

# Amazon AppStream 2.0: MATLAB<sup>®</sup> Deployment Guide

Build an Amazon AppStream 2.0 environment to stream [MATLAB](#) to your users.

March 2023

<https://aws.amazon.com/appstream2/>

## Welcome

This guide describes how to deploy and stream the MATLAB desktop application for your company (in this document, ExampleCo) by using [Amazon AppStream 2.0](#), a fully managed, secure application streaming service that runs in the AWS Cloud.

### What you'll accomplish:

- Provision an Amazon virtual private cloud ([Amazon VPC](#)) to provide an isolated virtual network infrastructure within the AWS Cloud. Your AppStream 2.0 resources will use this environment.
- Use the [AWS Management Console](#) to perform the basic administrative tasks required to build an AppStream 2.0 environment. Specifically, you'll:
  1. Install and configure MATLAB for streaming using an image builder.
  2. Provision a fleet of instances to stream your applications. The fleet will use the Graphics Pro instance type and adhere to scaling policies to match the number of users that you want to be able to stream concurrently.
  3. Provision a stack to create a web portal from which users can stream your applications.
  4. Configure persistent storage that users can access across application streaming sessions.

### What you need before starting:

- **An AWS account:** You need an AWS account to use AppStream 2.0 and other AWS services. For information about how to sign up for and activate an AWS account, see *Appendix A*.
- **A current email address:** During the user configuration process for your AppStream 2.0 environment, AWS sends you two emails. You must use these emails to complete the process.
- **Skill level:** You do not need prior experience with AWS to complete these exercises. A basic understanding of desktop computing is helpful but not required.

- **A MATLAB user license.** This license is required to use MATLAB. It must be valid for the MATLAB version intended for use with AppStream 2.0. Contact [MathWorks®](#) or your MATLAB reseller if you need to purchase a license. If you are a new user to MATLAB, a cloud-enabled trial license is [available](#).
- **Online Licensing Enabled.** MATLAB is licensed many different ways. To use MATLAB with AppStream 2.0, online licensing is the preferred solution but only certain MATLAB license types allow this. If you have an Individual license, go [here](#) to enable online licensing. For all other licenses, please contact [MathWorks Support](#) to enable it. More information can be found [here](#).
- **A MathWorks Account associated to the MATLAB license.** In the previous steps, you might have created a new MathWorks Account and/or associated your MathWorks Account with your MATLAB license. If you still do not have a MathWorks Account, you will need to create one and associate it with your license as this is required to use MATLAB with online licensing. You can create a MathWorks Account [here](#). To associate it to your MATLAB user license, go [here](#). If unsuccessful, contact [MathWorks Support](#).
- **MATLAB System Requirements:** The hardware and software requirements needed for smoothly running MATLAB is available on MathWorks [website](#). We recommend using the following instance types:

MATLAB without GPU: stream.compute.large or stream.memory.large. Compute- and Memory-optimized instances are perfectly suited for MATLAB workloads that do not require a GPU. To learn more about the number of vCPUs or memory specifications of these instance families, see [Amazon AppStream 2.0 pricing](#).

MATLAB with GPU and CUDA: stream.graphics.g4dn.xlarge. MATLAB workloads that need GPU acceleration for use with [Parallel Computing Toolbox™](#) can be run on Graphics G4 instances. Powered by NVIDIA T4 GPUs, there are six Graphics G4 instances types starting from 4 vCPUs, 16 GiB system memory, and 16 GiB graphics memory, to 64 vCPUs, 256 GiB system memory, and 16 GiB graphics memory. To learn more about Graphics Pro instances, see [Amazon AppStream 2.0 pricing](#).

- **End user client recommendations:** To use MATLAB delivered through AppStream 2.0, your user will need a modern HTML5 browser such as Google Chrome, Mozilla Firefox, Microsoft Edge or Internet Explorer 11+.
- **End user network recommendations:** AppStream 2.0 uses an adaptive streaming protocol (NICE DCV) to deliver an interactive streaming session to users. The protocol encodes pixels on a remote host, securely transmits them over the network, and renders them on a client device. It also accepts user keyboard and mouse input, enables file transfer between client and remote host, and provides clipboard support to provide an interactive experience for a user when using streamed applications. While the streaming protocol adapts to changes on the screen and only transmits pixels when required, it will use the available bandwidth on the network. Also, since the streaming session is interactive, and the application on the remote host needs to respond to user inputs on a client device, the round-trip latency will influence the responsiveness that a user will experience.

The amount of bandwidth used when transmitting pixels is proportional to the changes on the screen and the resolution of the display monitor(s) used by the client device. The changes on the screen and the resolution are determined by the type of application (3D versus business application) and usage pattern (switching between windows and menus quickly). A 3D application may require a high-resolution monitor and trigger large changes to the screen when a user is interacting with complex hi-fidelity models. To transmit these changes on the screen quickly and provide a responsive experience to the user, the protocol will use a large amount of bandwidth momentarily. On the other hand, a business application may only involve text input. While changes to text on screen can be transmitted with very small amount of bandwidth, switching quickly between windows or menus within even a text-based application will result in large changes to the screen and hence drive momentary increases in bandwidth used. The round-trip network latency influences the responsiveness that a user experiences when entering input and viewing changes on the screen. While other factors such as quality of network, client device performance, and remote host instance selection can also influence the responsiveness, latency should be considered as one of the primary factors. In general, lower latency connections will deliver more responsive and performant streaming experience. Below are the recommendations for sample MATLAB use cases.

Use case	Recommended bandwidth available per user	Recommended maximum roundtrip latency
<b>MATLAB without GPU</b>	1-2 Mbps	< 150 ms
<b>MATLAB with GPU – Streaming with low fidelity datasets with 2K monitors</b>	5-6 Mbps	< 100 ms
<b>MATLAB with GPU – Streaming high fidelity datasets with 4K monitors</b>	10-12 Mbps	< 50 ms

# Contents

Welcome .....	2
Step 1. Sign in to the AWS Management Console and select an AWS Region .....	7
Step 2: Create network resources .....	7
Step 3: Create an AppStream 2.0 image builder .....	10
Step 4: Connect to the image builder and install applications.....	12
Connect to the image builder instance .....	13
Install MATLAB.....	13
Configure MATLAB to use online licensing mode.....	14
Step 5: Use Image Assistant to create an AppStream 2.0 image .....	15
Create your AppStream 2.0 application catalog.....	15
Disable Internet Explorer Enhanced Security Configuration .....	17
Test your applications by using a local user account .....	18
Optimize the launch performance of your applications .....	18
Configure the image .....	19
Finish creating the image.....	19
Step 6: Provision a fleet .....	20
Provide fleet details .....	20
Choose an image.....	21
Configure the fleet.....	21
Configure the network .....	22
Step 7: Create an AppStream 2.0 stack and a streaming URL .....	23
Provide stack details and associate the stack with a fleet .....	23
Enable persistent storage for the stack .....	24
Create a streaming URL .....	24
Step 8: Manage user access with SAML 2.0.....	26
Step 9: Take the next step with AppStream 2.0.....	27
Contributors .....	28
Document revisions .....	28
Create your AWS account .....	29
Add a payment method.....	29
Verify your phone number .....	30
Choose an AWS Support plan .....	30
Watch for three AWS account confirmation emails .....	30
Appendix B. Clean up your AppStream 2.0 resources.....	32
Stop and delete your image builder .....	32
Disassociate your fleets from your stack and delete your stack .....	32
Stop and delete your fleet.....	32

## Step 1. Sign in to the AWS Management Console and select an AWS Region

If you do not have an AWS account, you must first complete the steps in *Appendix A*.

