Create a custom Windows AMI

You can launch an instance from an existing Windows AMI, customize the instance, and then save this updated configuration as a custom AMI. Instances launched from this new custom AMI include the customizations that you made when you created the AMI.

To help categorize and manage your AMIs, you can assign custom tags to them. For more information, see [Tagging your Amazon EC2 resources](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Using_Tags.html).

To create a custom Linux AMI, use the procedure for the type of volume for the instance. For more information, see [Creating an Amazon EBS-Backed Linux AMI](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-ebs.html) or [Creating an Instance Store-Backed Linux AMI](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/creating-an-ami-instance-store.html) in the Amazon EC2 User Guide for Linux Instances.

## **How the creation of a custom AMI works**

First, launch an instance from an AMI that's similar to the AMI that you'd like to create. You can connect to your instance and customize it. When the instance is set up the way you want it, ensure data integrity by stopping the instance before you create an AMI and then create the image. We automatically register the AMI for you.

During the AMI-creation process, Amazon EC2 creates snapshots of your instance's root volume and any other EBS volumes attached to your instance. You're charged for the snapshots until you deregister the AMI and delete the snapshots. For more information, see [Deregister your Windows AMI](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/deregister-ami.html). If any volumes attached to the instance are encrypted, the new AMI only launches successfully on instance types that support Amazon EBS encryption. For more information, see [Amazon EBS encryption](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/EBSEncryption.html).

Depending on the size of the volumes, it can take several minutes for the AMI-creation process to complete (sometimes up to 24 hours). You may find it more efficient to create snapshots of your volumes prior to creating your AMI. This way, only small, incremental snapshots need to be created when the AMI is created, and the process completes more quickly (the total time for snapshot creation remains the same). For more information, see [Creating Amazon EBS snapshots](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/ebs-creating-snapshot.html).

After the process completes, you have a new AMI and snapshot created from the root volume of the instance. When you launch an instance using the new AMI, we create a new EBS volume for its root volume using the snapshot.

If you add instance store volumes or Amazon EBS volumes to your instance in addition to the root device volume, the block device mapping for the new AMI contains information for these volumes, and the block device mappings for instances that you launch from the new AMI automatically contain information for these volumes. The instance store volumes specified in the block device mapping for the new instance are new and don't contain any data from the instance store volumes of the instance you used to create the AMI. The data on EBS volumes persists. For more information, see [Block device mapping](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/block-device-mapping-concepts.html).

**Note:** When you create a new instance from a custom AMI, you should initialize both its root volume and any additional EBS storage before putting it into production. For more information, see [Initializing Amazon EBS Volumes](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/ebs-initialize.html).

## **Create a Windows AMI from a running instance**

You can create an AMI using the AWS Management Console or the command line. The following diagram summarizes the process for creating an AMI from a running EC2 instance. Start with an existing AMI, launch an instance, customize it, create a new AMI from it, and finally launch an instance of your new AMI. The steps in the following diagram match the steps in the procedure below. If you already have a running Windows instance, you can go directly to step 4.


     Workflow for creating an AMI from an instance
    

**To create an AMI from an instance using the console**

1. Open the Amazon EC2 console at <https://console.aws.amazon.com/ec2/>.
2. In the navigation pane, choose **Images**, **AMIs**.
3. Use the **Filter** options to scope the list of AMIs to the Windows AMIs that meet your needs. For example, to view the Windows AMIs provided by AWS, choose **Public images** from the drop-down list. Choose the Search bar. Choose **Owner** from the menu and choose **Amazon images**. Choose **Source** from the menu and type one of the following, depending on the version of Windows Server that you need:
   * **amazon/Windows\_Server-2019**
   * **amazon/Windows\_Server-2016**
   * **amazon/Windows\_Server-2012**
   * **amazon/Windows\_Server-2008**

Add any other filters that you need. When you have chosen an AMI, select its checkbox.

1. Choose **Launch**. Accept the default values as you step through the wizard. For more information, see [Launching an instance using the Launch Instance Wizard](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/launching-instance.html). When the instance is ready, connect to it. For more information, see [Connecting to your Windows instance](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/connecting_to_windows_instance.html).
2. You can perform any of the following actions on your instance to customize it for your needs:
   * Install software and applications
   * Copy data
   * Reduce start time by deleting temporary files, defragmenting your hard drive, and zeroing out free space
   * Attach additional EBS volumes
   * Create a new user account and add it to the Administrators group

If you are sharing your AMI, these credentials can be supplied for RDP access without disclosing your default administrator password.

