ActivityId: ff5053c0-6a64-4769-8261-ea02beddf95c Error: This functionality is not supported. It will be included in a future release. Go to: https://go.microsoft.com/fwlink/?linkid=2146741

MSIX package already exist in host pool.	The same version of the package is already added to the host pool. Only one version of the MSIX package is allowed per host pool. **Add MSIX package** ✕ Adding MSIX package notifies WVD that a new application has been made available for publishing and assigning. MSIX image path ⓘ \\stgeorgi-0\msixshare\packages\Minecraft appxbundl... MSIX package * ⓘ Microsoft.MinecraftEducationEdition_1.12.601.0_x6... ⌄ ❌ MSIX package already exists in host pool Package applications ⓘ Microsoft.MinecraftEducationEdition Display name ⓘ Same package different path Version ⓘ 1.12.601.0 Registration type ⓘ ⦿ On-demand ⭘ Log on blocking State ⓘ ⭘ Inactive ⦿ Active
The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: **\<HostName\>**, Error: App contains untrusted signature."	**Add MSIX package** ✕ ❌ No MSIX packages could be retrieved from the image path Adding MSIX package notifies WVD that a new application has been made available for publishing and assigning. MSIX image path * ⓘ \\msixfileshare.file.core.windows.net\msixfileshare\vsc... ✓ **MSIX image path error** ERROR TYPE 📋 ActivityId: b3c1028f-9cd7-441e-b8e1-3cadba8fcb19 Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: stgeorgi115-2, Error: App contains untrusted signature. (Code: 400)
ActivityId: **\<ActivityId\>** Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host:**\<HostName\>**, Error: Msix structure error: No parent folder found for MSIX package at path ≤**\<MsixImagePath\>**≥.Session Host: **\<HostName\>** (Code: 400)	MSIX image was created using incorrect structure. Expand the MSIX package via the MSIXMGR. **Add MSIX package** ✕ ❌ No MSIX packages could be retrieved from the image path Adding MSIX package notifies WVD that a new application has been made available for publishing and assigning. MSIX image path * ⓘ \\msixfileshare.file.core.windows.net\msixfileshare\ci... ✓ **MSIX image path error** ERROR TYPE 📋 ActivityId: 05f0826f-a2ad-405a-aecf-15987022bfc2 Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: stgeorgi115-2, Error: Msix structure error: No parent folder found for MSIX package at path ≤\\msixfileshare.file.core.windows.net\msixfileshare\cim\wvdmigration.cim≥.Session Host: stgeorgi115-1, Error: App contains untrusted signature. (Code: 400)

UI times out while trying to expand a package.	When in the Azure portal UI an SMIX image path is specified. That path must be resovled and the MSIX imaget that it is pointing to must be mounted. If this operation takes over 400 ms the process will time out. This is most often due to the session host and the MSIX files share being in different regions with large latency between them.
ActivityId: **<ActivityId>** Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: **<HostName>**, Error: Native error 32..Session (Code: 400)	Option 1: Session host VMs are being restarted, or the RD Agent is in the process of updating, or the RD Agent Boot Loader is not running. Option 2: VHD(x) file had a exclusive lock on it.
ActivityId: **<ActivityId>** Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: **<HostName>**, Error: Could not find a part of the path '\\?\Volume{cf629aeb-0000-0000-0000-010000000000}'.. (Code: 400)	During expanding the RD Agent was not able to finish the expansion. The underlying VHD/VHD(x) or CIM reported not ready. Confirm that the storage account is in 400ms from session. Confirm that session host is not experiencing high memory and CPU utilization.
Error "No MISX package could be retrieved from image path" but path and permissions to MSIX share are correct.	The user does not have permissions to add MSIX packages.

ActivityId: **<ActivityId>** Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: **<HostName>**, Error: The operation was canceled.. (Code: 400)	This request has tied out due to the RD Agent on the session host not being able to complete the operation in the allowed time. This happens when the VM is experiencing high usage and the MSIX share is too far (latency based) from the session host. Alternatively, there could be a network appliance (firewall) or setting (NSG) blocking the access to the file share. **Add MSIX package** ✕ ❌ No MSIX packages could be retrieved from the image path Adding MSIX package notifies WVD that a new application has been made available for publishing and assigning. MSIX image path * ⓘ \\msixfileshare.file.core.windows.net\msixfileshare\Mi... ✓ **MSIX image path error** ERROR TYPE ActivityId: 6660c937-4052-4785-b0c6-ce542c7939d8 Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: stgeorgi115-2, Error: The operation was canceled.. (Code: 400)
Add MSIX package dialog does not load and continues to "spin".	This happens when a MSIX image is build from an MSIX bundle. **Add MSIX package** • • • • • ✕ Adding MSIX package notifies WVD that a new application has been made available for publishing and assigning. MSIX image path * ⓘ \\msixfileshare.file.core.windows.net\msixfileshare\Mi... ✓
ActivityId: **<ActivityId>** Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: **<HostName>**, Error: Virtual disk data corruption at ≤**<MsixImagePath>**≥	The VHD(x) or CIM cannot be parsed for valid content. This can be due to the MSIX image being empty, corrupted, bad formatting, or other file system issues.

