AppsAnywhere®

Troubleshooting Support Guide for AppsAnywhere

Contents

Who is this document for?

This document is aimed as a one-stop-go-to guide for new customers and those inducting new service desk colleagues on the day-to-day issues that end users may encounter when using AppsAnywhere.

It pulls together and links back to articles on the more extensive docs.appsanywhere.com documentation site, which is publicly accessible.

It may and may also link to knowledge base articles, which due to the information they may contain are behind a registration screen via AppsAnywhere Support. If you require access, then please ask your AppsAnywhere administrators to request an account.

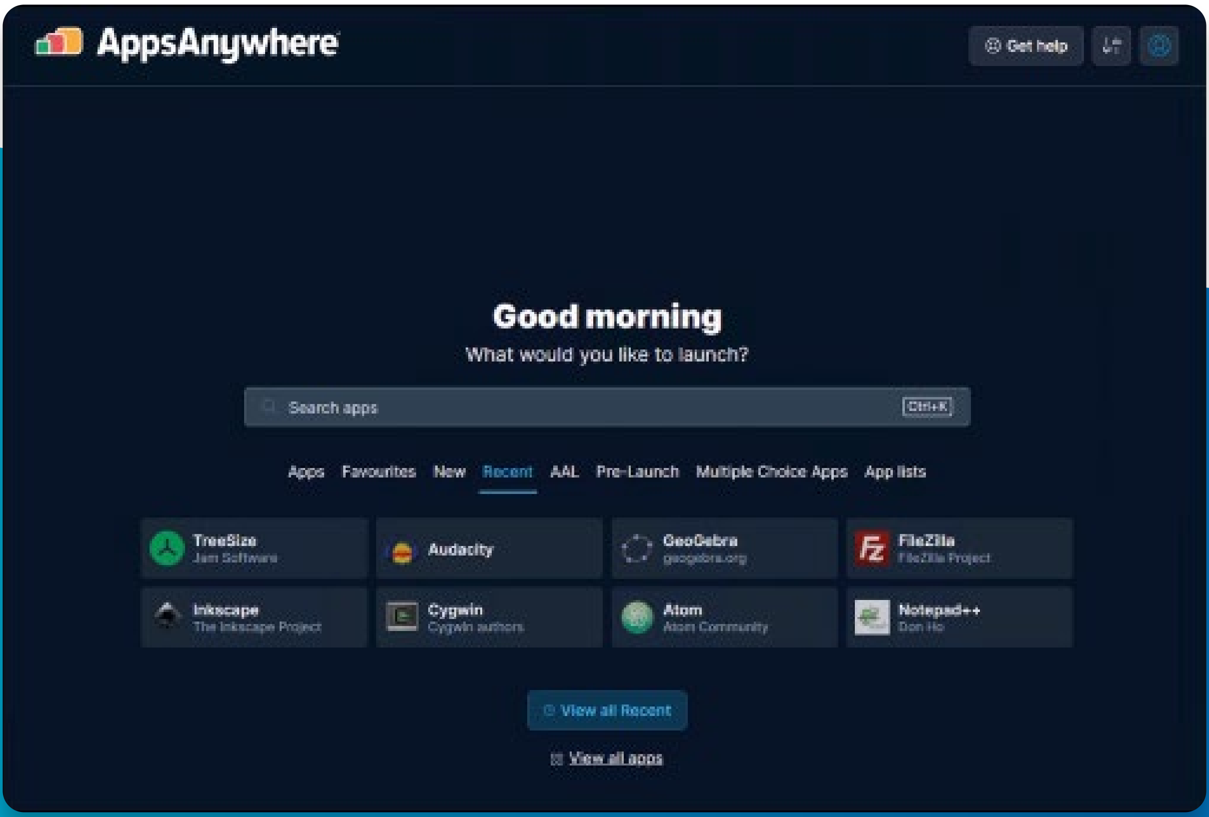
AppsAnywhere What is it?

AppsAnywhere is a web-based portal designed to ensure users receive applications in the most efficient way for their specific device.

These applications are presented in a simple app-store type interface via a secure logon, and allows you to specify which users can see which applications based on several device based environmental restrictions (device, location, BYOD/Managed device, RAM etc.)

The applications presented via the portal are effectively containers, which may host several **delivery methods**

These delivery methods range in scope from a simple URL link to open a **webpage** or a file, launching a physically installed **local application** to launching a virtualised Windows application **(Cloudpaging)** or launching a remote session via different **Connectors** such as **Parallels RAS**, **Azure Virtual Desktop** or **Amazon AppStream.**



Good morning

What would you like to launch?

The user is presented with the best delivery method based on the **restrictions** and choices you make when defining the application and are based on a number of environmentmental factors relating to the end users device, gathered during the logon process.

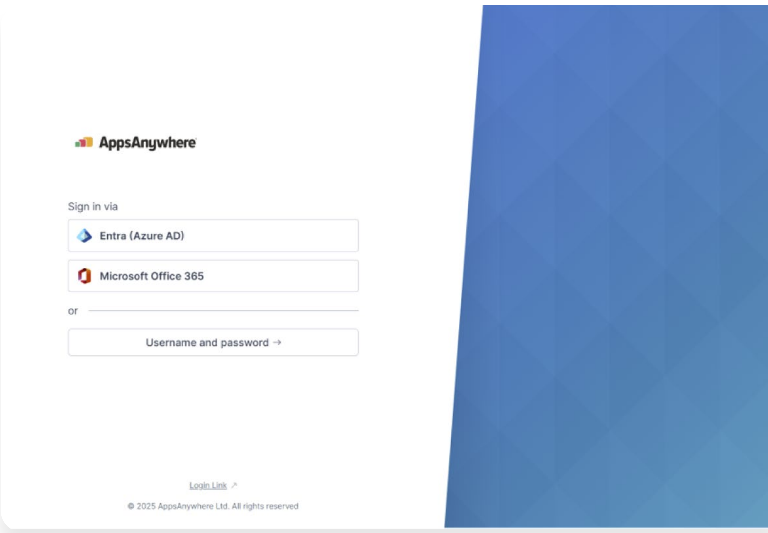
It is this logon process and the presentation of the applications that will be detailed further in this guide.

Accessing AppsAnywhere

The process follows two main steps: **authentication** and **validation.** Both are initiated by accessing the AppsAnywhere portal page, or in many cases via single sign-on (SSO) via that page:

1. User Authentication

- The user logs into AppsAnywhere using their credentials, or pass through via SSO.
- The AppsAnywhere portal presents all apps the user is provisioned access to.



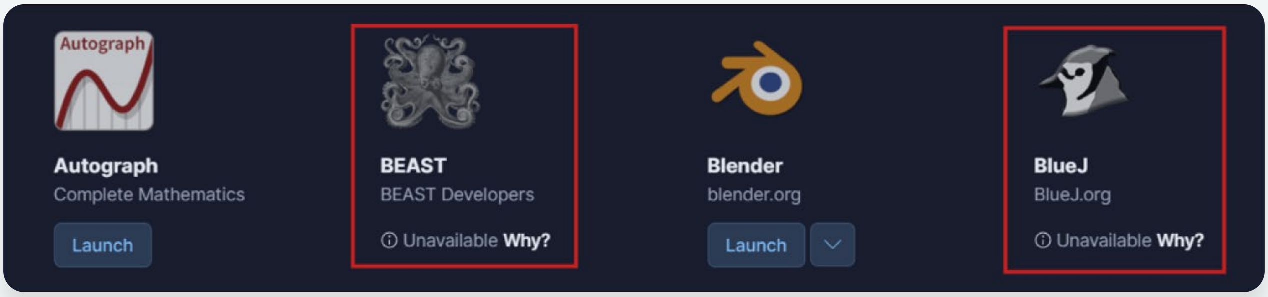
AppsAnywhere portal landing page showing different methods of login

2. Device Validation (MacOS and Windows)

- AppsAnywhere client analyzes the user's current device specifications.
- This step helps determine the best **delivery method** available for each provisioned application, ensuring compatibility and performance.
- If there are no suitable **delivery methods,** an app will show as **Unavailable.**
- When a user launches an app, AppsAnywhere delivers the application using the most suitable method for the user's device.

Common delivery methods include:

- **Locally Installed** Windows or Mac applications.
- **Downloads.**
- **External Websites.**
- **Cloudpaging,** Windows app virtualization for Windows devices.
- Hosted virtualized environments such as **Appstream, Frame**, **Parallels RAS**, **VMWare Horizon**, or our own managed **Cloud Delivery.**



Details on this process can be found here: **AppsAnywhere Process and Data flow**

By automating this process, AppsAnywhere ensures seamless access to applications while optimizing performance across different device types.