1. Sign in to the AppStream 2.0 console at <http://console.aws.amazon.com/appstream2>.
2. Type your email address or your AWS account ID, and choose **Next**.
3. Type your AWS account password, and choose **Sign In**.
4. In the menu in the upper right corner of the console, select the AWS Region for your environment. AWS currently hosts services in 23 different geographical areas.



5. Select one of the following 12 Regions in which AppStream 2.0 is available:

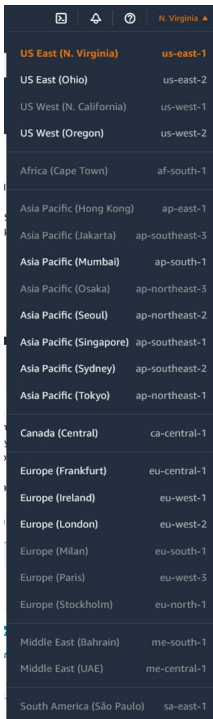


Figure 1: Available AWS Regions for AppStream 2.0.

## Step 2: Create network resources

In this section, you will create an Amazon virtual private cloud (VPC) and other network resources required for your AppStream 2.0 environment. The following steps use an

AWS CloudFormation template to automatically create and configure the necessary network resources.

1. Make sure that you are signed in to the [AWS Management Console](#).
2. From the choices below, open the CloudFormation template link associated with the AWS Region in which you want to build your AppStream 2.0 environment.
  - [US East \(N. Virginia\)](#)
  - [US East \(Ohio\)](#)
  - [US West \(Oregon\)](#)
  - [EU \(Frankfurt\)](#)
  - [EU \(Ireland\)](#)
  - [EU \(London\)](#)
  - [Asia Pacific \(Mumbai\)](#)
  - [Asia Pacific \(Seoul\)](#)
  - [Asia Pacific \(Singapore\)](#)
  - [Asia Pacific \(Sydney\)](#)
  - [Asia Pacific \(Tokyo\)](#)
  - [Canada \(Central\)](#)

The AWS CloudFormation console displays the URL of a template that is used to create your network resources and the name of the resulting AWS CloudFormation stack.

3. In the bottom right corner of the window, choose **Create**. AWS CloudFormation starts creating the resources and displays a status message to indicate progress.



**Quick create stack**

**Template**

Template URL  
<https://s3.amazonaws.com/appstream-demo-vpc-cfn-template/AppStreamDemoVPCtemplate.template>

Stack description  
 -

**Stack name**

Stack name

Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

**Parameters**

Parameters are defined in your template and allow you to input custom values when you create or update a stack.

No parameters  
 There are no parameters defined in your template

Cancel

Figure 2: Using a template in AWS CloudFormation to create network resources.

4. When the creation process completes, usually within five minutes, the AWS CloudFormation console displays the status **CREATE\_COMPLETE**.
5. Navigate to the Amazon VPC console at <https://console.aws.amazon.com/vpc/>.
6. In the navigation pane, under **Virtual Private Cloud**, choose **Your VPCs**. In the list of VPCs, you should see the following VPC that was automatically created:

<input type="checkbox"/>	Name	VPC ID	State	IPv4 CIDR
<input type="checkbox"/>	AppStream2 VPC	vpc-d6ff74ad	available	10.0.0.0/20

Figure 3: VPC created by AWS CloudFormation.

**Note:** The VPC ID value will differ for your VPC.

7. In the navigation pane, under **Virtual Private Cloud**, choose **Subnets**. In the list of subnets, you should see the following subnets that were automatically created:

Name	Subnet ID	State	VPC	IPv4 CIDR
AppStream2 Public Subnet	subnet-1a7dbb50	available	vpc-d6ff74ad   AppStream2 VPC	10.0.0.0/24
AppStream2 Private Subnet2	subnet-fd405299	available	vpc-d6ff74ad   AppStream2 VPC	10.0.2.0/24
AppStream2 Private Subnet1	subnet-c175b38b	available	vpc-d6ff74ad   AppStream2 VPC	10.0.1.0/24

Figure 4: Subnets created by AWS CloudFormation.

**Note:** The Subnet ID and VPC values will differ for your subnets.

- You have now successfully created your network resources by using AWS CloudFormation. You can proceed to Step 3.

## Step 3: Create an AppStream 2.0 image builder

AppStream 2.0 uses EC2 instances to stream applications. You launch instances, called *image builders*, from base images that AppStream 2.0 provides. To create your own custom image, you connect to an image builder instance, install and configure your applications for streaming, and then create your image by creating a snapshot of the image builder instance.

To install and configure applications to stream to your users, you must create an image builder instance as described in the following procedure.

### Launch an image builder to install applications

- Open the AppStream 2.0 console at <https://console.aws.amazon.com/appstream2>.
- If you have not previously configured any AppStream 2.0 settings, the following page appears:

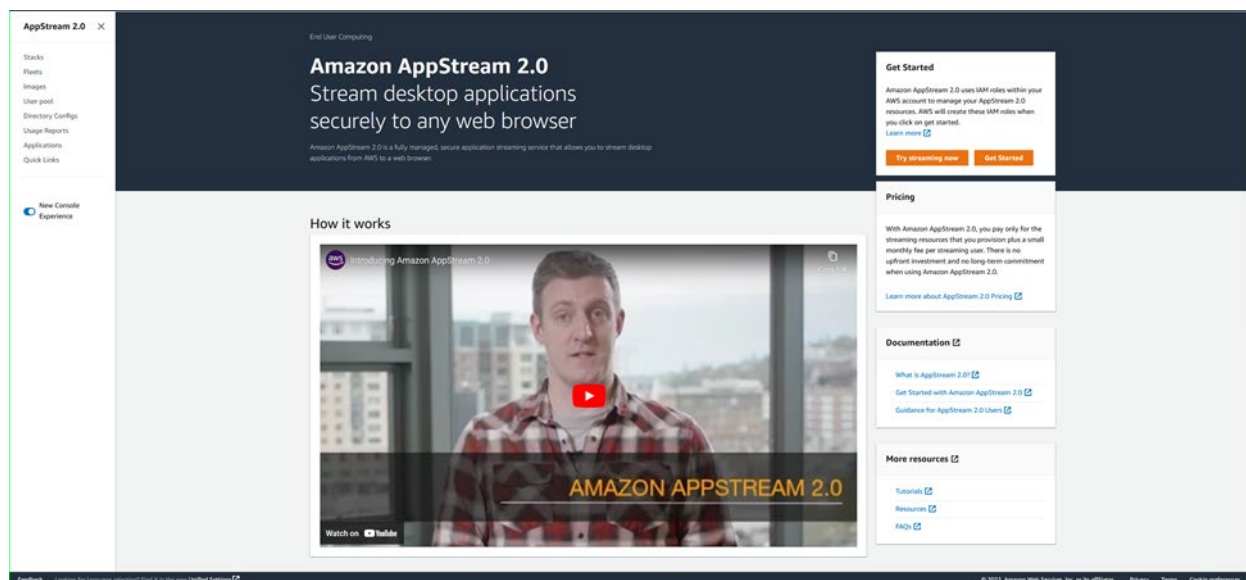


Figure 5: The AppStream 2.0 first experience page.