	Add MSIX package ✕ ❌ No MSIX packages could be retrieved from the image path Adding MSIX package notifies WVD that a new application has been made available for publishing and assigning. MSIX image path * ⓘ `\\msixfileshare.file.core.windows.net\msixfileshare\em...` ✓ **MSIX image path error** ERROR TYPE ⧉ ActivityId: d7605d6c-11f9-4cea-8448-bd122a04f194 Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: stgeorgi115-1, Error: Virtual disk data corruption at ≤\\msixfileshare.file.core.windows.net\msixfileshare\emptyapp.vhdx≥.Session Host:
ActivityId: **<ActivityId>** Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: **<HostName>**, Error: Connection not found: RD Agent does not have an active connection to the broker.	RD Broker and RD Agent are not able to exchange information due to networking issues. **Add MSIX package** ✕ ❌ No MSIX packages could be retrieved from the image path Adding MSIX package notifies WVD that a new application has been made available for publishing and assigning. MSIX image path * ⓘ `\\advm\AppAttach\AppAttachShare20Apps\chrome\...` ✓ **MSIX image path error** ERROR TYPE ⧉ ActivityId: 3404118b-bbee-4676-9ee1-1b519869e75c Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: stgeorg1020-0, Error: Connection not found: RD Agent does not have an active connection to the broker.. (Code: 400)
ActivityId: **<ActivityId>** Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host **<HostName1>**, Error: Virtual disk not found at ≤**<HostName1>**, **<PackagePath>**≥.Session Host: **<HostName2>**, Error:	One of the session host in the host pool is not able to access the path where the MSIX image is stored.

The operation was canceled.. (Code: 400)	**Add MSIX package** ✕ ❌ No MSIX packages could be retrieved from the image path Adding MSIX package notifies WVD that a new application has been made available for publishing and assigning. MSIX image path * ⓘ `C:\temp\GoogleChrome_68.46.66.0_x64__74vyvr5aw9...` ✓ **MSIX image path error** ERROR TYPE 📋 ActivityId: cfe0b64b-ebbe-4133-880c-8fbd125ede79 Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: stgeorg1208-1, Error: Virtual disk not found at ≤C:\temp\GoogleChrome_68.46.66.0_x64__74vyvr5aw93s6.vhdx≥.Session Host: stgeorg1208-0, Error: The operation was canceled.. (Code: 400)
ActivityId: **<ActivityId>**Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: **<HostName1>**, Error: App contains untrusted signature.Session Host: **<HostName2>**, Error: A required certificate is not within its validity period when verifying against the current system clock or the timestamp in the signed file. (Exception from HRESULT: 0x800B0101). (Code: 400)	This error will happen when the certificate used to package the MSIX applications has expired. **Add MSIX package** ✕ ❌ No MSIX packages could be retrieved from the image path Adding MSIX package notifies WVD that a new application has been made available for publishing and assigning. MSIX image path * ⓘ `\\msixfileshare.file.core.windows.net\msixfileshare\Te...` ✓ **MSIX image path error** ERROR TYPE 📋 ActivityId: a16ab939-36ff-49d6-b0ed-a9f34d0a1301 Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: stgeorg1026-0, Error: App contains untrusted signature.Session Host: stgeorg1026-1, Error: A required certificate is not within its validity period when verifying against the current system clock or the timestamp in the signed file. (Exception from HRESULT: 0x800B0101). (Code: 400)
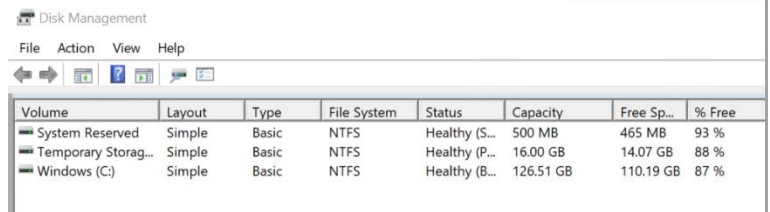
Adding MSIX package to host pool

In the Azure portal, these are the steps to deliver packages to users or user groups

1) In Azure portal, open **WVD UI**
2) Navigate to a host pool
3) Click the **MSIX packages** tab
4) Upload a package by giving the full path to the image that contains the package

5) Open a DAR or RAG and publish an MSIX application from inside the package
6) Assign users to the DAG or RAG