Troubleshooting

The key to troubleshooting issues in AppsAnywhere is to identify at which step they originate. The most common issues encountered in AppsAnywhere can be categorized as:

1. **Login Issues**
2. **Validation Issues**
3. **Access to Applications Issues**
4. **Delivery Method Issues**

Here are some terms this documentation references that can affect the errors you might see, or the troubleshooting steps you engage in:

- **Cloud-Hosted** – This refers to our Standard Deployment Model where the AppsAnywhere infrastructure is hosted by us using Azure Cloud services.
- **Self-Hosted –** Legacy & Limited Use - customer hosts the infrastructure on-premise and is responsible for all aspects of server, network and database **infrrastructure.**
- **BYOD (Bring Your Own Device)** - end user personal devices
- **Managed Devices -** organisation owned devices, typically domain joined.
- **App Admin** – Admins with rights to import users and provision applications

- **Systm Admin –** Admins with rights to system settings in addition to app admin permissions
- **SAML -** our default method of granting user access to AppsAnywhere. By using industry standard Security Assertion Markup Language 2.0 (SAML) access can be granted securely and can be easily configured to work in most environments.
- **IdP -** By levarging your existing Identity Provider (IdP) using SAML your users are validated according to your security standards (MFA etc) using a common interface that they are accustomed to

1. Login Issues

There are two primary authentication methods for AppsAnywhere.

1. **LDAPS (Active Directory, Open LDAP).**The common authentication method for self-hosted AppsAnywhere environments.
2. **SAML (Entra, Shibboleth, CAS, Okta, etc.).** SAML is the only recommended authentication method.

SAML can also be added to selft-hosted installations to improve the logon and single sign-on process.

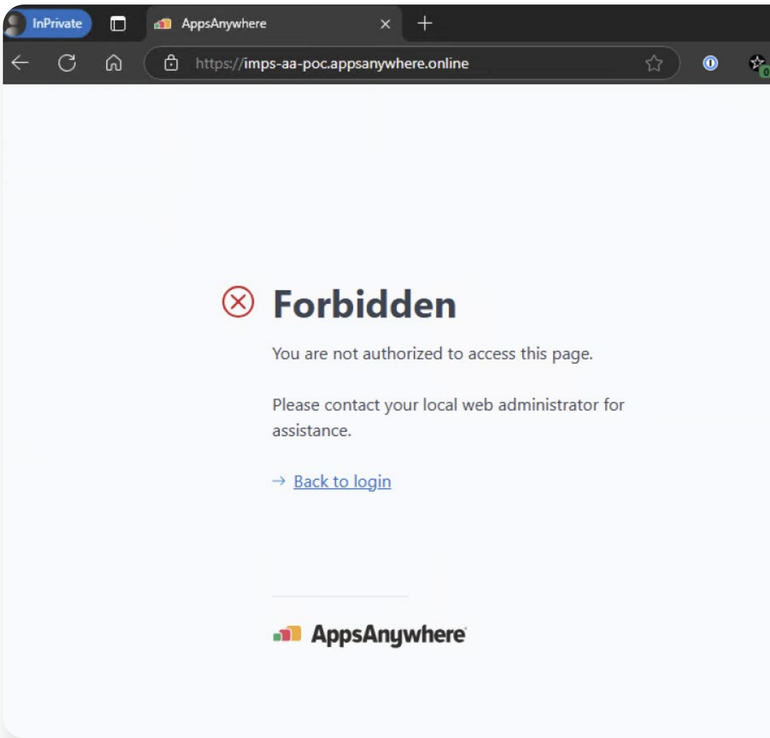
If you are using AppsAnywhere via our cloud hosted package, then the only supported login method is SAML

Forbidden page after logging in

Issue Text

Forbidden
You are not authorized to access this page.

Please contact your local web administrator for assistance.
Back to login



Forbidden

You are not authorized to access this page.

Please contact your local web administrator for assistance.

→ Back to login

Root Cause

User authenticated successfully, but they are not included in a Directory Mapping.

a. LDAPS: Importing directory entities

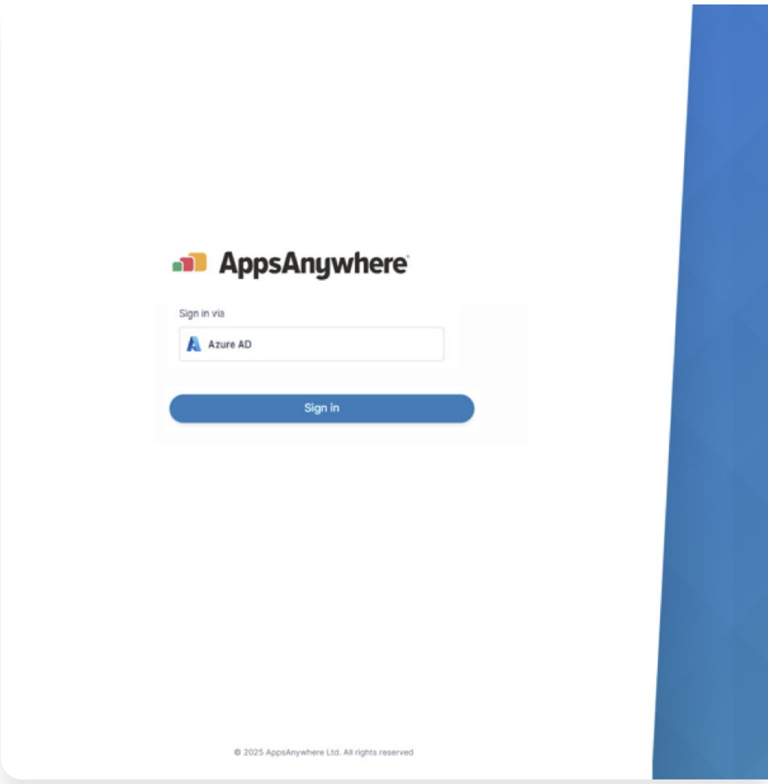
b. SAML: Creating SAML Attribute Mappings

User arrives back at the AppsAnywhere Login page after logging in via SAML

Root Cause

There are required fields in AppsAnywhere's **SSO Settings** that don't match the claim sent by the IdP. An AppsAnywhere system admin will want to confirm that the attribute names and certificate listed in the AppsAnywhere **SSO Settings** match what the IdP is sending.

As the SAML intergration is a key component to AppsAnywhere access to the settings and configuration are limitedto those support staff who have been granted System Admin rights in AppsAnywhere, however all users are able touse the troubleshotting tips to check what is being sent in the SAML response file.



Steps to Troubleshoot and Resolve

1. You can still use a SAML trace extension to capture the response from the IdP sent to AppsAnywhere after login.
   a. Many SAML tracer extensions exist for all browsers. Here is an example: **SAML, WS-Federation and OAuth 2.0 tracer - Microsoft Edge Addons**
2. If you have system admin level permissions in AppsAnywhere, compare the output in the Attribute and Certificate sections of the SAML trace with the data in AppsAnywhere's **SSO Settings.** Otherwise, send the SAML trace to a system admin who can check those settings.

Documentation: SAML 2.0 Common

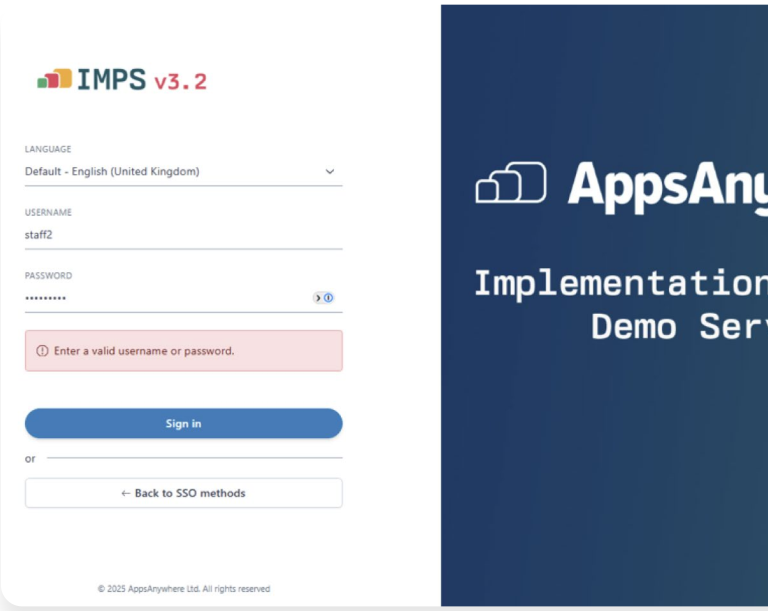
Active Directory or OpenLDAP Login Error

If an individual user is unable to logon, then basic checks of username and password, or account lockout should be undertaken.

If multiple or all user are having issues then your internal AppsAnywhere admins.

As the AD login uses a service account, this account would be the first place to check to see if this was locked out.

If the password has changed/expired then they will need to escalate this directly with AppsAnywhere via **https://support.appsanywhere.com**



LANGUAGE
Default - English (United Kingdom)

USERNAME
staff2

PASSWORD
•••••••••

ⓘ Enter a valid username or password.