* + [Windows Server 2016 and later] Configure settings using EC2Launch. To generate a random password at launch time, use the adminPasswordType setting. For more information, see [Configuring EC2Launch](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/ec2launch.html#ec2launch-config).
  + [Windows Server 2012 R2 and earlier] Configure settings using EC2Config. To generate a random password at launch time, enable the Ec2SetPassword plugin; otherwise, the current administrator password is used. For more information, see [EC2Config settings files](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/ec2config-service.html#UsingConfigXML_WinAMI).
  + [Windows Server 2008 R2] If the instance uses RedHat drivers to access Xen virtualized hardware, upgrade to Citrix drivers before you create an AMI. For more information, see [Upgrade Windows Server 2008 and 2008 R2 instances (Redhat to Citrix PV upgrade)](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Upgrading_PV_drivers.html#win2008-citrix-upgrade).

1. In the navigation pane, choose **Instances** and select your instance. Choose **Actions**, **Image and templates**, and **Create image**.

**Tip:** If this option is disabled, your instance isn't an Amazon EBS-backed instance.

1. Specify a unique name for the image and an optional description (up to 255 characters).

By default, Amazon EC2 shuts down the instance, takes snapshots of any attached volumes, creates and registers the AMI, and then reboots the instance. Choose **No reboot** if you don't want your instance to be shut down.

**Warning:** If you choose **No reboot**, we can't guarantee the file system integrity of the created image. (Optional) Modify the root volume, Amazon EBS volumes, and instance store volumes as needed. For example:

* + To change the size of the root volume, locate the **Root** volume in the **Type** column, and fill in the **Size** field.
  + To suppress an Amazon EBS volume specified by the block device mapping of the AMI used to launch the instance, locate the EBS volume in the list and choose **Delete**.
  + To add an Amazon EBS volume, choose **Add New Volume**, **Type**, and **EBS**, and fill in the fields. When you then launch an instance from your new AMI, these additional volumes are automatically attached to the instance. Empty volumes must be formatted and mounted. Volumes based on a snapshot must be mounted.
  + To suppress an instance store volume specified by the block device mapping of the AMI used to launch the instance, locate the volume in the list and choose **Delete**.
  + To add an instance store volume, choose **Add New Volume**, **Type**, and **Instance Store**, and select a device name from the **Device** list. When you launch an instance from your new AMI, these additional volumes are automatically initialized and mounted. These volumes don't contain data from the instance store volumes of the running instance from which you based your AMI.

When you are finished, choose **Create Image**.

1. While your AMI is being created, you can choose **AMIs** in the navigation pane to view its status. Initially, this is pending. After a few minutes, the status should change to available.

(Optional) Choose **Snapshots** in the navigation pane to view the snapshot that was created for the new AMI. When you launch an instance from this AMI, we use this snapshot to create its root device volume.

1. Launch an instance from your new AMI. For more information, see [Launching an instance using the Launch Instance Wizard](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/launching-instance.html). The new running instance contains all of the customizations you applied in previous steps, and any additional customization you add when launching the instance, such as user data (scripts that run when the instance starts).

### To create an AMI from an instance using the command line

You can use one of the following commands. For more information about these command line interfaces, see [Accessing Amazon EC2](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/concepts.html#access-ec2).

* [create-image](https://docs.aws.amazon.com/cli/latest/reference/ec2/create-image.html) (AWS CLI)
* [New-EC2Image](https://docs.aws.amazon.com/powershell/latest/reference/items/New-EC2Image.html) (AWS Tools for Windows PowerShell)

## **Create a standardized Amazon Machine Image (AMI) using Sysprep**

The Microsoft System Preparation (Sysprep) tool simplifies the process of duplicating a customized installation of Windows. You can use Sysprep to create a standardized Amazon Machine Image (AMI). You can then create new Amazon EC2 instances for Windows from this standardized image.

We recommend that you use [EC2 Image Builder](https://docs.aws.amazon.com/imagebuilder/latest/userguide/what-is-image-builder.html) to automate the creation, management, and deployment of customized, secure, and up-to-date "golden" server images that are pre-installed and preconfigured with software and settings.

If you use Sysprep to create a standardized AMI, we recommend that you run Sysprep with [EC2Launch v2](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/ec2launch-v2.html). If you are still using the EC2Config (Windows Server 2012 R2 and earlier) or EC2Launch (Windows Server 2016 and later) agents, see the documentation for using Sysprep with EC2Config and EC2Launch below.

**Important:** Do not use Sysprep to create an instance backup. Sysprep removes system-specific information; removing this information might have unintended consequences for an instance backup.

To troubleshoot Sysprep, see [Troubleshooting Sysprep](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/sysprep-troubleshoot.html).