Troubleshooting session host

Error	Explanation/Fix
When adding a session host to an existing host pool it goes unhealthy right after provisioning has been completed.	Confirm that the new session has permissions to the MSIX share. The new session host automatically inherits MSIX packages from the host pool. Name ↑↓ · Status ↑↓ · Drain mode ↑↓ · Assigned User ↑↓ jushiah-2.jushiahwvdt··· · ❗Unavailable · Off · - · 0
The package was added to the host pool but the VHD(x) does not show in Disk Management.	The RD agent checks for new packages every 5 minutes. If an MSIX image was added to a host it may take 5 minutes for the VHD/VHDx to appear in the Disk Management or mountvol output. Note: CIMFS does not appear in Disk Management.
The session host reports that is unable to stage **"SessionHost unhealthy: MSIX packages have failed to properly stage"** but **Disk Management** already shows the VHD/VHDx has been mounted (staged)	This happens on agents that are older than *1262. Manually unmount the VHD/VHDx and restart session host. This is due to RD Agent crashing.

PowerShell errors

Error	Explanation/Fix
Expand-AzWvdMsixImage_ExpandExpanded: ActivityId: **<ActivityId>** Error: The MSIX Application metadata expand request failed on all Session Hosts that it was sent to. Session Host: stgeorgi-0, Error: Unable to mount virtual disk at ≤**<MsixImagePath>**≥	The RD agent was not able to mount the MSIX image using the VMs machine/system account.
Expand-AzWvdMsixImage - HostPoolName $hp -	The MSIX application inside the MSIX images is not correctly packaged.

ResourceGroupName $rg -SubscriptionId $subID -Uri '**<MsixImagePath>**' Expand-AzWvdMsixImage_ExpandExpanded: ActivityId: **<ActivityId>** Error: Could not get MSIX applications for Host Pool '**<HostName>**' from image '**<MsixImagePath>**'. Check inner exception for details.	

New-AzWvdMsixPackage -HostPoolName $hp -ResourceGroupName $rg -SubscriptionId $subId -PackageAlias $obj.PackageAlias -DisplayName PowerBI -ImagePath '**<MsixImagePath>**' -IsActive:$true
 	
Expand-AzWvdMsixImage_ExpandExpanded: C:\temp\msix test\azure-powershell\src\DesktopVirtualizatio n\custom\New-AzWvdMsixpackage_PackageAlias.ps 1:134
Line |
 134 | $extractPackage = Az.DesktopVirtualization\Expand-AzWvdMsixIm …
     |
~~
~~
~
   | ActivityId: **<ActivityId>** Error: Could not get MSIX applications for Host Pool '**<HostName>** ' from image '**<MsixImagePath>**'. Check inner exception for details.
 	
New-AzWvdMsixPackage_PackageAlias: Cannot bind argument to parameter 'FullName' because it is an empty string.
	The value passed for the application full name is not matching the application full name in the package.

Get-AzWvdMsixPackage - HostPoolName $hp - ResourceGroupName $rg - SubscriptionId $subId -FullName Power Get-AzWvdMsixPackage_Get: ActivityId: **<ActivityId>** Error: MSIX Package does not exist	No package exists in the pool that matches the full name specified. Make sure a package has been added to the host pool.