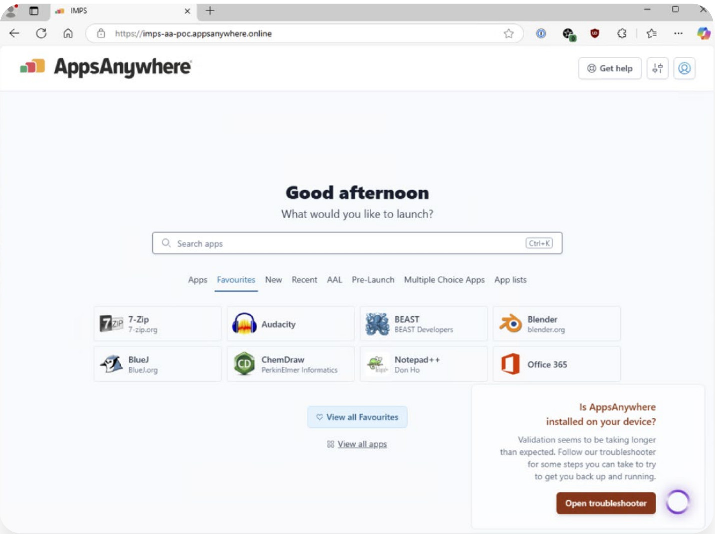
© 2025 AppsAnywhere Ltd. All rights reserved

Documentation: LDAP Connection Details

2. Validation Issues

After login, the AppsAnywhere portal attempts to validate a user's device to gather information. This is done via the AppsAnywhere Client (MacOS & Windows) The AppsAnywhere client gathers information from the end user device and is used to determine which delivery method to provide for each application the user has access to.

Validation involves the AppsAnywhere client running on the user's device and sending the output to the AppsAnywhere portal. A detailed breakdown of the validation process can be viewed here:



Good afternoon

What would you like to launch?

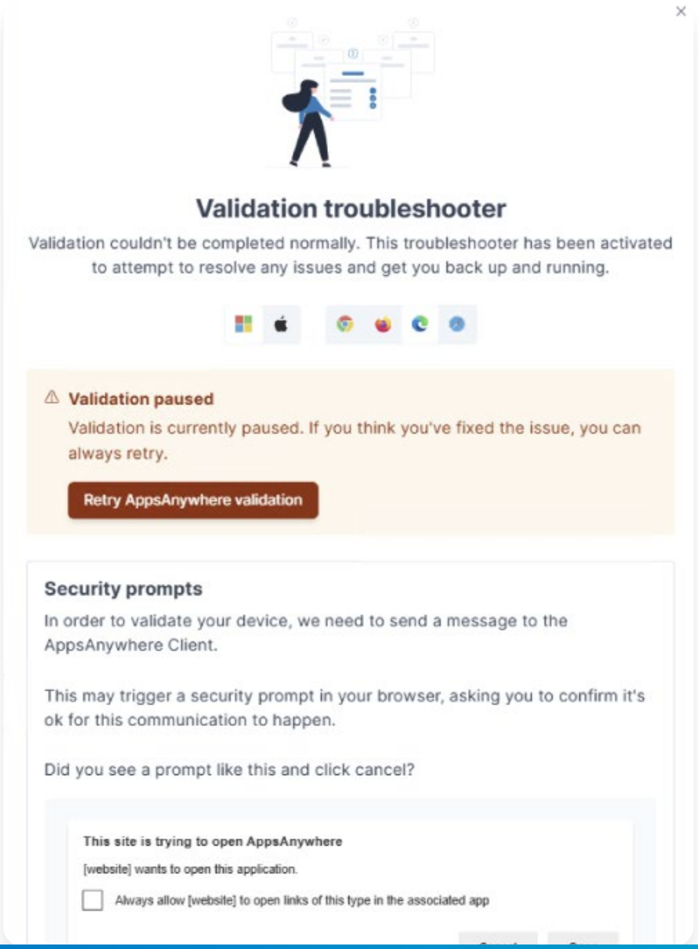
Is AppsAnywhere installed on your device?

Validation seems to be taking longer than expected. Follow our troubleshooter for some steps you can take to try to get you back up and running.

AppsAnywhere Validation Error

AppsAnywhere Process and Data flow - User Validation

If validation fails, a Validation Troubleshooter will appear to help guide the user through common causes.



Validation troubleshooter

Validation couldn't be completed normally. This troubleshooter has been activated to attempt to resolve any issues and get you back up and running.

⚠ **Validation paused**

Validation is currently paused. If you think you've fixed the issue, you can always retry.

Security prompts

In order to validate your device, we need to send a message to the AppsAnywhere Client.

This may trigger a security prompt in your browser, asking you to confirm it's ok for this communication to happen.

Did you see a prompt like this and click cancel?

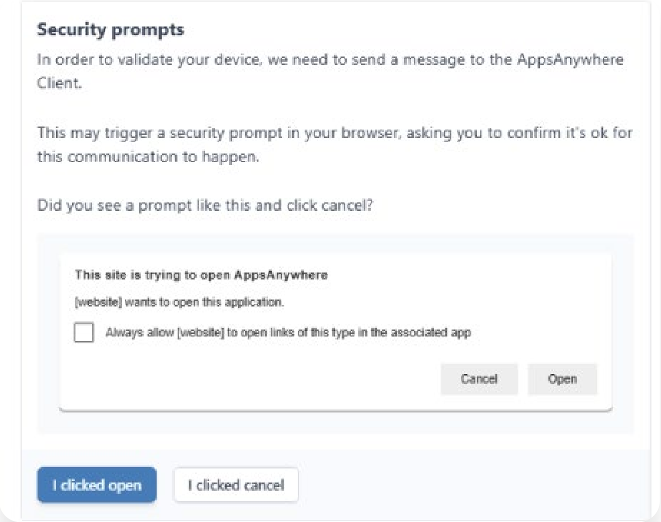
This site is trying to open AppsAnywhere

[website] wants to open this application.

□ Always allow [website] to open links of this type in the associated app

Troubleshooting Step 1:
Security Prompts

The first step in validation troubleshooting is ensuring the user approves the security prompt from the browser. Retry validation and make sure they click "Open".



Security prompts

In order to validate your device, we need to send a message to the AppsAnywhere Client.

This may trigger a security prompt in your browser, asking you to confirm it's ok for this communication to happen.

Did you see a prompt like this and click cancel?

This site is trying to open AppsAnywhere

[website] wants to open this application.

□ Always allow [website] to open links of this type in the associated app

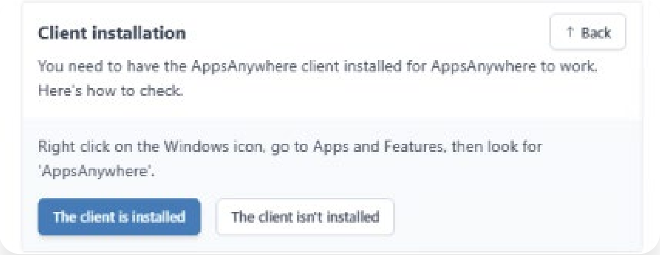
Cancel | Open

I clicked open | I clicked cancel

AppsAnywhere Troubleshooter

Troubleshooting Step 2:
Security Prompts

The next step is ensuring the AppsAnywhere client is installed on their machine. If it is not, the user can click the "The client isn't installed" button and get a download of the AppsAnywhere client installer to run on their device.



Client installation

↑ Back

You need to have the AppsAnywhere client installed for AppsAnywhere to work. Here's how to check.

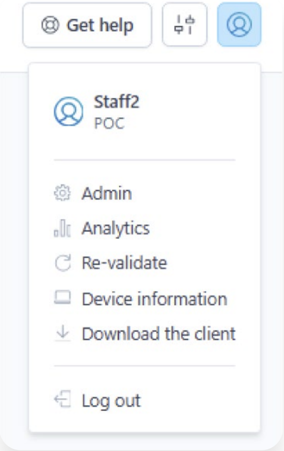
Right click on the Windows icon, go to Apps and Features, then look for 'AppsAnywhere'.

The client is installed | The client isn't installed

AppsAnywhere Troubleshooter

This download can also be accessed from the user profile menu.

If validation still isn't working, it's worth uninstalling the AppsAnywhere client on the user's device and installing it new.



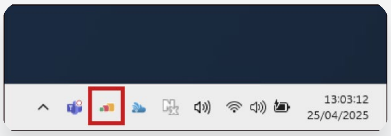
AppsAnywhere User Profile Menu

Still Not Working?

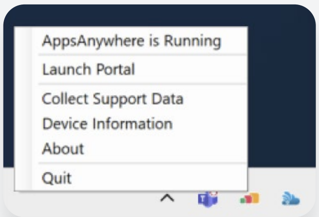
If you are not able to resolve then this should initially be escalated within your organisations IT department, and if necessary, then with AppsAnywhere support.
In all cases, it is essential than the full client log files are included, as detailed in the next section.

Collecting Log Files

The AppsAnywhere client can be used to easily collate all necessary log files from the client device. These will include logs and event viewer files for the AppsAnywhere and Cloudpaging Player clients, as well as logs for any 3rd party Intergrations that may be configured in AppsAnywhere, such as Parallels, Citrix, Vmware Horizon.



Right-click on the **AppsAnywhere** client shortcut in the system tray.