**Note:** If the AppStream 2.0 navigation page appears instead, skip to step 5.

3. Choose **Get started**.
4. In the lower right corner of the page, choose **Skip** (this guide walks you through a different process for getting started with AppStream 2.0).

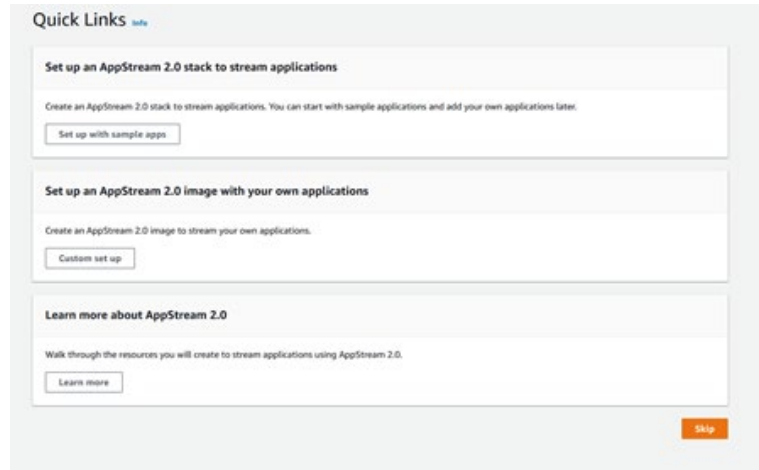


Figure 6: AppStream 2.0 getting started options.

5. In the navigation pane, choose **Images, Image Builder, Launch Image Builder**.
6. In the **Step 1: Choose Image** window, in the list of images, select the image with the name *AppStream-Graphics-G4dn-WinServer2019-mm-dd-yyyy*, where *mm-dd-yyyy* represents the most recent date [for example, *AppStream-Graphics-G4dn-WinServer2019-10-05-2022 (Public)*]. Base images include the latest updates to Microsoft Windows and the AppStream 2.0 agent software. You use this base image to create a custom image that includes your own applications.
7. At the bottom of the page, choose **Next**.
8. In **Step 2: Configure Image Builder**, the following image builder configuration options are displayed:
9. Type the following information and then choose **Review**.

**Configure image builder fields**

Option	Value
<b>Name</b>	Provide a unique name identifier for the image builder, such as ExampleCo_Image_v1_mmddyyyy, using any of the following characters: a-Z,0-9,-,_,.
<b>Display Name</b>	Provide an optional name, such as ExampleCo Image v1 March 2023, to be displayed in the console for easier reference and readability.
<b>Instance Type</b>	Provide an optional name, such as ExampleCo Image v1 March 2023, to be displayed in the console for easier reference and readability.

Choose **Next** to continue to **Step 3: Configure Network** and then type the following information.

Option	Value
<b>Default Internet Access</b>	Make sure that this option is not selected.
<b>VPC</b>	Select the option corresponding to <b>AppStream 2 VPC</b> .
<b>Subnet</b>	Select the subnet with the IP address range 10.0.1.0/24 ( <b>AppStream2 Private Subnet1</b> ).
<b>Security group(s)</b>	Accept the default security group listed.
<b>Active Directory Domain (Optional)</b>	Do <b>not</b> configure any options.

10. Choose **Review**, and confirm the details for the image builder. To change the configuration for any section, choose **Edit** and make your changes.
11. After you finish reviewing the configuration details, choose **Launch**. If an error message notifies you that you don't have sufficient quotas to create the image builder, submit a quota increase request through the AWS Support Center. For more information, see [AWS service quotas](#).
12. The image builder creation process takes about 15 minutes to complete. During this process, the status of the image builder displays as **Pending** while AppStream 2.0 provisions the necessary resources.
13. Click the **Refresh** icon periodically to update the image builder status. After the status changes to **Running**, the image builder is ready to use and you can create a custom image.

**Note:** Charges accrue for an image builder instance while it is running, even if no user is actively connected. You can stop or delete the image builder at any time. No user fees are incurred when users connect to an image builder. For more information, see [AppStream 2.0 Pricing](#).

## Step 4: Connect to the image builder and install applications

Now that you have launched an image builder, you can use it to install and configure the applications to stream to users. First, you must establish a streaming connection to the instance to install and configure your applications.

## Connect to the image builder instance

1. Open the AppStream 2.0 console at <https://console.aws.amazon.com/appstream2>.
2. In the navigation pane, choose **Images, Image Builder**.
3. Select the image builder instance that you created earlier (*ExampleCo\_Image1\_mmdyyyy*). Verify that its status is **Running** and choose **Connect**.

**Note:** If the status is **Stopped**, select the instance, and choose **Actions, Start**. Click the **Refresh** icon periodically to update the instance list until the status is **Running**.

4. The new browser tab opens, displaying options for logging into the image builder instance. Choose the **Local User** tab, then choose **Administrator**.

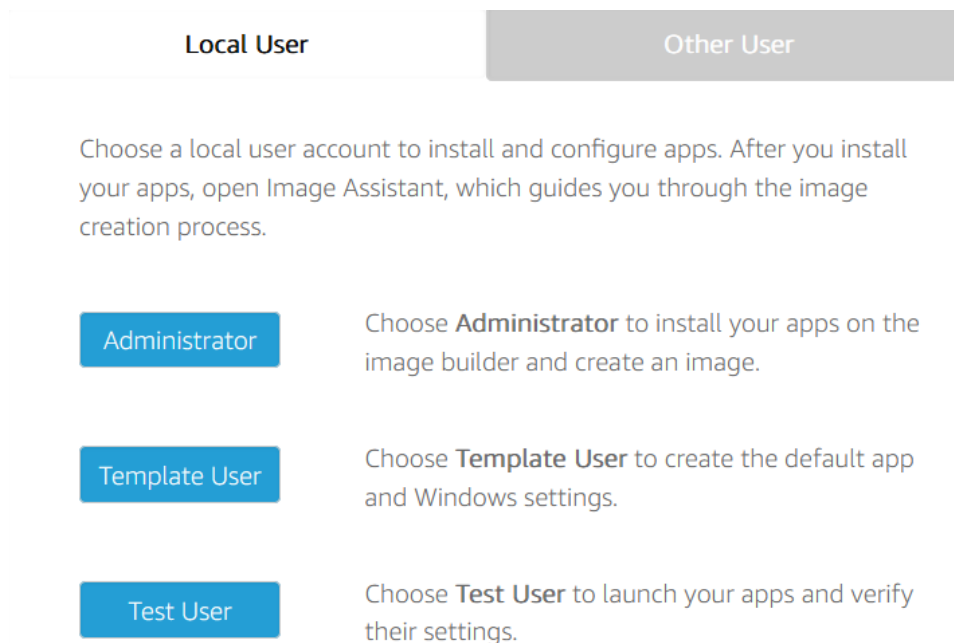


Figure 7: The image builder instance login options window.