OS and service-related errors
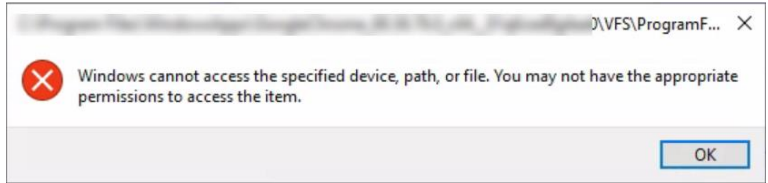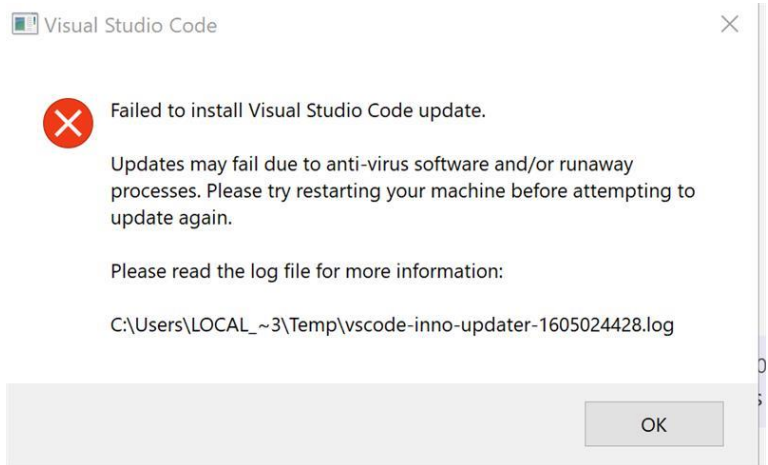
Error/Behavior	Explanation/Fix
AppAttachServiceImpl - AppAttachDestageAsync: Dismounting virtual disk for package '\\?\Volume{cf629aeb-0000-0000-0000-010000000000}' failed with exception: System.InvalidOperationException: Native error 21. at Microsoft.RDInfra.VirtualDiskAPI.VirtualDisk.Detach() in E:\agent_work\2\s\src\RDAgent\src\Service\VirtualDisk\VirtualDisk.cs:line 633 at Microsoft.RDInfra.RDAgent.AppAttach.VhdVolume.Dispose() in E:\agent_work\2\s\src\RDAgent\src\Service\VirtualDisk\VhdVolume.cs:line 127 at Microsoft.RDInfra.RDAgent.AppAttach.VirtualDiskManager.RefDispose(String imagePath, String canonicalPath) in E:\agent_work\2\s\src\RDAgent\src\Service\VirtualDisk\VirtualDiskManager.cs:line 205 at Microsoft.RDInfra.RDAgent.AppAttach.VirtualDiskManager.VirtualDiskHandle.Dispose() in E:\agent_work\2\s\src\RDAgent\src\Service\VirtualDisk\VirtualDiskManager.cs:line 48 at Microsoft.RDInfra.AppAttach.AppAttachServiceImpl.<>c__DisplayClass3	Native error 21 indicates that the media containing the MSIX image (VHD(x) or CIM) was not ready to perform read operations. This happens due to: • High network latency • Session host experiencing high memory and/or CPU consumption

Error/Behavior	Explanation/Fix
1_1.<<DestagePackagesAsync>b__0>d.MoveNext() in E:\agent_work\2\s\src\RDAgent\src\Service\AppAttach\AppAttachServiceImpl.cs:line 674	

User experience

Error/Behavior	Explanation/Fix
The application does show in the user remote desktop but fails to start.	Option 1: Verify that all application dependencies have been delivered in an image by reviewing dependencies in the Azure portal or exploring Option 2: Confirm the package is set to Active in the MSIX management UI

The application does not show in the user remote desktop.	1. Verify that application has been assigned to the DAG
	2. Confirm application dependencies are present
	3. Run query to see Registration errors
	``
	let GetDataFrom = (clusterName:string)
	{
	cluster(clusterName).database('WVD').
	RDOperation
	| where HostPool == "<Hostpool>"
	| where RDTenant == "<Tenant>"
	| where Name == "RegisterMsixPackages"
	| where TIMESTAMP >= datetime(2020-10-26T21:42:26.1265554Z)
	};
	 union GetDataFrom('<Cluster>')
	``
	4. If self-signed certificate is being used confirm certificate is installed in the Trusted people folder

The application shows in feed but when session loads user sees a File explorer screen.	1. Verify that application has been assigned to the DAG
	2. Confirm application dependencies are present
	3. Run query to see Registration errors
	``
	let GetDataFrom = (clusterName:string)
	{
	cluster(clusterName).database('WVD').
	RDOperation
	| where HostPool == "<Hostpool>"
	| where RDTenant == "<Tenant>"
	| where Name == "RegisterMsixPackages"
	| where TIMESTAMP >= datetime(2020-10-26T21:42:26.1265554Z)
	};
	 union GetDataFrom('<Cluster>')
	``