Select the **Collect Support Data** option to export all log files

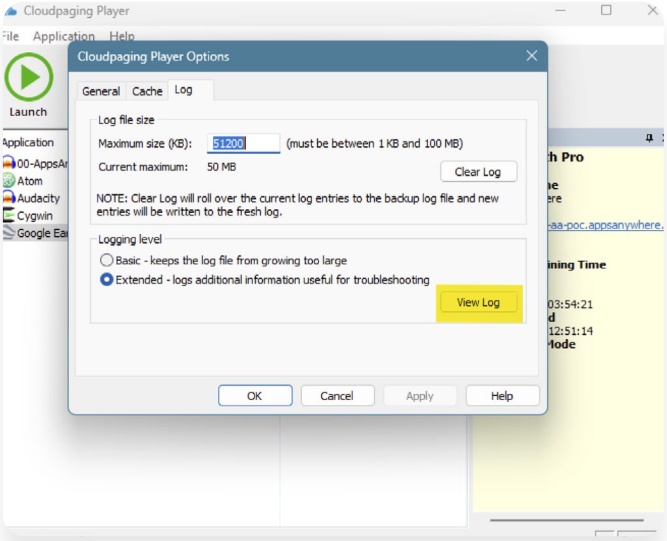
AppsAnywhere

The simplest way to collect the AppsAnywhere log is via the AppsAnywhere system tray icon detailed above. If manually collecting the log file, then the default location is:

```
C:\Users\<user>\AppData\Local\Software2\AppsAnywhere\Log\S2HubTracelog.log
```

Cloudpaging

The Cloudpaging player log can be viewed directly from the Cloudpaging Player via File Options menu, then selecting the Logs tab.



Log file size

Maximum size (KB): 51200 (must be between 1 KB and 100 MB)

Current maximum: 50 MB

NOTE: Clear Log will roll over the current log entries to the backup log file and new entries will be written to the fresh log.

Logging level

○ Basic - keeps the log file from growing too large

● Extended - logs additional information useful for troubleshooting

The default location of the log file is

```
C:\ProgramData\Endeavors Technologies\StreamingCore\Log\StreamingCore.log
```

Third Party Client Logs

If your institution is using Citrix, Parallels or other third party connectors, then these will have their ownn specific log files. More details can be found in our article:

AppsAnywhere Log Files – AppsAnywhere Support *(registration needed)*

Again, the easiest way to collect these is via the AppsAnywhere system tray icon, which will gather all available logs and Event Viewer logs.

Escalation

If you need to escalate a issue, both internally with your own helpdesk, or to the AppsAnywhere support team, we do advise collecting the full support bundle and including in any escalation ticket, and not copy the one or two lines withing the log file that may highlight an issue.

3. Access to Applications Issues

Access Issue:
User Cannot Find Their App

Users can only see applications in AppsAnywhere if they are granted access via an AppsAnywhere Provision.

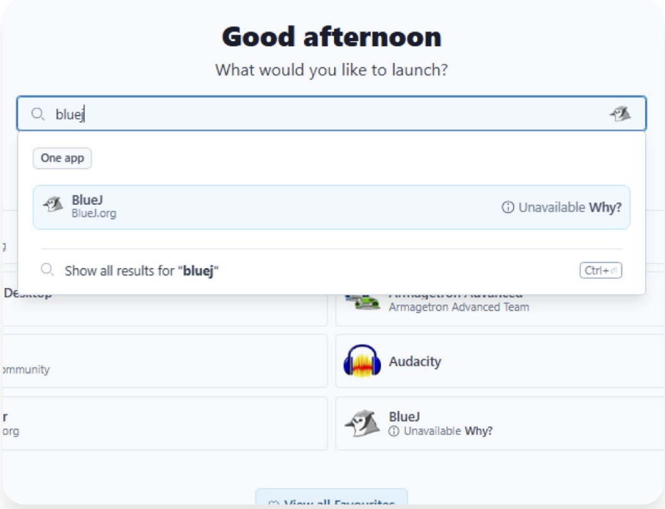
Documentation: AppsAnywhere Provisioning

1. Add the user to a provision that includes the app in question. Most often this is done by adding them to a group already provisioned to access that app, but they can also be added individually to a provision.
2. Have the user log out and log back in. They should now see their application.

Access Issue:
Unavailable Applications

An application shows unavailable for a user when they are permitted access to that application via a provision, but there is no way to deliver the app to them.

Most of the time, this is due to the application having no delivery methods which support them or their machine. There is one scenario when a non-delivery method setting could cause an application to become unavailable.



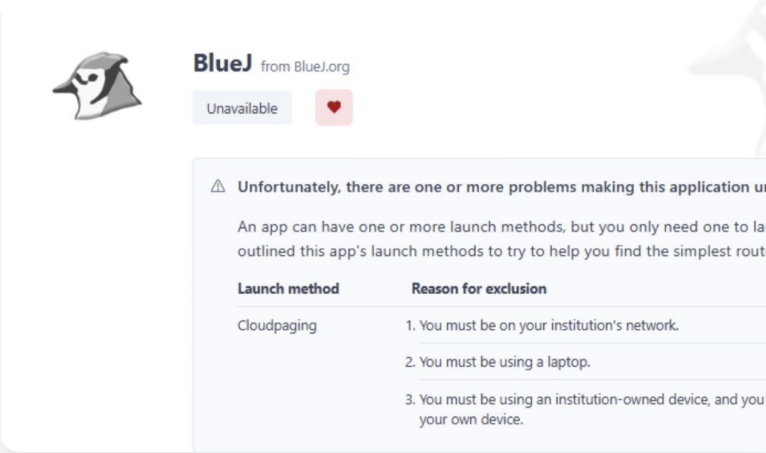
Good afternoon

What would you like to launch?

Delivery Method settings that can cause an Unavailable Application

The user's device failed to meet any set delivery method restrictions.

Documentation: Restrictions



BlueJ from BlueJ.org

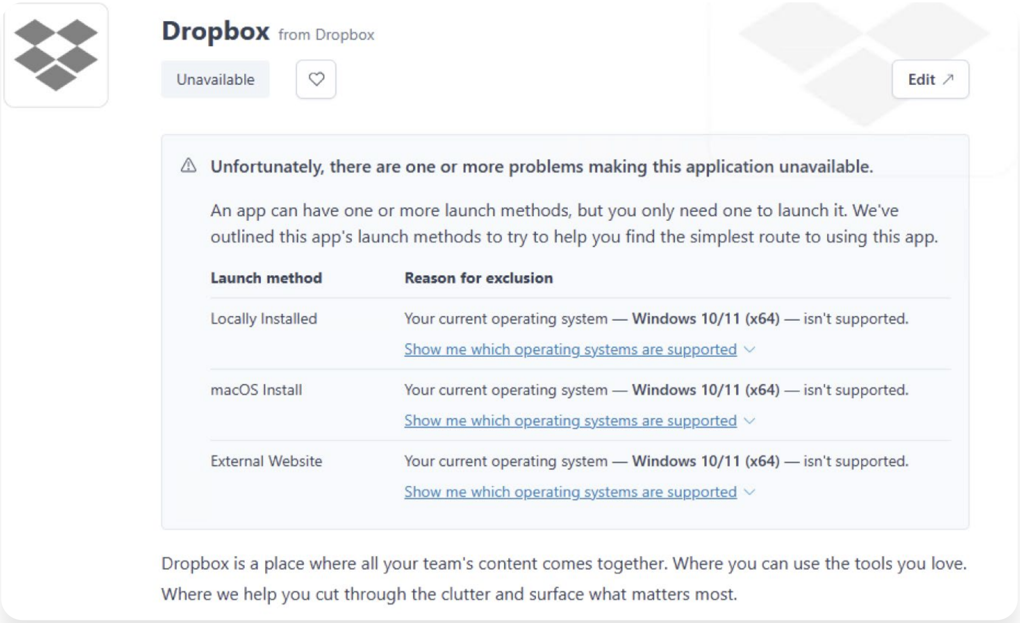
⚠ **Unfortunately, there are one or more problems making this application u**

An app can have one or more launch methods, but you only need one to la outlined this app's launch methods to try to help you find the simplest rout

Launch method	Reason for exclusion
Cloudpaging	1. You must be on your institution's network. 2. You must be using a laptop. 3. You must be using an institution-owned device, and you your own device.

AppsAnywhere will attempt to inform the user of any delivery method restrictions theirdevice failed to meet for an application listed Unavailable.

The delivery method does not support the user’s operating system.



⚠ **Unfortunately, there are one or more problems making this application unavailable.**

An app can have one or more launch methods, but you only need one to launch it. We've outlined this app's launch methods to try to help you find the simplest route to using this app.

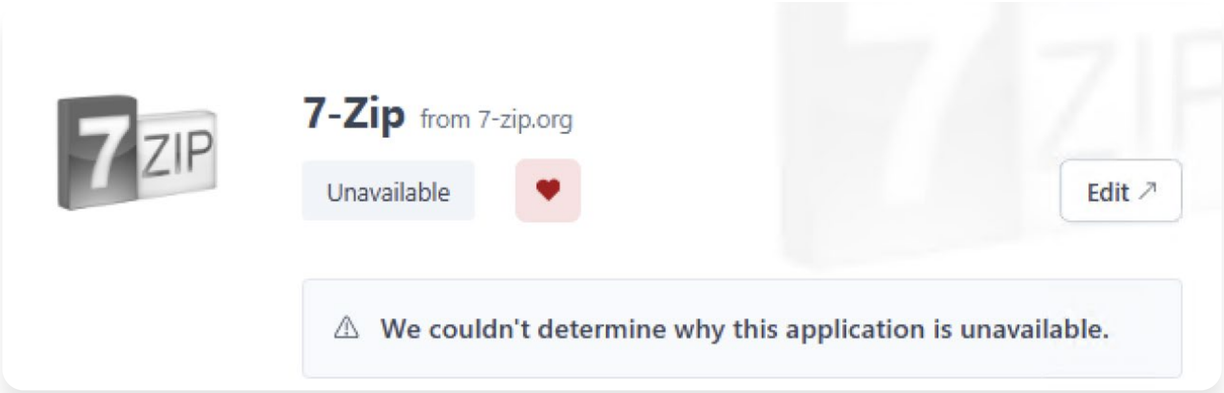
Launch method	Reason for exclusion
Locally Installed	Your current operating system — **Windows 10/11 (x64)** — isn't supported. Show me which operating systems are supported
macOS Install	Your current operating system — **Windows 10/11 (x64)** — isn't supported. Show me which operating systems are supported
External Website	Your current operating system — **Windows 10/11 (x64)** — isn't supported. Show me which operating systems are supported

Dropbox is a place where all your team's content comes together. Where you can use the tools you love. Where we help you cut through the clutter and surface what matters most.