**Note:** If a new browser tab does not open, configure your browser to allow pop-ups from <https://console.aws.amazon.com/>.

5. After a few moments, you are connected to the image builder instance with administrator rights.

## Install MATLAB

1. Launch Firefox and browse to <https://www.mathworks.com/downloads/>. Choose **Download R2022b** or any version after R2017b. The following steps correspond

to R2022b. The process may differ for other versions. On the next page, choose **Windows (64-bit)**.

2. After the download is complete, launch the downloaded executable. This process will extract the installation files to a known file location in image builder.
3. Once the extraction is complete, the installation process will start automatically.
4. In the prompted dialog, choose **Log in with a MathWorks Account** for the installation method choice and choose **Next**.
5. In the next screen, provide your MathWorks Account login credentials (username and password) and choose **Next**. If you are successfully authenticated, you will proceed to the next step in installation.
6. In the next screen, choose **Install** and click **Next**.
7. Choose one of your licenses in the next screen and choose **Next**.
8. Leave the default installation folder path unchanged and choose **Next**.
9. Select the MATLAB products that you want to install and choose **Next**.
10. Leave the options unchanged in the next screen and choose **Next, Install**.
11. In the final screen after installation, uncheck the option to activate MATLAB and choose **Finish**.

### ***Configure MATLAB to use online licensing mode***

1. Transfer the license configuration file linked [here](#) to a known file location on your local device.
2. Choose the folder icon in the streaming session toolbar to launch the **My Files** web dialog. Select the **Temporary Files** folder.
3. Choose the **Upload Files** option to upload license\_info.xml to the image builder.
4. Once the file is uploaded, copy it to the licenses folder under the MATLAB folder. For R2022b, the location is **C:\Program Files\MATLAB\R2022b\licenses**. If no such folder exists, you can manually create one and copy the xml file to the folder.
5. This change ensures that MATLAB always uses the online licensing mode to authorize use.

6. Launch MATLAB, sign in with your MathWorks Account credentials and verify that the application launches successfully.

## Step 5: Use Image Assistant to create an AppStream 2.0 image

At this point, you have launched an image builder instance and installed MATLAB on the image builder. Now you'll prepare the application for streaming, optimize it for streaming performance, and create your image.

In this section, you'll do the following:

- Create an application catalog by using Image Assistant.
- Disable the Internet Explorer Enhanced Security Protection feature.
- Test the application by using a local user account that has the same permissions that end users will have in their streaming sessions.
- Optimize the application's launch performance.
- Configure the image.
- Finish creating the image.

### *Create your AppStream 2.0 application catalog*

The process of creating an AppStream 2.0 application catalog includes specifying the name, display name, executable file to launch, and icon to display for each application that you plan to stream.

1. From the image builder desktop, open Image Assistant.
2. On the **ADD APPS** tab, choose **Add App**.

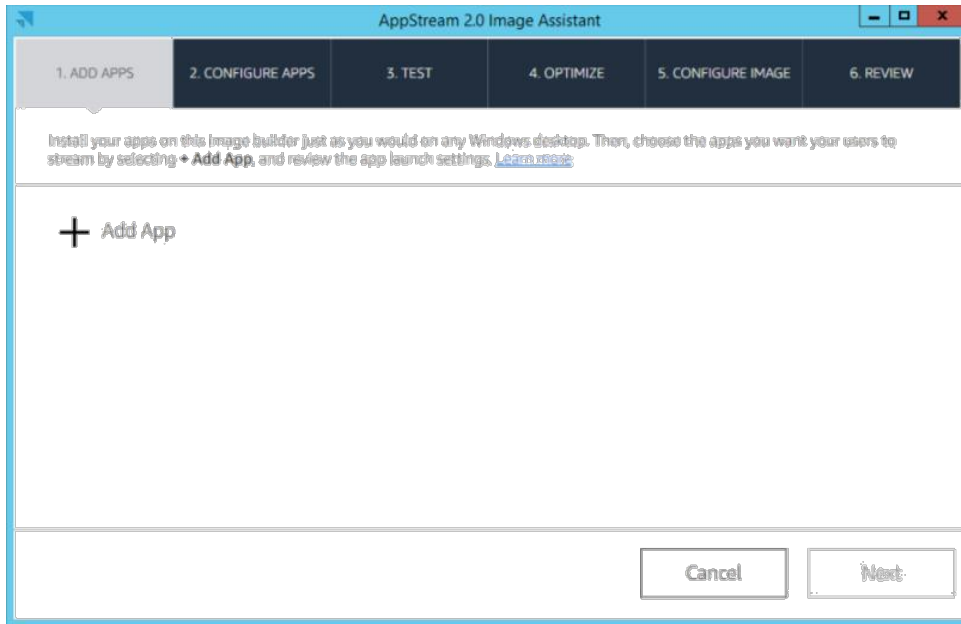


Figure 8: The *Add Applications to Image* dialog box in Image Assistant.

3. Navigate to the location of the MATLAB executable (for example, **C:\Program Files\MATLAB\R2022b\bin**), select the application executable, and then choose **Open**.
4. In the **App Launch Settings** window, type the following information and choose **Save**.

Option	Value
<b>Name</b>	Name of the application executable. Leave the default value.
<b>Display Name</b>	The name of the application that is displayed to end users. Type <b>MATLAB</b> .
<b>Launch Path</b>	The location of your application executable file. Leave the default value.
<b>Icon Path</b>	Leave the default value of C:\ProgramData\Amazon\Photon\AppCatalogHelper...
<b>Launch Parameters</b>	-useStartupFolderPref "C:\Users\PhotonUser\My Files"
<b>Working Directory</b>	Leave this blank.

5. Now that you have added MATLAB to your catalog, choose **Next**.



## Disable Internet Explorer Enhanced Security Configuration

Applications use Internet Explorer (IE) to open http links embedded in the applications. When you launch one of these links, IE displays a warning message for every webpage that it opens. This behavior is due to Internet Explorer Enhanced Security Configuration, a security setting of IE that blocks access to web content and application scripts for security reasons. This setting may prevent MATLAB from functioning properly. We can safely disable this feature to proceed further. To disable this feature for AppStream 2.0 users, do the following:

1. Connect to your image builder as **Administrator**.
2. Open **Server Manager** from the Windows Start menu.
3. In the left pane, choose **Local Server**.
4. Next to **IE Enhanced Security Configuration**, choose **On**.
5. Select the **Off** option for both Administrators and Users.
6. Choose **OK**.