	4. If self-signed certificate is being used confirm certificate is installed in the Trusted people folder
Applications has been assigned to a user and after logoff and log in back in the application is gone, however registration is reported as successful in telemetry.	Confirm package is not staged for any users using the below PS command `Get-AppxPackage *vs* -AllUsers` When no packages are listed this is due to the last user on this session host VM having admin permissions. When that users logs off or is being log off due to inactivity this triggers a deregistration of MSIX packages that have been fully registered (this will not impact on-demand registered MSIX packages). Due to current design this deregistration will also fire destaging. The outcome of this is that subsequent users will not get the applications to display in the Start Menu or start for them. There will be no errors reported. Option 1 (recommended): The **$path** variable is the path to the volume which contains the MSIX image and can be obtained via **mountvol** or **Disk Management.** Replace **\\?** with **file://** Append **/apps/\<packageName\>/AppxManifest.xml** to the file path**.**

```
[Windows.Management.Deployment.PackageManag
er,Windows.Management.Deployment,ContentTyp
e=WindowsRuntime] | Out-Null
Add-Type -AssemblyName
System.Runtime.WindowsRuntime
$asTask =
([System.WindowsRuntimeSystemExtensions].Ge
tMethods() | Where { $_.ToString() -eq
'System.Threading.Tasks.Task`1[TResult]
AsTask[TResult,TProgress](Windows.Foundatio
n.IAsyncOperationWithProgress`2[TResult,TPr
ogress])'})[0]
$asTaskAsyncOperation =
$asTask.MakeGenericMethod([Windows.Manageme
nt.Deployment.DeploymentResult],
[Windows.Management.Deployment.DeploymentPr
ogress])
$packageManager =
[Windows.Management.Deployment.PackageManag
er]::new()
$path =
"file://Volume{volumeguid}/apps/packageFull
Name"
```

	```\n$path = ([System.Uri]$path).AbsoluteUri\n$asyncOperation =\n$packageManager.StagePackageAsync($path,\n$null, "StageInPlace")\n$task = $asTaskAsyncOperation.Invoke($null,\n@($asyncOperation))\n $task\n``` Option 2: Inactivate the MSIX image and remove it from the host pool. Wait 5+ minutes and add the package.
MSIX package has been added to the host pool but does not show in Disk Management.	RD Agent checks with broker every 5 minutes to see if new packages have been added to the host pool. There are few ways to force the RD agent to pick up changes right away. Option 1: Set the RD agent check-in interval to 1 minute via registry ([sample](#)) ```\n[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\RDInfraA\ngent\MSIXAppAttach]\n"PackageListCheckIntervalMinutes"=dword:0000000\n1\n``` Option 2: Restart the RDAgentBootLoader service Option 3: Restart the VM
User gets an error message when click on an icon for MSIX app attached application.	This is due to the MSIX application not being staged and register for this user.)\VFS\ProgramF... ✕ ❌ Windows cannot access the specified device, path, or file. You may not have the appropriate permissions to access the item. OK
Application repots failed to install an update. This message may be different based on application implementation. By design auto update for MSIX app attach delivered applications must be disabled.	Visual Studio Code ✕ ❌ Failed to install Visual Studio Code update. Updates may fail due to anti-virus software and/or runaway processes. Please try restarting your machine before attempting to update again. Please read the log file for more information: C:\Users\LOCAL_~3\Temp\vscode-inno-updater-1605024428.log OK

Two icons for same application appear in the start menu. This happens when the MSIX app attach application has attempted to auto update itself. All MSIX app attach delivered applications must be update via MSIX app attach. Auto update is not supported.	
MSIX app attached application reports that a file or a process is being used by another users	This application has been pacakged in a way that allows only on instance of the application to be run at a time. This application must be repacakge and the file in question included in the MSIX package.