Similar to restrictions, AppsAnywhere will attempt to inform the user of the supported operating systems for delivery methods of an application listed Unavailable.

Documentation: Operating System Compatibility

The delivery method has a Mapping defined which doesn’t include the user

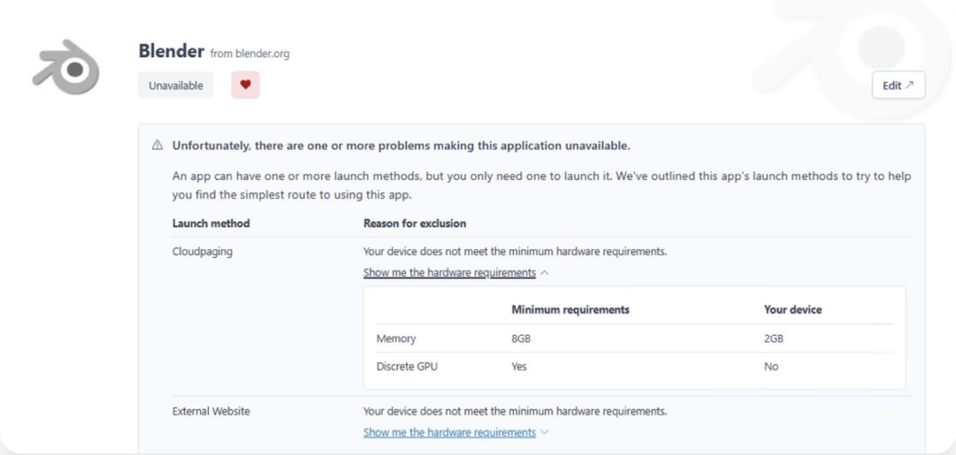


⚠ We couldn't determine why this application is unavailable.

Similar to restrictions, AppsAnywhere will attempt to inform the user of the supported operating systems for delivery methods of an application listed Unavailable.

Documentation: Delivery Method Mappings

The device does not meet the delivery method’s hardware requirements.



⚠ **Unfortunately, there are one or more problems making this application unavailable.**

An app can have one or more launch methods, but you only need one to launch it. We've outlined this app's launch methods to try to help you find the simplest route to using this app.

Launch method	Reason for exclusion
Cloudpaging	Your device does not meet the minimum hardware requirements. Show me the hardware requirements ^
External Website	Your device does not meet the minimum hardware requirements. Show me the hardware requirements ⌄

	Minimum requirements	Your device
Memory	8GB	2GB
Discrete GPU	Yes	No

Similar to restrictions, AppsAnywhere will attempt to inform the user of the supported operating systems for delivery methods of an application listed Unavailable.

There are no settings limiting access to the user or their device, but the delivery method(s) themselves do not support their device. See the Delivery Method Troubleshooting section below for details.

Documentation: Hardware Profiles

Other reasons for an unavailable Application

Restrictions are a key component in configuring AppsAnywhere, and this article explains the range variousmethod by which access to an application or an individual delivery method can be granted or restricited.

4.Delivery Method Launch Issues

Locally Installed Issues

AppsAnywhere can detect locally installed apps on Windows or MacOS.
If an app is not detected as locally installed by the AppsAnywhere client, it will be due to the information set in the Locally Installed delivery method for the app in AppsAnywhere not matching the app installed on the machine.

Documentation: Locally Installed Delivery Method

If you lack admin permissions in AppsAnywhere and need to escalate to someone who does, here is the useful information to gather from the user's machine for the app in question:

Windows

Information	Example(s)
App Paths registry key	application.exe
Uninstall registry keyregistry key	{3B7E914A-93D5-4A29-92BB-AF8C3F66C431} Audacity®_is1
App directory containing the launch executable	\Application\bin

Mac

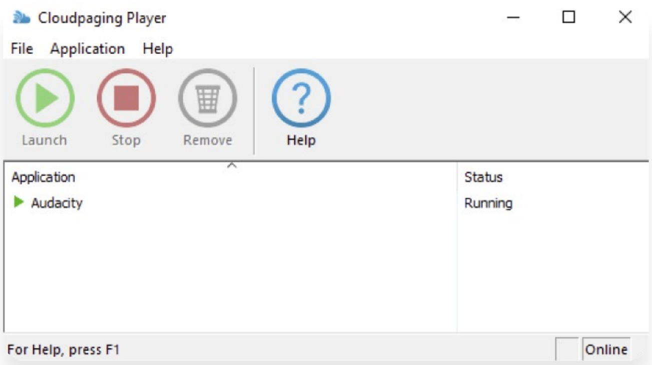
Information	Example(s)
MacOS application bundle name	VLC.app

Cloudpaging Issues

The best place to start troubleshooting Cloudpaging issues is to first determine if the app ever gets to the “Running” state in the player.

Running

A Cloudpaged app that gets to the “Running” status in the player has launched with no issues from the Cloudpaging point of view. The player will display the status as “Running”.



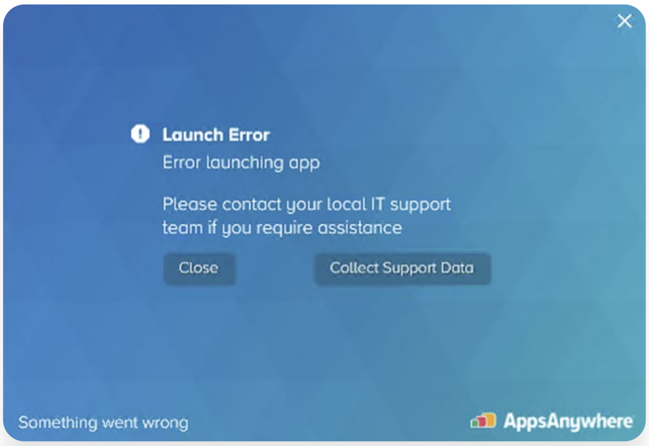
Application	Status
Audacity	Running

For Help, press F1

Online

Not Running

A Cloudpaged app that gets to the “Running” status in the player has launched with no issues from the Cloudpaging point of view. The player will display the status as “Running”.

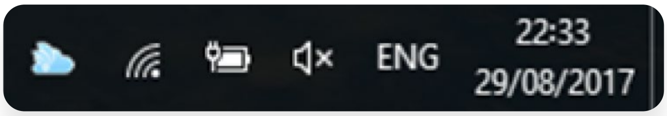


Launch Error

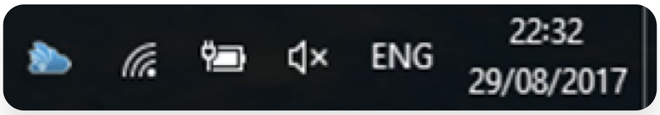
Error launching app

Please contact your local IT support team if you require assistance

Flashing icon when the app is paging



Normal icon



Cloudpaging Issue:
Application takes a very long time to Launch

This can happen if the Player is taking a very long time to get the data it needs from the server. This can happen for twoprimary reasons:

1. “Noise” in the Cloudpaging Package. Noise refers to unnecessary assets in a Cloudpaging package. If an appsethas a large amount of noise, it can interfere with the data the Player needs to run the app and can impact the timeit takes to get that data.

   If the appset was packaged outside of our packaging service, it’s worth inspecting the appset for any noise orother ways to make it more efficient.

ⓘ

If you suspect that the application is deploying assets or includes ‘noise’ that are causing issues, and the appset has been created internally by your instuitution, then this will need to be referred back to your colleagues responsible for creating the appset.
If the appset is pre-packaged by AppsAnywhere, then that same team should validate your findings, and then if necessary refer back to AppsAnywhere support

For larger and more complex applications then Pre-caching can be used to bring the data closer to managed machines tospeed up the first-time launch of applications.