**Contents**

* [Before you begin](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Creating_EBSbacked_WinAMI.html#sysprep-begin)
* [Use Sysprep with EC2Launch v2](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Creating_EBSbacked_WinAMI.html#sysprep-using-ec2launchv2)
* [Use Sysprep with EC2Launch](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Creating_EBSbacked_WinAMI.html#ec2launch-sysprep)
* [Use Sysprep with EC2Config](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Creating_EBSbacked_WinAMI.html#sysprep-using)

### Before you begin

* Before performing Sysprep, we recommend that you remove all local user accounts and all account profiles other than a single administrator account under which Sysprep will be executed. If you perform Sysprep with additional accounts and profiles, unexpected behavior could result, including loss of profile data or failure to complete Sysprep.
* Learn more about [Sysprep](https://technet.microsoft.com/en-us/library/cc721940.aspx" \t "_blank) on Microsoft TechNet.
* Learn which [server roles are supported for Sysprep](https://docs.microsoft.com/en-us/windows-hardware/manufacture/desktop/sysprep-support-for-server-roles).

### Use Sysprep with EC2Launch v2: This section contains details about the different Sysprep execution phases and the tasks performed by the EC2Launch v2 service as the image is prepared. It also includes the steps to create a standardized AMI using Sysprep with the EC2Launch v2 service.

**Sysprep with EC2Launch v2 topics**

* [Sysprep phases](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Creating_EBSbacked_WinAMI.html#sysprep-phases)
* [Sysprep actions](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Creating_EBSbacked_WinAMI.html#sysprep-actions-ec2launchv2)
* [Post Sysprep](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Creating_EBSbacked_WinAMI.html#sysprep-post-ec2launchv2)
* [Run Sysprep with EC2Launch v2](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Creating_EBSbacked_WinAMI.html#sysprep-gui-procedure-ec2launchv2)

#### Sysprep phases: Sysprep runs through the following phases:

* **Generalize**: The tool removes image-specific information and configurations. For example, Sysprep removes the security identifier (SID), the computer name, the event logs, and specific drivers, to name a few. After this phase is completed, the operating system (OS) is ready to create an AMI.

**Note:** When you run Sysprep with the EC2Launch v2 service, the system prevents drivers from being removed because the PersistAllDeviceInstalls setting is set to true by default.

* **Specialize**: Plug and Play scans the computer and installs drivers for any detected devices. The tool generates OS requirements, like the computer name and SID. Optionally, you can execute commands in this phase.
* **Out-of-Box Experience (OOBE)**: The system runs an abbreviated version of Windows Setup and asks you to enter information such as system language, time zone, and registered organization. When you run Sysprep with EC2Launch v2, the answer file automates this phase.

#### Sysprep actions : Sysprep and EC2Launch v2 perform the following actions when preparing an image.

1. When you choose **Shutdown with Sysprep** in the **EC2Launch settings** dialog box, the system runs the ec2launch sysprep command.
2. EC2Launch v2 edits the content of the unattend.xml file by reading the registry value at HKEY\_USERS\.DEFAULT\Control Panel\International\LocaleName. This file is located in the following directory: C:\ProgramData\Amazon\EC2Launch\sysprep.
3. The system executes the BeforeSysprep.cmd. This command creates a registry key as follows:

**reg add "HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG\_DWORD /d 1 /f**

The registry key disables RDP connections until they are re-enabled. Disabling RDP connections is a necessary security measure because, during the first boot session after Sysprep has run, there is a short period of time where RDP allows connections and the Administrator password is blank.

1. The EC2Launch v2 service calls Sysprep by running the following command:

**sysprep.exe /oobe /generalize /shutdown /unattend: "C:\ProgramData\Amazon\EC2Launch\sysprep\unattend.xml"**

##### Generalize phase

* EC2Launch v2 removes image-specific information and configurations, such as the computer name and the SID. If the instance is a member of a domain, it is removed from the domain. The unattend.xml answer file includes the following settings that affect this phase:
  + **PersistAllDeviceInstalls**: This setting prevents Windows Setup from removing and reconfiguring devices, which speeds up the image preparation process because Amazon AMIs require certain drivers to run and re-detection of those drivers would take time.
  + **DoNotCleanUpNonPresentDevices**: This setting retains Plug and Play information for devices that are not currently present.
* Sysprep shuts down the OS as it prepares to create the AMI. The system either launches a new instance or starts the original instance.

##### Specialize phase : The system generates OS-specific requirements, such as a computer name and an SID. The system also performs the following actions based on configurations that you specify in the unattend.xml answer file.

* **CopyProfile**: Sysprep can be configured to delete all user profiles, including the built-in Administrator profile. This setting retains the built-in Administrator account so that any customizations you make to that account are carried over to the new image. The default value is True.

**CopyProfile** replaces the default profile with the existing local administrator profile. All accounts that you log in to after running Sysprep receive a copy of that profile and its contents at first login.

If you don’t have specific user-profile customizations that you want to carry over to the new image, then change this setting to False. Sysprep will remove all user profiles (this saves time and disk space).