Add-AppxPackage : Deployment failed with HRESULT: 0x80073CF6, Package could not be registered. error 0x80073D27: While processing the request, the system failed to register the windows.service extension due to the following error: The package installation failed because a version of the service exists outside of APPX packaging. Please contact your software vendor.	This error happens when the MSIX package is trying to start a service, however that service is already running on the machine. This can happen when a Win32 and MSIX app attach versions of the same app are delviered on same guest OS.
This error is due to the inability of the app to start a COM component. Unable to start a DCOM Server: **<PackageName>** as Unavailable/Unavailable. The error: "2147958023" Happened while starting this command:	
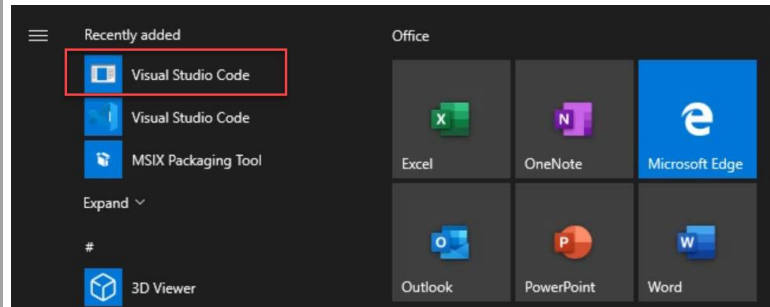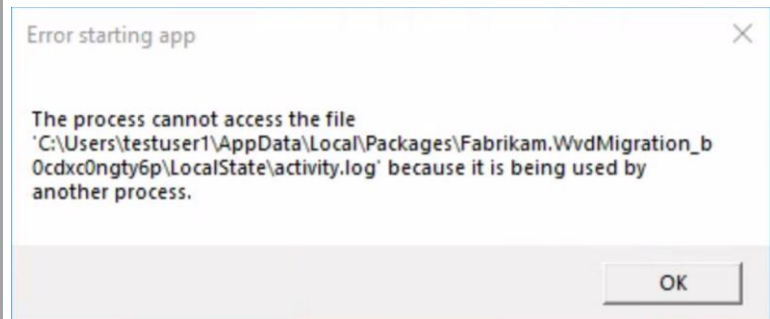
"**<PackageName>**" - ServerName:App.AppXfda2ecmmv7v peq0mkwrj7mkgkg8efffm.mca	
MSIX application has been removed from a remote app or desktop group but user can still see it in their session.	Changes to MSIX applcation assignments do not take effect in active user sessions. Users must log off and log back in for chagnes to take effect.
Application shows a white bar under its icon but never starts. This issue is due to the application not being able to start its containers. Errors related to this are under AppxDeployment-Server	

Admin experience

Error/Behavior	Explanation/Fix
MSIX package was added to a host pool but cannot be seen in Disk Manager or/and mountvol.	For the preview of MSIX app attach in WVD the session host check with the RD Broker every 5 minutes if new package has been added to the host pool. Hence between package being added to a host pool and it being staged there is a 5 minutes interval. Any users that login prior to staging being complete will not get the MSIX applications they have been granted access to (this will generate registration errors in the telemetry). Please note that MSIX images delivered via CIM files do not display in **Disk Manager** and only in **mountvol**.
Application has been removed from a RAG or a DAG, but user can still see it.	Removing an application will only take effect for a user after they have fully disconnected for the host pool.
Application has been removed from a RAG or a DAG and this generates a destaging error.	When user logs on, the RD Agent queries the RD Broker for list of applications that must registered. Same happens when user logs off with the distinction that this time apps will be deregistered and destaged. If an application has been removed (or added) to the RAG or DAG while the user had an active session that will result in deregistration and destaging error.

	It is recommended not to do modify access to applications while users have active sessions.
0x80070057	This error occurs when the MSIX application being installed was not able to obtain elevated permission to install one or more of its components.
0x80070032	This error occurs when staging an MSIX package which version is already present on the machine. Usually due to regular MSIX install.
0x800B0109	Sysstem.Exception: A certificate chain processed, but terminated in a root certificate which is not trusted by the trust provider. error 0x800B0109: Opening the loose file package from location **<PackageName>** failed Follow instructions to install certificate in **Trusted people.**

CimFS errors

Error/Behavior	Explanation/Fix
CIMFS image does not appear in the Disk Management.	This is by design. Disk Management is not aware of the CIMFS format.
CIMFS image has multiple files.	This is by design.

Kusto

For MSIX app attach we have defined custom operation listed in the List of MSIX app attach operations section.

When running a query there are fields that should be observed that help understand what the outcome of the operation is and what the operation was ran on:

ResType – Success/Error identifies the outcome of the operation

ResDesc – Short description of the operation

Props – Details list of the objects (MSIX packages) on which the operation was performed. For example,