Documentation: How to Configure and Use Pre-caching on Managed Devices – AppsAnywhere Support *registration required*

Cloudpaging Issue:
AppsAnywhere Client Launch Error

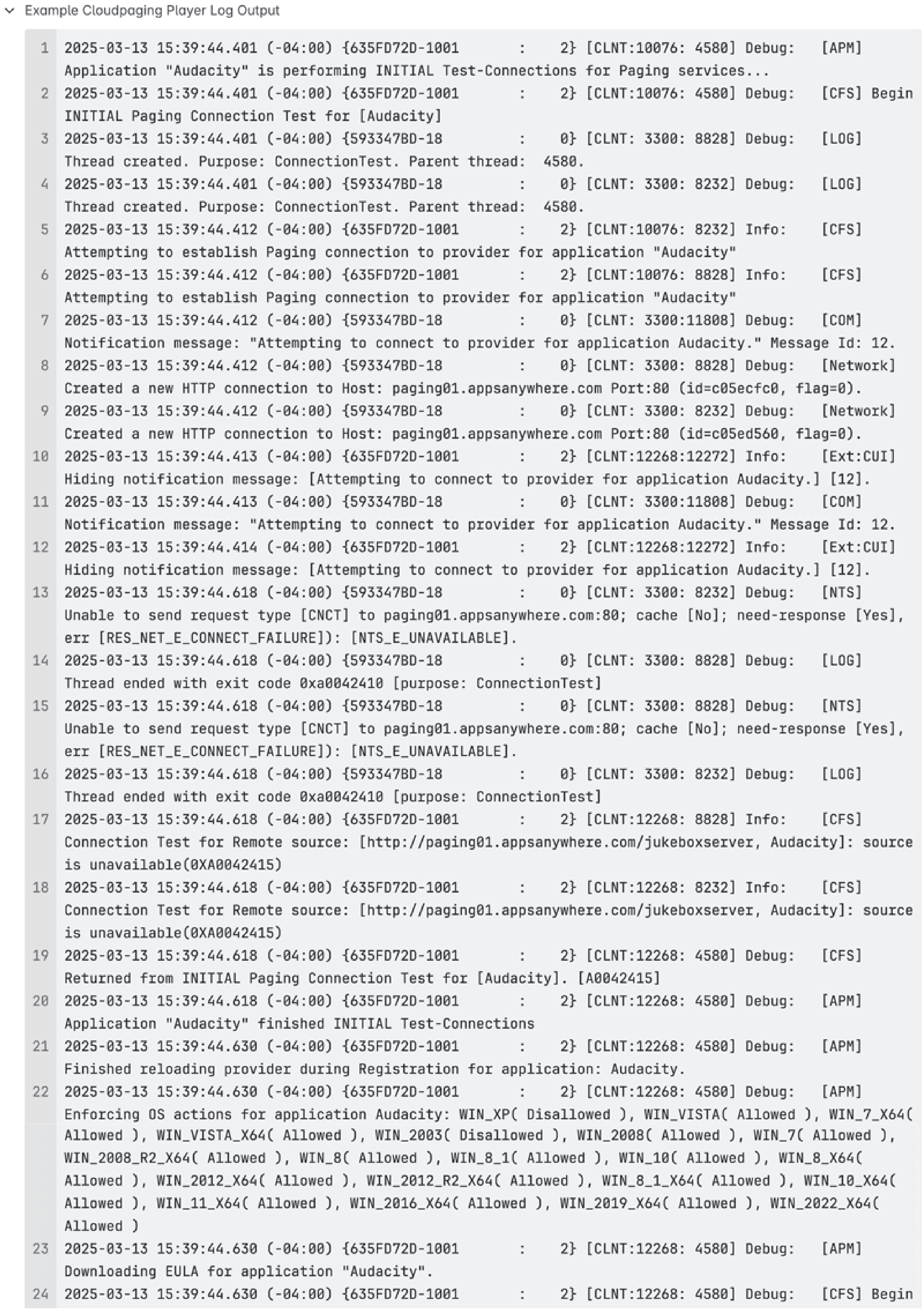
Sometimes an app doesn’t launch in the Cloudpaging Player at all. In this case, the AppsAnywhere client will display a Launch Error.

Troubleshooting Steps

First, check the client log files. The Cloudpaging Player log can be found at

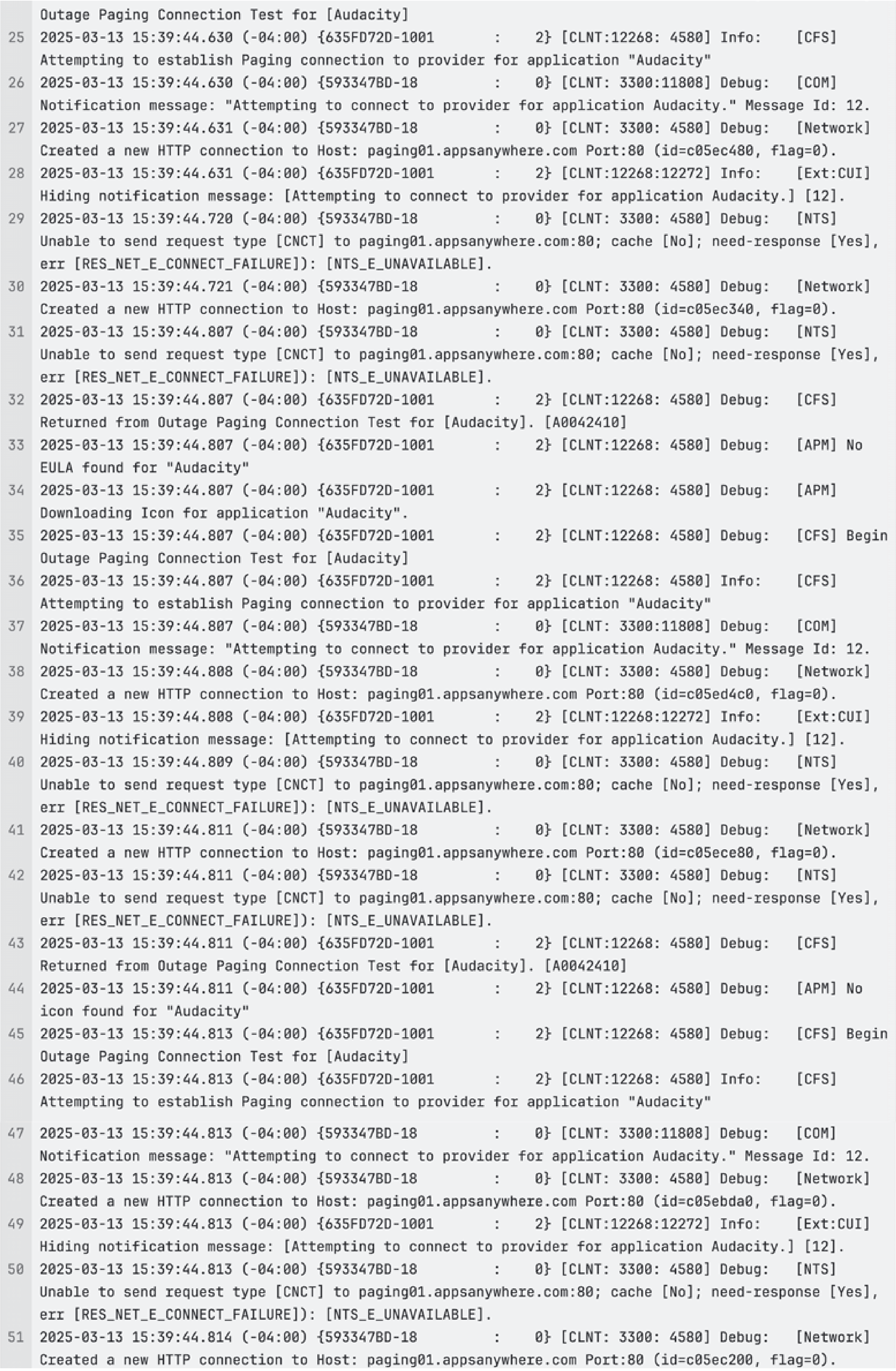
```
C:\ProgramData\Endeavors Technologies\StreamingCore\Log\StreamingCore.log
```