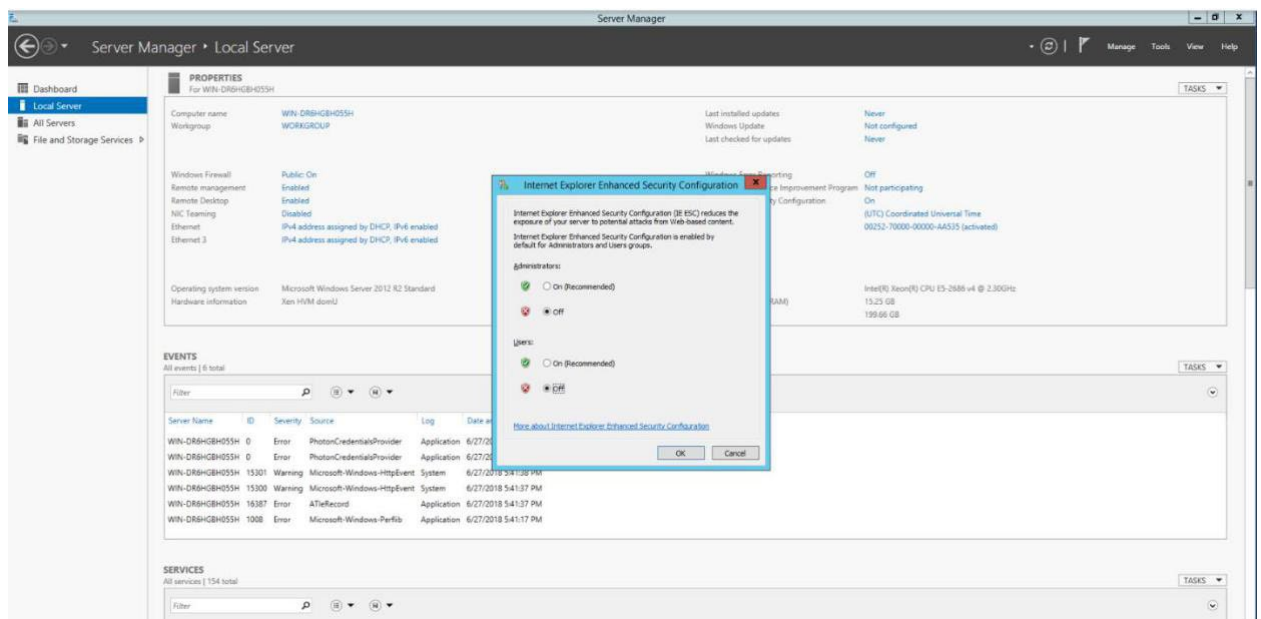


Figure 9: IE ESC - Server Manager in Image Builder

10. On the AppStream 2.0 toolbar, choose **Admin Commands**, then choose **Switch User**.
11. Choose **Template User**.
12. Once you are logged into the Template User account, launch **Internet Explorer**.
12. In the upper right corner of Internet Explorer, choose the **Settings** (gear) icon, then **Internet Options**.
13. In the **Internet Options** window, choose the **Advanced** tab.

14. Choose the **Reset** button. Choose **Reset** again in the prompted dialog. Close Internet Explorer.
15. Switch to **Administrator** account. Launch **Image Assistant**. Choose **Next** to proceed to the Configure step. From this tab, click **Save settings**. This will save the template user settings as default user settings.
16. Switch to Test User. Launch Internet Explorer. Confirm that the message “Internet Explorer Enhanced Security configuration is not enabled” is displayed in the home page. Browse to any website to confirm that IE is not displaying any blocking prompts.

### ***Test your applications by using a local user account***

An image builder includes a test user account that enables you to test your applications by using the same policies and permissions as your users. Follow these steps to confirm that your applications open correctly.

1. In the **Test** tab, choose **Switch User, Test User**. You are now logged into the same Windows Server as a local user who has regular (non-administrative) user rights.
2. Open Image Assistant. In **Test Applications**, the MATLAB application that you added is displayed.
3. Choose the application to open it. Sign in with your MathWorks Account credentials to launch the application.
4. If you get a license error, contact your administrator.
5. After successful authentication, wait for the application to launch fully. After validating the launch, sign out from the application and close the application window.
7. Choose **Switch User**.
8. On the **Local User** tab, choose **Administrator**.
9. On the Image Assistant **Test** tab, choose **Next**.

### ***Optimize the launch performance of your applications***

During this step, Image Assistant opens your application, identifies their launch dependencies, and performs optimizations to improve performance.

1. On the **Optimize** tab, choose **MATLAB, Launch**.
2. Wait for MATLAB to completely start.

- After you complete the first run experience for the application and verify that it functions as expected, choose **Continue**.

## Configure the image

- On the **Configure Image** tab, type the following information.

Option	Value
<b>Name</b>	The unique name identifier for the image, such as <i>ExampleCo_Image_MATLAB_v1_mmddyyyy</i> , using any of the following characters: a-Z,0-9,-,_,.,. Note: The name cannot begin with "Amazon," "AWS," or "AppStream."
<b>Display Name</b>	A user-friendly name to display in the console
<b>Description</b>	An optional description for the image: for example, <i>Image v1 created by (your initials or name) on mm/dd/20yy</i> .
<b>Always use latest agent version</b>	Leave this check box selected so that streaming instances that are launched from your image always include the latest AppStream 2.0 features, performance improvements, and security updates. For more information, see <a href="#">Amazon AppStream 2.0 Agent Release Notes</a> .

## Finish creating the image

Complete the following steps to disconnect from the remote session and start the image creation process.

- Review the image details, and choose **Disconnect and Create Image**.
- The remote session disconnects within a few moments. When the **Lost Connectivity** message appears, close the browser tab.

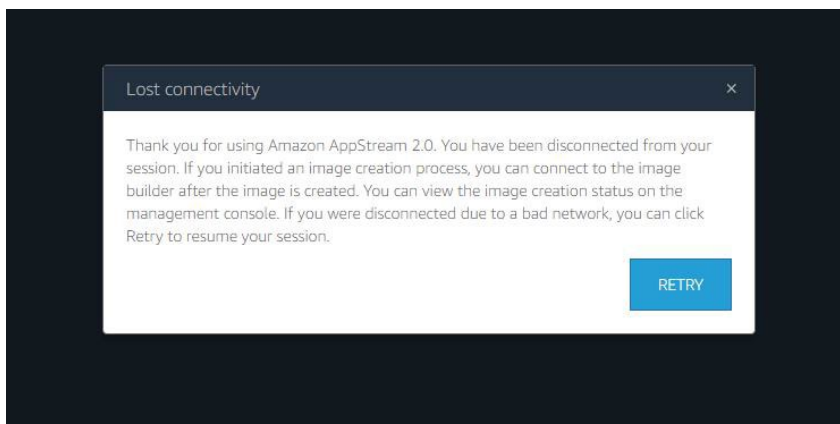


Figure 10: The **Lost connectivity** message indicating that the image creation process has started.

- Return to the [Amazon AppStream 2.0 console](#) and choose **Images, Image Registry**. While your image is being created, the image status in the image registry of the console appears as **Pending**. While your image is being created, you cannot connect to it.
- Click the **Refresh** icon periodically to update the status. Image creation takes about 20 minutes. After your image is created, the image status changes to **Available** and the image builder is automatically stopped.

**Note:** To make changes to your image, such as adding other applications or updating existing applications, you must create a new image. To do so, restart and reconnect to the image builder, make your changes, and then repeat the Image Assistant process to create a new image that includes the changes.

## Step 6: Provision a fleet

An AppStream 2.0 fleet defines the hardware, network, Active Directory (if applicable), and scaling configuration for your application streaming infrastructure. For more information, see [Amazon AppStream 2.0 Stacks and Fleets](#).

In this section, you'll do the following:

- Provide details for the fleet.
- Choose an image.
- Configure the fleet.
- Configure the network.