```
{"NumExpectedPackages":2,"NumActualPackages":2,"Result.VSCode_1.15.0.0_x64
__81q6ced8g4aa0":"On demand registration successful for VSCode_1.15.0.0_x6
4__81q6ced8g4aa0","Result.GoogleChrome_68.36.76.0_x64__81q6ced8g4aa0":"On
demand registration successful for GoogleChrome_68.36.76.0_x64__81q6ced8g4
aa0"}
```

Sample Kusto queries

Scenario	Query

Host pool health check for MSIX app attach ("AppAttachHealthCheck") based on host pool name	```
	let GetDataFrom = (clusterName:string)
	{
	cluster(clusterName).database('WVD').
	RDOperation
	| where HostPool == "<HostPoolName>"
	| where Name == "AppAttachHealthCheck"
	| where TIMESTAMP >= datetime(2020-11-
	04T21:42:26.1265554Z)
	};
	union GetDataFrom('<cluster>')
	``

Session host health check for MSIX app attach for both dynamic and regular MSIX app attach stage ("AppAttachHealthCheck") based on host pool name	```
	//Health Check using Session Host Name and only
	 show failures
	let GetDataFrom = (clusterName:string)
	{
	cluster(clusterName).database('WVD').
	RDOperation
	| where HostInstance == "<sessionhost>"
	| where Name == "StageMsixPackages" or Name ==
	"DynamicStageMsixPackages" or Name == "AppAtta
	chHealthCheck"
	| where ResType != "Success"
	| where TIMESTAMP >= ago(1h)
	| project TIMESTAMP, AADTenantId, ActivityId, R
	ing, HostPool, HostInstance, Name, ResType, Res
	Signature, ResDesc, Props
	};
	 union GetDataFrom("<cluster>")
	``

Check for MSIX package registration outcome per host pool and tenant.	```
	let GetDataFrom = (clusterName:string)
	{
	cluster(clusterName).database('WVD').
	RDOperation
	| where HostPool == "<hostpool>"
	| where RDTenant == "<tenant>"
	| where Name == "RegisterMsixPackages"
	| where TIMESTAMP >= datetime(<timestamp>)
	};
	 union GetDataFrom('<cluster>')
	``

Get list of packages that have failed to stage.	```
	let GetDataFrom = (clusterName:string)
	{
	cluster(clusterName).database('WVD').
	RDOperation
	| where HostPool == "<HostPoolName>"
	| where Name == "AppAttachHealthCheck"
	| where ResTye != " Success"
	| where TIMESTAMP >= datetime(2020-11-
	04T21:42:26.1265554Z)
	};
	``

	union GetDataFrom('**<cluster>**') Look at the **Props** field there will be a list of the packages that outcome for each package.
List all MSIX app attach related operations that have messages related to the operation.	RDOperation | where Name contains "msix" or Name contains "appattach" | where HostPool == "**<hostpool>**" | where Props != "" | where PreciseTimeStamp > datetime(**<timestamp>**) | limit 1000

Event Viewer

The best way to utilize event viewer is to import the custom view from
https://github.com/stgeorgi/msixappattach.

Scenario	Query
0x80004005 AppAttachServiceImpl - AppAttachStageAsync: Mounting virtual disk for package '**<PacakgeName>**' failed with exception: System.ComponentModel.Win32Exception (0x80004005): VHD probably already attached at Microsoft.RDInfra.VirtualDiskAPI.VirtualDisk.<AttachAsync>d__23.MoveNext() in E:\agent_work\2\s\src\RDAgent\src\Service\VirtualDisk\VirtualDisk.cs:line 404	This error occurs when the RD Agent is trying to stage a that is already staged. The RD Agent has lost its stated due to restart, update, or a crash of it or its service. There are two options to fix this error: 1) Unmount the MSIX image (VHD(x) or CIM). Wait up to 5 minutes and the RD Agent will selfheal 2) Restart the VM
AppAttachServiceImpl - AppAttachStageAsync: Failed to get packages to staging: Microsoft.RDInfra.Shared.Common.RestError.RestException: Guid parameter must not be an empty GUID (Parameter 'lastPublishingEventGuid') ---> Microsoft.RDInfra.Shared.Common.RestError.InnerRestException: Guid parameter must not be an empty GUID (Parameter 'lastPublishingEventGuid')	This error occurs when the host pool in not configured to be part of the validation eniornment.

Reference

MSIX app attach in WVD terminology

Scenario	Query

MSIX container	An MSIX container is where MSIX apps are run. To learn more, see MSIX containers.
MSIX application	An application stored in an .MSIX file.
MSIX packages	An MSIX package is an MSIX file or application
MSIX share	An MSIX share is a network share that holds expanded MSIX packages. MSIX shares support SMB 3 or later. Applications get staged from this MSIX share without having to move application files to the system drive.
MSIX image	An MSIX image is a VHD, VHDx, or CIM file that contains one or more MSIX packaged applications. Each application is delivered in the MSIX image using the MSIXMGR tool.
Repackage	Repackaging takes a non-MSIX application and converts it into MSIX using the MSIX Packaging Tool (MPT). For more information, see MSIX Packaging Tool overview.
Expansion	Expanding an MSIX package is a multi-step process. Expansion takes the MSIX file and puts its content into a VHD(x) or CIM file.
Upload an MSIX package	Uploading an MSIX package involves uploading the VHD(x) or CIM that contains an expanded MSIX package to the MSIX share.

In Windows Virtual Desktop, uploads happen once per MSIX share. Once you upload a package, all host pools in the same subscription can reference it. |
| Add MSIX package to a host pool | In Windows Virtual Desktop, adding an MSIX package links it to a host pool. |
| Publish an MSIX package | In Windows Virtual Desktop, a published MSIX package must be assigned to an Active Directory Domain Service (AD DS) or Azure Active Directory (Azure AD) user or user group. |
| Staging | Staging involves two things:

Mounting the VHD(x) or CIM to the VM.
Notifying the OS that the MSIX package is available for registration. |
| Registration | Registration makes a staged MSIX package available for your users. Registering is on a per-user basis. If you haven't explicitly registered an app for that specific user, they won't be able to run the app.

There are two types of registration: regular and delayed. |