Example: Paging Servers Unavailable

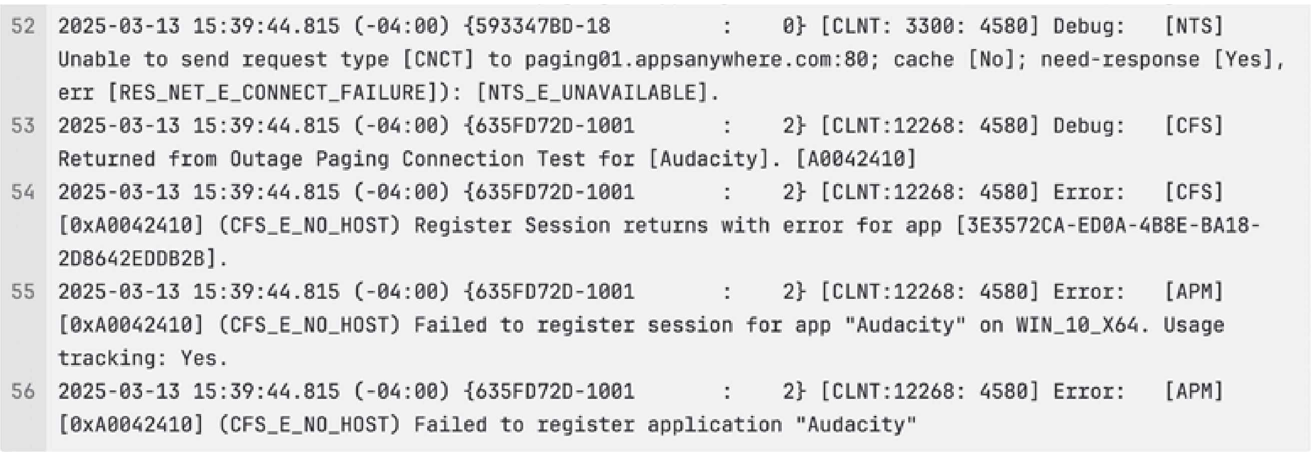


Example Cloudpaging Player Log Output

```
2025-03-13 15:39:44.401 (-04:00) {635FD72D-1001       :    2} [CLNT:10076: 4580] Debug:   [APM] Application "Audacity" is performing INITIAL Test-Connections for Paging services...
2025-03-13 15:39:44.401 (-04:00) {635FD72D-1001       :    2} [CLNT:10076: 4580] Debug:   [CFS] Begin INITIAL Paging Connection Test for [Audacity]
2025-03-13 15:39:44.401 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 8828] Debug:   [LOG] Thread created. Purpose: ConnectionTest. Parent thread:  4580.
2025-03-13 15:39:44.401 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 8232] Debug:   [LOG] Thread created. Purpose: ConnectionTest. Parent thread:  4580.
2025-03-13 15:39:44.412 (-04:00) {635FD72D-1001       :    2} [CLNT:10076: 8232] Info:    [CFS] Attempting to establish Paging connection to provider for application "Audacity"
2025-03-13 15:39:44.412 (-04:00) {635FD72D-1001       :    2} [CLNT:10076: 8828] Info:    [CFS] Attempting to establish Paging connection to provider for application "Audacity"
2025-03-13 15:39:44.412 (-04:00) {593347BD-18         :    0} [CLNT: 3300:11808] Debug:   [COM] Notification message: "Attempting to connect to provider for application Audacity." Message Id: 12.
2025-03-13 15:39:44.412 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 8828] Debug:   [Network] Created a new HTTP connection to Host: paging01.appsanywhere.com Port:80 (id=c05ecfc0, flag=0).
2025-03-13 15:39:44.412 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 8232] Debug:   [Network] Created a new HTTP connection to Host: paging01.appsanywhere.com Port:80 (id=c05ed560, flag=0).
2025-03-13 15:39:44.413 (-04:00) {635FD72D-1001       :    2} [CLNT:12268:12272] Info:    [Ext:CUI] Hiding notification message: [Attempting to connect to provider for application Audacity.] [12].
2025-03-13 15:39:44.413 (-04:00) {593347BD-18         :    0} [CLNT: 3300:11808] Debug:   [COM] Notification message: "Attempting to connect to provider for application Audacity." Message Id: 12.
2025-03-13 15:39:44.414 (-04:00) {635FD72D-1001       :    2} [CLNT:12268:12272] Info:    [Ext:CUI] Hiding notification message: [Attempting to connect to provider for application Audacity.] [12].
2025-03-13 15:39:44.618 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 8232] Debug:   [NTS] Unable to send request type [CNCT] to paging01.appsanywhere.com:80; cache [No]; need-response [Yes], err [RES_NET_E_CONNECT_FAILURE]): [NTS_E_UNAVAILABLE].
2025-03-13 15:39:44.618 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 8828] Debug:   [LOG] Thread ended with exit code 0xa0042410 [purpose: ConnectionTest]
2025-03-13 15:39:44.618 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 8828] Debug:   [NTS] Unable to send request type [CNCT] to paging01.appsanywhere.com:80; cache [No]; need-response [Yes], err [RES_NET_E_CONNECT_FAILURE]): [NTS_E_UNAVAILABLE].
2025-03-13 15:39:44.618 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 8232] Debug:   [LOG] Thread ended with exit code 0xa0042410 [purpose: ConnectionTest]
2025-03-13 15:39:44.618 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 8828] Info:    [CFS] Connection Test for Remote source: [http://paging01.appsanywhere.com/jukeboxserver, Audacity]: source is unavailable(0XA0042415)
2025-03-13 15:39:44.618 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 8232] Info:    [CFS] Connection Test for Remote source: [http://paging01.appsanywhere.com/jukeboxserver, Audacity]: source is unavailable(0XA0042415)
2025-03-13 15:39:44.618 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Debug:   [CFS] Returned from INITIAL Paging Connection Test for [Audacity]. [A0042415]
2025-03-13 15:39:44.618 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Debug:   [APM] Application "Audacity" finished INITIAL Test-Connections
2025-03-13 15:39:44.630 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Debug:   [APM] Finished reloading provider during Registration for application: Audacity.
2025-03-13 15:39:44.630 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Debug:   [APM] Enforcing OS actions for application Audacity: WIN_XP( Disallowed ), WIN_VISTA( Allowed ), WIN_7_X64( Allowed ), WIN_VISTA_X64( Allowed ), WIN_2003( Disallowed ), WIN_2008( Allowed ), WIN_7( Allowed ), WIN_2008_R2_X64( Allowed ), WIN_8( Allowed ), WIN_8_1( Allowed ), WIN_10( Allowed ), WIN_8_X64( Allowed ), WIN_2012_X64( Allowed ), WIN_2012_R2_X64( Allowed ), WIN_8_1_X64( Allowed ), WIN_10_X64( Allowed ), WIN_11_X64( Allowed ), WIN_2016_X64( Allowed ), WIN_2019_X64( Allowed ), WIN_2022_X64( Allowed )
2025-03-13 15:39:44.630 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Debug:   [APM] Downloading EULA for application "Audacity".
2025-03-13 15:39:44.630 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Debug:   [CFS] Begin
```



```
Outage Paging Connection Test for [Audacity]
2025-03-13 15:39:44.630 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Info:    [CFS]
Attempting to establish Paging connection to provider for application "Audacity"
2025-03-13 15:39:44.630 (-04:00) {593347BD-18         :    0} [CLNT: 3300:11808] Debug:   [COM]
Notification message: "Attempting to connect to provider for application Audacity." Message Id: 12.
2025-03-13 15:39:44.631 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 4580] Debug:   [Network]
Created a new HTTP connection to Host: paging01.appsanywhere.com Port:80 (id=c05ec480, flag=0).
2025-03-13 15:39:44.631 (-04:00) {635FD72D-1001       :    2} [CLNT:12268:12272] Info:    [Ext:CUI]
Hiding notification message: [Attempting to connect to provider for application Audacity.] [12].
2025-03-13 15:39:44.720 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 4580] Debug:   [NTS]
Unable to send request type [CNCT] to paging01.appsanywhere.com:80; cache [No]; need-response [Yes],
err [RES_NET_E_CONNECT_FAILURE]): [NTS_E_UNAVAILABLE].
2025-03-13 15:39:44.721 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 4580] Debug:   [Network]
Created a new HTTP connection to Host: paging01.appsanywhere.com Port:80 (id=c05ec340, flag=0).
2025-03-13 15:39:44.807 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 4580] Debug:   [NTS]
Unable to send request type [CNCT] to paging01.appsanywhere.com:80; cache [No]; need-response [Yes],
err [RES_NET_E_CONNECT_FAILURE]): [NTS_E_UNAVAILABLE].
2025-03-13 15:39:44.807 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Debug:   [CFS]
Returned from Outage Paging Connection Test for [Audacity]. [A0042410]
2025-03-13 15:39:44.807 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Debug:   [APM] No
EULA found for "Audacity"
2025-03-13 15:39:44.807 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Debug:   [APM]
Downloading Icon for application "Audacity".
2025-03-13 15:39:44.807 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Debug:   [CFS] Begin
Outage Paging Connection Test for [Audacity]
2025-03-13 15:39:44.807 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Info:    [CFS]
Attempting to establish Paging connection to provider for application "Audacity"
2025-03-13 15:39:44.807 (-04:00) {593347BD-18         :    0} [CLNT: 3300:11808] Debug:   [COM]
Notification message: "Attempting to connect to provider for application Audacity." Message Id: 12.
2025-03-13 15:39:44.808 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 4580] Debug:   [Network]
Created a new HTTP connection to Host: paging01.appsanywhere.com Port:80 (id=c05ed4c0, flag=0).
2025-03-13 15:39:44.808 (-04:00) {635FD72D-1001       :    2} [CLNT:12268:12272] Info:    [Ext:CUI]
Hiding notification message: [Attempting to connect to provider for application Audacity.] [12].
2025-03-13 15:39:44.809 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 4580] Debug:   [NTS]
Unable to send request type [CNCT] to paging01.appsanywhere.com:80; cache [No]; need-response [Yes],
err [RES_NET_E_CONNECT_FAILURE]): [NTS_E_UNAVAILABLE].
2025-03-13 15:39:44.811 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 4580] Debug:   [Network]
Created a new HTTP connection to Host: paging01.appsanywhere.com Port:80 (id=c05ece80, flag=0).
2025-03-13 15:39:44.811 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 4580] Debug:   [NTS]
Unable to send request type [CNCT] to paging01.appsanywhere.com:80; cache [No]; need-response [Yes],
err [RES_NET_E_CONNECT_FAILURE]): [NTS_E_UNAVAILABLE].
2025-03-13 15:39:44.811 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Debug:   [CFS]
Returned from Outage Paging Connection Test for [Audacity]. [A0042410]
2025-03-13 15:39:44.811 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Debug:   [APM] No
icon found for "Audacity"
2025-03-13 15:39:44.813 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Debug:   [CFS] Begin
Outage Paging Connection Test for [Audacity]
2025-03-13 15:39:44.813 (-04:00) {635FD72D-1001       :    2} [CLNT:12268: 4580] Info:    [CFS]
Attempting to establish Paging connection to provider for application "Audacity"
2025-03-13 15:39:44.813 (-04:00) {593347BD-18         :    0} [CLNT: 3300:11808] Debug:   [COM]
Notification message: "Attempting to connect to provider for application Audacity." Message Id: 12.
2025-03-13 15:39:44.813 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 4580] Debug:   [Network]
Created a new HTTP connection to Host: paging01.appsanywhere.com Port:80 (id=c05ebda0, flag=0).
2025-03-13 15:39:44.813 (-04:00) {635FD72D-1001       :    2} [CLNT:12268:12272] Info:    [Ext:CUI]
Hiding notification message: [Attempting to connect to provider for application Audacity.] [12].
2025-03-13 15:39:44.813 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 4580] Debug:   [NTS]
Unable to send request type [CNCT] to paging01.appsanywhere.com:80; cache [No]; need-response [Yes],
err [RES_NET_E_CONNECT_FAILURE]): [NTS_E_UNAVAILABLE].
2025-03-13 15:39:44.814 (-04:00) {593347BD-18         :    0} [CLNT: 3300: 4580] Debug:   [Network]
Created a new HTTP connection to Host: paging01.appsanywhere.com Port:80 (id=c05ec200, flag=0).
```



```
52  2025-03-13 15:39:44.815 (-04:00) {593347BD-18        :    0} [CLNT: 3300: 4580] Debug:   [NTS]
    Unable to send request type [CNCT] to paging01.appsanywhere.com:80; cache [No]; need-response [Yes],
    err [RES_NET_E_CONNECT_FAILURE]): [NTS_E_UNAVAILABLE].
53  2025-03-13 15:39:44.815 (-04:00) {635FD72D-1001      :    2} [CLNT:12268: 4580] Debug:   [CFS]
    Returned from Outage Paging Connection Test for [Audacity]. [A0042410]
54  2025-03-13 15:39:44.815 (-04:00) {635FD72D-1001      :    2} [CLNT:12268: 4580] Error:   [CFS]
    [0xA0042410] (CFS_E_NO_HOST) Register Session returns with error for app [3E3572CA-ED0A-4B8E-BA18-
    2D8642EDDB2B].
55  2025-03-13 15:39:44.815 (-04:00) {635FD72D-1001      :    2} [CLNT:12268: 4580] Error:   [APM]
    [0xA0042410] (CFS_E_NO_HOST) Failed to register session for app "Audacity" on WIN_10_X64. Usage
    tracking: Yes.
56  2025-03-13 15:39:44.815 (-04:00) {635FD72D-1001      :    2} [CLNT:12268: 4580] Error:   [APM]
    [0xA0042410] (CFS_E_NO_HOST) Failed to register application "Audacity"
```