### Provide fleet details

- Open the AppStream 2.0 console at <https://console.aws.amazon.com/appstream2>.
- In the navigation pane, choose **Fleets, Create Fleet**.
- For **Step 1: Provide Fleet Details**, type the following text and choose **Next**.

Option	Value
<b>Name</b>	The unique name identifier for the fleet, such as <i>ExampleCo_Fleet_v1_mmddyyyy</i> , using any of the following characters: a-Z,0-9,-,_,. Note: The name cannot begin with "Amazon," "AWS," or "AppStream."
<b>Display Name</b>	The name displayed in the console, such as <i>ExampleCo Fleet v1 February 2023</i> .
<b>Description</b>	An optional description for the fleet. For example, <i>Fleet v1 created by &lt;your initials or name&gt; on mm/dd/20yy</i> .

## Choose an image

For **Step 2: Choose an image**, choose the image that you created, scroll to the bottom of the page, and then choose **Next**.

## Configure the fleet

1. For **Step 3: Configure fleet**, in **Choose instance type**, you define the hardware configuration for each of the instances that make up your fleet. Because you created the image by using the Graphics G4 family, the instance type is already populated. However, you can select any of the six instance type options that are presented.
2. For this exercise, select **stream.graphics.g4dn.xlarge**. For more information, see [Amazon AppStream 2.0 Instance Families](#).
3. Under **Fleet Type details**, choose a fleet type that suits your needs. The fleet type determines the availability of streaming instances and affects your costs. You can choose either of the following:
  - **Always-on**: Instances run all the time, even when no users are streaming applications. When this option is selected, instances are immediately available for the next user to connect to immediately.
  - **On-Demand**: Instances run only when users are streaming applications. Idle instances that are available for streaming are in a stopped state. When this option is selected, a user must wait for one to two minutes for an instance to start up.

For this exercise, select the **On-Demand** option.

4. Under **User session details**, define the maximum amount of time that users can be connected to streaming sessions and how long streaming sessions should remain active after users disconnect.
  - **Maximum session duration** defines how long user streaming sessions can remain active. If users are still connected to a streaming session five minutes before this limit is reached, they are prompted to save any open documents before being disconnected. Choose **8 hours**.
  - **Disconnect timeout** defines how long user streaming sessions can remain active after users are disconnected. If users try to reconnect to the streaming session after a disconnection or network interruption within this time interval, they are connected

to the previous session. After the disconnect timeout expires, the session is terminated, and the user must start a new session to reconnect. Leave the default setting of **15 minutes**.

5. Under **Fleet capacity**, set **Minimum capacity** to 2 and **Maximum Capacity** to 4.

**Notes:**

Capacity is defined in terms of the number of instances within a fleet and, consequently, every unique user streaming session that is served by a separate instance.

The minimum capacity for your fleet is the minimum number of users who are expected to be streaming at the same time.

The maximum capacity for your fleet is the maximum number of users who are expected to be streaming at the same time.

6. Choose **Next**.

### **Configure the network**

1. For **Step 4: Configure Network**, make sure that the **Default Internet Access** check box is not selected. This option does not need to be selected because you already configured a VPC with a NAT gateway to provide internet access.
2. For **VPC**, select **vpc-xxxxxxx (AppStream2 VPC)**.
3. For **Subnet 1**, choose **subnet-xxxxxxx | (10.0.1.0/24)**. This is the AppStream2 Private Subnet1.
5. For **Subnet 2**, choose **subnet-xxxxxxx | (10.0.2.0/24)**. This is the AppStream2 Private Subnet2.
6. Choose **Next**.
7. Confirm the fleet configuration details. To change settings for any section, choose **Edit**, and make the needed changes. After you finish reviewing the configuration details, choose **Create**.
8. In the pricing acknowledgement dialog box, select the acknowledgement check box, and choose **Create** to begin provisioning your fleet with the initial set of running instances.

Create AppStream 2.0 Fleet [On-Demand] ✕

You will be charged the streaming instance fees when users are connected, and a small hourly fee for each instance in the fleet that is not streaming apps. You will also be charged for monthly user fees for any users who connect and stream their applications in a month. Charges for streaming instances and user fees are not eligible for the AWS Free Tier. [Learn more](#)

I acknowledge that I have read the pricing details and want to continue.

Figure 11: The AppStream 2.0 streaming instance pricing acknowledgement dialog box.

**Note:** If an error message notifies you that you don't have sufficient limits to create the fleet, submit a limit increase request to the AWS Support Center. For more information, see [Amazon AppStream 2.0 Service Limits](#).

Fleet provisioning usually takes 20 minutes to finish. While your fleet is being created and fleet instances are provisioned, the status of your fleet displays as **Starting** in the **Fleets** list. Choose the **Refresh** icon periodically to update the fleet status until the status is **Running**.

8. After the status changes to **Running**, the fleet is available and you can use it to create a stack.

## Step 7: Create an AppStream 2.0 stack and a streaming URL

An AppStream 2.0 stack consists of a fleet, user access policies, and storage configurations. You create a stack to start streaming applications to users.

In this section, you'll do the following:

- Provide details for your stack and associate your stack with a fleet.
- Enable persistent storage for the stack.
- Create a streaming URL.

### Provide stack details and associate the stack with a fleet

1. Open the AppStream 2.0 console at <https://console.aws.amazon.com/appstream2>.
2. In the navigation pane, choose **Stacks**, **Create Stack**.
3. For **Step 1: Stack Details**, type the following information and choose **Next**.

Option	Value
<b>Name</b>	The unique name identifier for the stack, such as <i>ExampleCo_Stack_mmddyyyy</i> , using any of the following characters: a-Z,0-9,-,_. Note: The name cannot begin with "Amazon," "AWS," or "AppStream."
<b>Display name</b>	A friendly name for the stack, such as <i>ExampleCo Stack mmddyyyy</i> .
<b>Description</b>	Leave this field blank.

<b>Redirect URL</b>	Leave this field blank.
<b>Feedback URL</b>	Leave this field blank.
<b>Fleet</b>	Select the <i>ExampleCo_Fleet_v1_mmddyyy</i> fleet that you created.

### **Enable persistent storage for the stack**

1. For **Step 2: Enable Storage**, make sure that the **Enable Home Folders** option is selected. When this option is selected for an AppStream 2.0 stack, users of the stack are presented with a persistent storage folder in their AppStream 2.0 sessions. Data stored by users in their Home Folders is backed up to an Amazon S3 bucket that is automatically created in your AWS account. You can also enable *Google Drive for Google Workspace* or *OneDrive for Business* as user storage options if you use of these storage providers. For more information, see [Persistent Storage with AppStream 2.0](#).

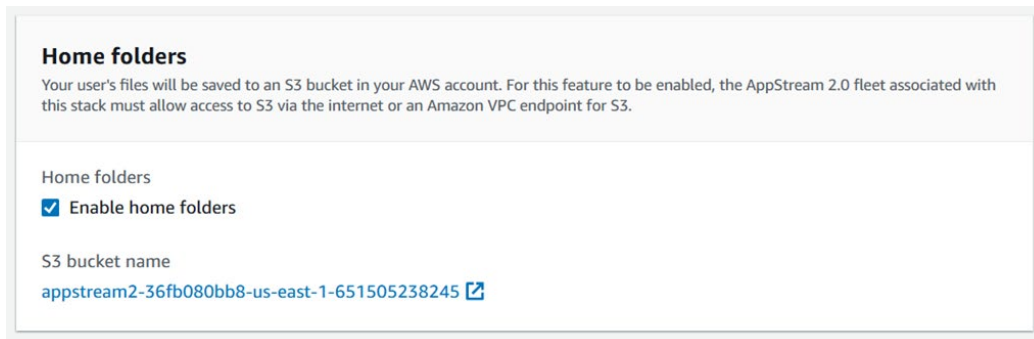


Figure 12: The **Enable Home Folders** page, displaying the Amazon S3 bucket that is automatically created.