| Delayed registration | In delayed registration, each application assigned to the user is only partially registered. Partial registration means that the Start menu tile and double-click file associations are registered. Registration happens while the user signs into their session, so it has minimal impact on the time it takes to start using Windows Virtual Desktop. Registration completes only when the user runs the application in the MSIX package. |

	Delayed registration is currently the default configuration for MSIX app attach.
Dregeristration	Deregistration removes a registered but non-running MSIX package for a user. Deregistration happens while the user signs out of their session. During deregistration, MSIX app attach pushes application data specific to the user to the local user profile.
Destage	Destaging notifies the OS that an MSIX package or application that currently isn't running and isn't staged for any user can be unmounted. This removes all reference to it in the OS.

List of error message for MSIX app attach

Section	Action	Scenario	Current error message
MSIX Packages	Add a new MSIX package	No available session hosts	The host pool {0} does not contain available Virtual Machines.
	Add a new MSIX package	Error retrieving SH instances	There was an error retrieving the **session hosts**. Please refresh the page and try again
	Retrieve MSIX packages	Error retrieving MSIX instances	There was an error retrieving the **MSIX packages**. Please refresh the page and try again
	Change MSIX package state (batch)	Error updating resources	Title: Execute **Update** command on {0} selected entities; Desc: Success: {0}, Failed: {1}
	Remove MSIX packages (batch)	Error removing resources	Title: Execute **Delete** command on {0} selected entities; Desc: Success: {0}, Failed: {1}
	Change MSIX package state	Error updating resource	Failed to update **MSIX package** '{0}'
	Edit MSIX package	Error updating resource	Failed to update **MSIX package** '{0}'
	Remove MSIX package	Error removing resource	Failed to delete **MSIX package** '{0}'
	Add a new MSIX package	Error adding new package	Failed to add **MSIX package** '{0}'
Add or edit MSIX package pane	Extract image	Invalid path control validation	Enter a Windows rooted absolute path or a network shared path. Files names can't contain any of the following characters : * ? " < > | / and only VHD, VHDx, and CIM extensions are supported

	Extract image	Empty field control validation	The value must not be empty
	Extract image	ARM API error	*[Error message returned from the backend displayed below the input field]*
	Extract image	ARM API error	No MSIX packages could be retrieved from the image path
	Select an MSIX package	MSIX already exists in host pool	MSIX package already exists in host pool
Add or edit MSIX applications	Retrieve MSIX packages	Error retrieving MSIX instances	There was an error retrieving the **MSIX packages**. Please refresh the page and try again
	Select an MSIX package	There're no active packages in host pool	There are no active MSIX packages in the host pool {0}
	Select an MSIX package	Empty field control validation	The value must not be empty
	Select an MSIX application	Empty field control validation	The value must not be empty
Applications (App group manager)	Edit MSIX application	Error updating resource	Failed to update **application** '{0}'
	Remove MSIX application	Error removing resource	Failed to delete **application** '{0}'
	Add a new MSIX application	Error adding new package	Failed to add **application** '{0}'
	Remove MSIX applications (batch)	Error removing resources	Title: Execute **Delete** command on {0} selected entities; Desc: Success: {0}, Failed: {1}

List of MSIX app attach operations

The following operations are reported via Kusto and pertain to MSIX app attach.

- **AppAttachHealthCheck** – health check for the RD Agent. MSIX conditions are part of what sets this status. If MSIX packages cannot be stage the session host will become unavailable.

- **DeregisterMsixPackages** – this operation occurs when an MSIX package has been deregistered due to machine shut down or last admin user logging off from the machine.
- **DynamicDestageMsixPackages** – this operation occurs when the RD Agent dynamically destages an MSIX package.
- **DynamicStageMsixPackages** - this operation occurs when the RD Agent dynamically stages an MSIX package.
- **GetAppAttachPackageListAsync** - this operation occurs when the RD Agent queries the RD broker for the list of packages to be staged.
- **MountMsixDisk** – this operation occurs when the RD Agent mounts an MSIX image.
- **MSG:ExtractMsixDataAsync** – this operation occurs when the RD Agent is asked to expand an MSIX image.
- **ProcessMsixPackage** – this operation occurs when the RD Agent parses an MSIX image and verified MSIX application is a valid MSIX package.
- **RegisterMsixPackages** – this operation occurs when the RD Agent registers an application for user that has just established a connection to the session host.
- **TraverseMsixDisk** – this operation occurs when the RD Agent parses an MSIX image to enumerate all the MSIX applications inside.
- **GetMsixMinOsVersion** - this operation occurs when the RD Agent needs to check if the OS version is valid for MSIX
- **StageMsixPackages** - this operation occurs when the RD Agent first stages MSIX package
- **DestageMsixPackages** - this operation occurs when the RD Agent dynamically destages MSIX packages

List of MSIX extraction errors

Extraction occurs when IT pro is adding a MSIX image to a host pool. During that process the RD Agent is parsing the content of the MSIX image to extract information about the applications in that image.

Error message	Error code
Unknown	0
InvalidImageExtension	1
InvalidImagePath	2
DiskSharingViolation	3
MsixIntegrityViolation	4
PackageInvalidException	5
AccessException	6
StructureNoParentFolder	7
VirtualDiskNotFound	8
VirtualDiskImageMountFailure	9

VirtualDiskCorruption	10

Additional materials for troubleshooting MSIX

- Debugging MSIX packaged apps - https://docs.microsoft.com/en-us/windows-hardware/drivers/debugger/plmdebug
- Troubleshooting tips for packaged apps can be found here – https://docs.microsoft.com/en-us/windows/win32/appxpkg/troubleshooting