So the above is displaying issues getting a response from the Paging servers (which deliver the appset to the end users Cloudpaging Player (lines `29,31,40, 43, 50 & 52)` which eventually forces the error trying to deliver (lines `54,55,56)`

Cloudpaging Issue:

App stops running or is removed from the player after launching

This can happen if the Cloudpaging session for the app is expired. Common reasons an app session is expired include:

1. AppsAnywhere admin manually expired the session via the Cloudpaging Session settings.

2. An update has been applied by an AppsAnywhere admin, so the application will remove itself from the player, before launching the new version

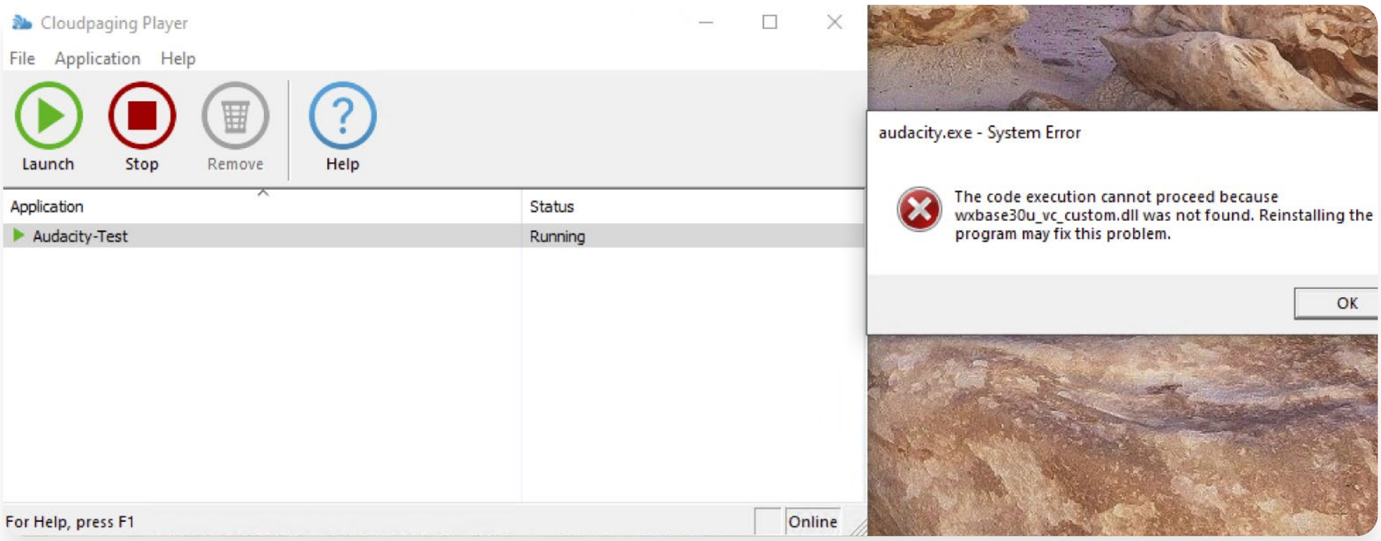
3. There was a technical error renewing the Cloudpaging session.
   - This will happen if there's any issues renewing the Cloudpaging session. Normally, the Cloudpaging Player checks in with the server post launch and every 5 minutes to ensure the user's Cloudpaging session for that app is still active.
   - Documentation on Load Balancer Configuration: **Load Balancer Configuration**

4. The Cloudpaging Player was unable to check in with the Cloudpaging server. Usually this happens when the machine loses internet access, but it can also happen when an app that was allowed offline for a certain number of days has surpassed the number of allowed offline days.

Cloudpaging Issue:

App showing errors while running

If an app launches in the player but does not perform how you would expect, it indicates a packaging issue with that application.



Application	Status
Audacity-Test	Running

For Help, press F1

Online

audacity.exe - System Error

The code execution cannot proceed because wxbase30u_vc_custom.dll was not found. Reinstalling the program may fix this problem.

In this case, the Audacity appset is missing a key file (for e.g. a dll) it needs to run. Common situations like this can include:

- Missing application prerequisites such as c++ redistributables
- Missing or incompatible version of .Net
- Misconfigured service or driver

This must be resolved by investigating and fixing the appset. Helpful information to escalate to a packager who can then compare to a machine where the appset works includes:

- Specific app error message / image
- Operating system and architecture
- Installed C++ Redistributable
- .NET version
- Other installed applications
- Event viewer application logs

ⓘ

Even though the application is a virtualized Cloudpaging application, it is still a Windows application launching and interacting with a Windows desktop device.
So normal troubleshooting and research avenues you would normally take to investigate software issues are still valid.

- Vendor documentation
- Windows updates
- know issues
- Application/Vendor community groups

External Website or Direct Download Issues

Both of these delivery methods work by launching the url set in the delivery method settings in a new browser tab on the user's machine.

Make sure the URL works by launching it outside of AppsAnywhere.

Documentation:

External Website Delivery Method

Direct Download Delivery Method

Secure Download Issues

This delivery method works by having the AppsAnywhere servers host a file. When a user clicks the Launch button, the AppsAnywhere portal delivers the app to them as a download.

Documentation: Secure Download Delivery Method

ⓘ

For Secure Downloads to work the user must be logged into AppsAnywhere and validated by the AppsAnywhere client. As a result this method is only available for MacOS and Windows users.

Troubleshooting Step 1:
Verify the local file path

The first thing to check is to make sure the local file path in the Secure Download delivery method settings is correct. Because the download is hosted on Linux AppsAnywhere servers, the path should always be:

```
/var/www/appsanywhere/secure/<application.msi>
```

where `<application.msi>` is the file you've uploaded to the Secure Downloads file share.

Troubleshooting Step 2:
Verify the file in the Secure Downloads share

This step will look different depending on if your AppsAnywhere environment is self-hosted or cloud hosted.

Self-Hosted
In self-hosted environments, the institution has a file share set up that the AppsAnywhere servers can access. Make sure the file exists in that folder.

Cloud-Hosted
In cloud-hosted environments, we provide access to the Secure Downloads share via the Azure Storage Explorer which you can use to verify the app.

Documentation: Uploading Apps to AppsAnywhere Cloud

Troubleshooting Step 3:
Escalate to Support

If the file still isn't downloading, there may be an issue in the communication between the AppsAnywhere servers and the Secure Downloads share. At this point, it is best to escalate to AppsAnywhere support to identify what the issue is.

Other Delivery Method Launch Issues

If you have any other delivery methods not launching correctly or encounter a situation not outlined in this guide, the best steps are to check the documentation for that delivery method at:

Delivery methods

Afterwards if you cannot solve the issue, proceed to collect client log files and escalate to support.