2. **Disable application settings persistence.** Make sure to uncheck the box for Enable application settings persistence. Choose **Next**.
3. Confirm the stack configuration details on the **Review and Create** page. After you finish reviewing the configuration details, choose **Create**.

After a few moments, the **Stacks** list reappears. Your stack is listed with a status of **Active**.

### **Create a streaming URL**

To quickly test application streaming without setting up users, create a temporary URL that can be pasted into a new browser window.

1. In the navigation pane, choose **Stacks**.
2. For **Stacks**, select the stack that you just created.
3. Choose **Actions, Create streaming URL**.



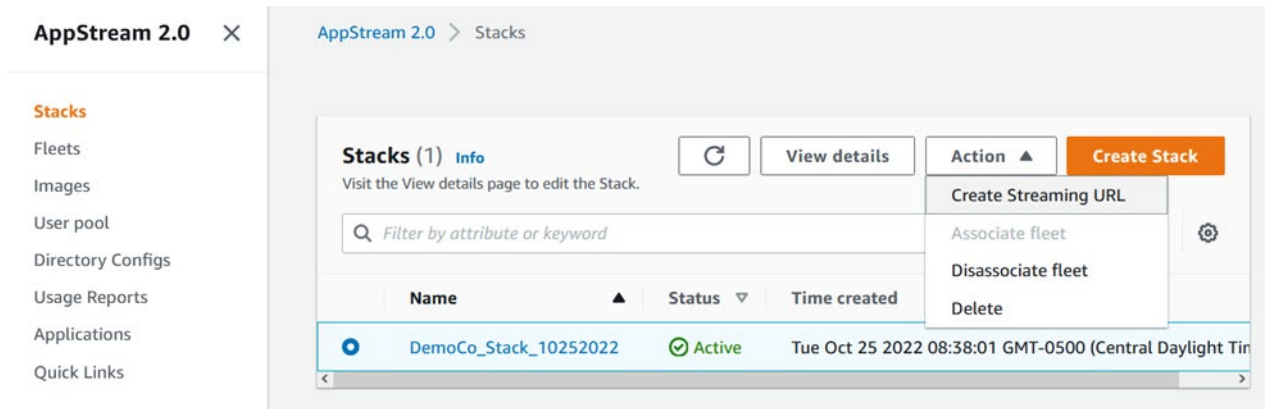


Figure 13: The **Create streaming URL** menu item.

4. In the **Create streaming URL** dialog box, type the following information and choose **Get URL**.

Option	Value
<b>User ID</b>	An ID for the user of the streaming URL. For this exercise, type <b>ExampleCoTestUser1</b> .  Note: This entry is not tied to Active Directory or an LDAP type of directory service. It is simply an identifier for creating a unique URL.
<b>Session Expiration</b>	The length of time that this URL is available to use. For this exercise, choose <b>1 hour</b> .

5. The **Create streaming URL** dialog box refreshes, displaying the user ID that you entered and the URL that AppStream 2.0 generated for the user.
6. Choose **Copy Link** to copy the full URL to the clipboard.

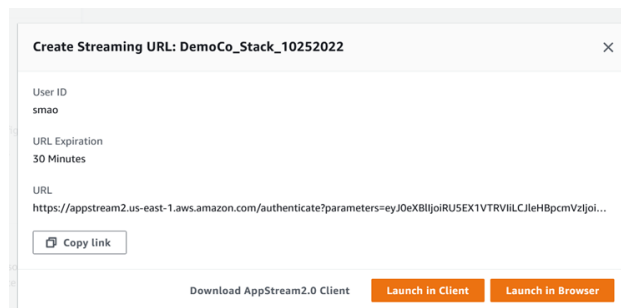


Figure 14: The **Create streaming URL** dialog box.

- The **Create streaming URL** dialog box refreshes again, confirming that the link was successfully copied to the clipboard. Close the **Create Streaming URL** dialog box.

**Notes:**

- Providing access to an AppStream 2.0 streaming session through a console-generated link as described in this procedure is for testing only.
  - In a production environment, several authentication and authorization options are available to provide your users with access to AppStream 2.0. These options include federation through SAML 2.0, the AppStream 2.0 user pool (a built-in identity management feature), and custom identity solutions. For more information, see [Setting up SAML](#) and [Manage Access with the AppStream 2.0 User Pool](#).
- In a browser, open a new tab, paste the streaming URL into the address bar, and press **Enter**. AppStream 2.0 displays an application catalog page that lists the applications that you have configured for streaming.
  - Choose the MATLAB icon to stream the application, and confirm that it functions as expected.

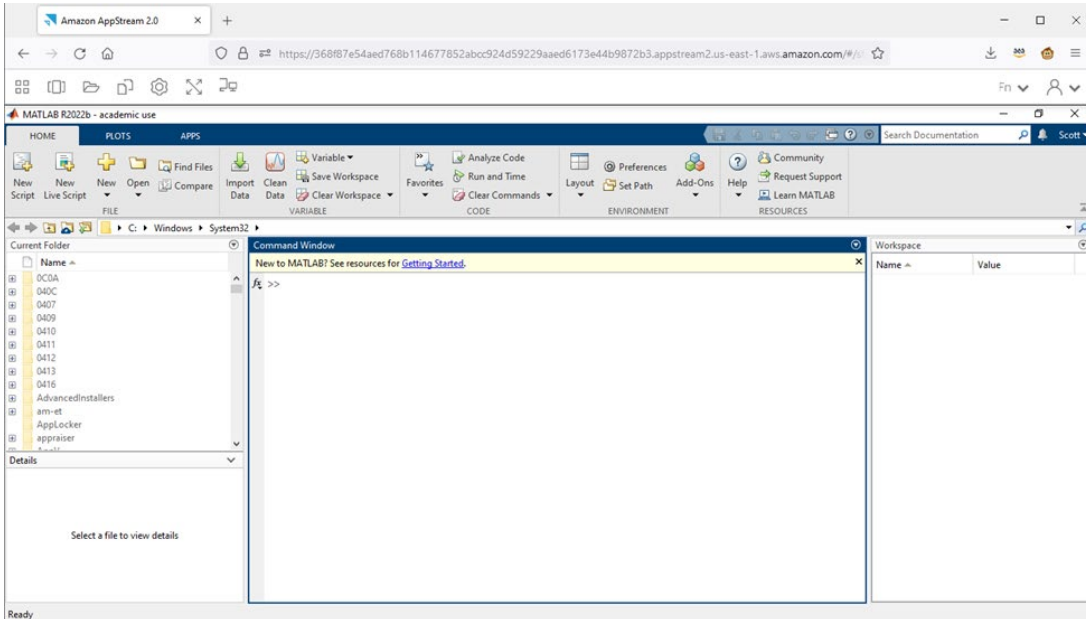


Figure 15: Streaming experience

## Step 8: Manage user access with SAML 2.0

Amazon AppStream 2.0 supports identity federation to AppStream 2.0 stacks through Security Assertion Markup Language 2.0 (SAML 2.0). You can use an identity provider (IdP) that supports SAML 2.0, such as Azure AD, Google Workspace, or Okta, to provide an onboarding flow for your AppStream 2.0 users.

This feature offers your users the convenience of one-click access to their AppStream 2.0 applications using their existing identity credentials. You also have the security benefit of identity authentication by your IdP. By using your IdP, you can control which users have access to a particular AppStream 2.0 stack. For more information, see [Amazon AppStream 2.0 Integration with SAML 2.0](#).

## Step 9: Take the next step with AppStream 2.0

Congratulations, you have now successfully created an AppStream 2.0 environment to stream applications. This guide provided an introduction to AppStream 2.0 by walking you through basic configuration and deployment exercises for MATLAB. To increase your understanding of AppStream 2.0 and take advantage of more features, consider doing the following:

1. Try using different instance types and sizes to match your application's requirements. For information about the different instance types and sizes available for AppStream 2.0, and their pricing, see [Amazon AppStream 2.0 Pricing](#).
2. Join your AppStream 2.0 fleets and image builders to domains in Microsoft Active Directory. Your users can then benefit from access to Active Directory network resources such as printers and file shares from within their streaming sessions. You can also apply Group Policy settings to your streaming instances and users to meet the needs of your organization. For more information, see [Using Active Directory with AppStream 2.0](#).
3. Configure your fleet scaling policies to increase or decrease the number of instances available to users in response to changes in user demand or according to time of day. For more information, see [Fleet Auto Scaling for Amazon AppStream 2.0](#).

**Important:** Remember to delete the resources that you created in these exercises to avoid further charges to your account. For information about how to delete AppStream 2.0 resources, see *Appendix B*. For more information about AppStream 2.0 pricing, see [Amazon AppStream 2.0 Pricing](#).

## Contributors

Contributors to this document include:

- Dylan Barlett, Senior EUC Solutions Architect
- Shun Mao, Senior Partner Solutions Architect
- Vinothkumar Narasimhan, Senior Product Manager - Technical

## Document revisions

Date	Description
March 2023	Updated for MATLAB R2022b and Graphics G4 instances, streamlined content.
October 2018	First publication.

## Appendix A: Create and activate an AWS account

If you do not already have an AWS account, complete the following steps to create and activate one. During this process, you do the following:

- Create your AWS account.
- Add a payment method.
- Verify your phone number.
- Select an AWS Support plan.
- Watch for three account confirmation emails.

### Create your AWS account

1. In a browser window, open the [Amazon Web Services](#) webpage.
2. Choose **Create a new AWS Account**. If you've signed in to AWS recently, you might see **Sign in to the Console** instead. If **Create a new AWS account** isn't visible, choose **Sign in to a different account, Create a new AWS account**.
3. On the **Create an AWS Account** page, type a valid email address, a password and password confirmation, and an AWS account name.
4. You must note the account name, email address, and password that you choose for your AWS account because you need these credentials to sign in to AWS.
5. Choose **Continue**.
6. On the **Contact Information** page, the option to choose a company account or personal account is available. These two account types function identically. For the exercises in this guide, choose **Personal Account**, and then enter the requested contact information.
7. Review the **AWS Customer Agreement**, and select the corresponding check box.
8. Choose **Create Account and Continue**.  
**Note:** After you receive an email to confirm that your account is created, you can sign in to your new account by using the email address and password that you provided. However, you must continue with the activation process before you can use AWS services.

### Add a payment method

On the **Payment Information** page, type the requested information associated with your payment method. If the address for your payment method is the same as the address you provided for your account, choose **Secure Submit**.

Otherwise, choose **Use a new address**, type the billing address for your payment method, and then choose **Secure Submit**.

### **Verify your phone number**

1. On the **Phone Verification** page, type a phone number that you can use to accept incoming calls.
2. Type the code displayed in the CAPTCHA.
3. When you're ready to receive the call, choose **Call me Now**. In a few moments, you'll receive an automated call from AWS that prompts you to enter your PIN to validate the AWS account.
4. When you receive the call, enter the provided PIN on your phone's keypad.
5. After the process is complete, choose **Continue**.

### **Choose an AWS Support plan**

On the **Select a Support Plan** page, choose **Basic**. For information about AWS Support, see [AWS Support Features](#).

After you choose a Support plan, a confirmation page indicates that your AWS account is being activated. Accounts are usually activated within a few minutes, but the process may take up to 24 hours. If you attempt to sign in to the AWS Management Console before your account is active, the following message appears:

#### **Your service sign-up is almost complete!**

Thanks for signing up with Amazon Web Services. Your services may take up to 24 hours to fully activate. If you're unable to access AWS services after that time, here are a few things you can do to expedite the process:

1. Make sure you provided all necessary information during signup. Complete your AWS registration.
2. Check your email to see if you have received any requests for additional information. If you have, please respond to those emails with the information requested.
3. Verify your credit card information is correct. Also, check your credit card activity to see if there's a \$1 authorization (this is not a charge). You may need to contact your card issuer to approve the authorization.

If the problem persists, please contact Support:



*Figure 17: Message that appears if you sign in before your account activation is complete.*

### **Watch for three AWS account confirmation emails**

When you sign up for your account, you receive three account confirmation emails:

- The first email, with a subject line of “Welcome to Amazon Web Services,” confirms the creation of your AWS account and is sent almost immediately after you verify your phone number.
- The second email, with a subject line of “AWS Support (Basic) Sign-Up Confirmation”, confirms the AWS support option that you selected during the account creation process.
- The third email, with a subject line of “Your AWS Account is Ready - Get Started Now,” is sent after your AWS account ID is ready to use. After you receive this email, you can access AWS services by using the [AWS Management Console](#).

## Appendix B. Clean up your AppStream 2.0 resources

Although you can continue to use this AppStream 2.0 environment, keep in mind that you pay for your running resources. For more information, see [Amazon AppStream 2.0 Pricing](#).

Cleaning up the resources that you created frees up resources and helps you avoid unintended charges to your account.

### *Stop and delete your image builder*

1. Open the AppStream 2.0 console at <https://console.aws.amazon.com/appstream2>.
2. In the navigation pane, choose **Images, Image Builder**.
3. Confirm whether the image builder that you created in Step 3 in this guide is in a stopped state. If not, select the image builder and choose **Actions, Stop**.
4. After the image builder has stopped, choose **Actions, Delete**. Repeat this step for each image builder that you created.

### *Disassociate your fleets from your stack and delete your stack*

1. In the navigation pane, choose **Stacks**.
2. Select the stack you created and choose **Actions, Dissociate Fleet**.
3. To delete the stack, choose **Actions, Delete**.

### *Stop and delete your fleet*

1. In the navigation pane, choose **Fleets**.
2. Confirm whether the fleet that you created in Step 6 in this guide is in a stopped state. If not, select the fleet and choose **Actions, Stop**.
3. After the fleet has stopped, choose **Actions, Delete**.