* **TimeZone**: The time zone is set to Coordinate Universal Time (UTC) by default.
* **Synchronous command with order 1**: The system executes the following command, which enables the administrator account and specifies the password requirement:

**net user Administrator /ACTIVE:YES /LOGONPASSWORDCHG:NO /EXPIRES:NEVER /PASSWORDREQ:YES**

* **Synchronous command with order 2**: The system scrambles the administrator password. This security measure is designed to prevent the instance from being accessible after Sysprep completes if you did not enable the ec2setpassword setting.

**C:\Program Files\Amazon\Ec2ConfigService\ScramblePassword.exe" -u Administrator**

* **Synchronous command with order 3**: The system executes the following command:

**C:\Program Files\Amazon\Ec2ConfigService\Scripts\SysprepSpecializePhase.cmd**

This command adds the following registry key, which re-enables RDP:

**reg add "HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\Terminal Server" /v fDenyTSConnections /t REG\_DWORD /d 0 /f**

##### OOBE phase

1. The system specifies the following configurations using the EC2Launch v2 answer file:
   * <InputLocale>en-US</InputLocale>
   * <SystemLocale>en-US</SystemLocale>
   * <UILanguage>en-US</UILanguage>
   * <UserLocale>en-US</UserLocale>
   * <HideEULAPage>true</HideEULAPage>
   * <HideWirelessSetupInOOBE>true</HideWirelessSetupInOOBE>
   * <ProtectYourPC>3</ProtectYourPC>
   * <BluetoothTaskbarIconEnabled>false</BluetoothTaskbarIconEnabled>
   * <TimeZone>UTC</TimeZone>
   * <RegisteredOrganization>Amazon.com</RegisteredOrganization>
   * <RegisteredOwner>EC2</RegisteredOwner>

**Note:** During the generalize and specialize phases, EC2Launch v2 monitors the status of the OS. If EC2Launch v2 detects that the OS is in a Sysprep phase, then it publishes the following message to the system log:

**Windows is being configured. SysprepState=IMAGE\_STATE\_UNDEPLOYABLE**

1. The system runs EC2Launch v2.

#### Post Sysprep

After Sysprep completes, EC2Launch v2 sends the following message to the console output:

Windows sysprep configuration complete.

EC2Launch v2 then performs the following actions:

1. Reads the content of the agent-config.yml file and runs configured tasks.
2. Executes all tasks in the preReady stage.
3. After it is finished, sends a Windows is ready message to the instance system logs.
4. Executes all tasks in the PostReady stage.

For more information about EC2Launch v2 , see [Configuring a Windows instance using EC2Launch v2](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/ec2launch-v2.html).

#### Run Sysprep with EC2Launch v2 : Use the following procedure to create a standardized AMI using Sysprep with EC2Launch v2.

1. In the Amazon EC2 console, locate or [create](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Creating_EBSbacked_WinAMI.html) an AMI that you want to duplicate.
2. Launch and connect to your Windows instance.
3. Customize it.
4. From the Windows **Start** menu, search for and choose **Amazon EC2Launch settings**. For more information about the options and settings in the Amazon **EC2Launch settings** dialog box, see [EC2Launch v2 settings](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/ec2launch-v2-settings.html).
5. Select **Shutdown with Sysprep** or **Shutdown without Sysprep**.

When you are asked to confirm that you want to run Sysprep and shut down the instance, click **Yes**. EC2Launch v2 runs Sysprep. Next, you are logged off the instance, and the instance shuts down. If you check the **Instances** page in the Amazon EC2 console, the instance state changes from Running to Stopping to Stopped. At this point, it's safe to create an AMI from this instance.

You can manually invoke the Sysprep tool from the command line using the following command:

**"%programfiles%\amazon\ec2launch\ec2launch.exe" sysprep**

### Use Sysprep with EC2Launch

EC2Launch offers a default answer file and batch files for Sysprep that automate and secure the image-preparation process on your AMI. Modifying these files is optional. These files are located in the following directory by default: C:\ProgramData\Amazon\EC2-Windows\Launch\Sysprep.

**Important:** Do not use Sysprep to create an instance backup. Sysprep removes system-specific information. If you remove this information there might be unintended consequences for an instance backup.

**Sysprep with EC2Launch topics**

* [EC2Launch answer and batch files for Sysprep](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Creating_EBSbacked_WinAMI.html#ec2launch-sysprep-answer-batch)
* [Running Sysprep with EC2Launch](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Creating_EBSbacked_WinAMI.html#ec2launch-sysprep-running)
* [Updating metadata/KMS routes for Server 2016 and later when launching a custom AMI](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/Creating_EBSbacked_WinAMI.html#update-metadata-KMS)

#### EC2Launch answer and batch files for Sysprep

The EC2Launch answer file and batch files for Sysprep include the following:

Unattend.xml

This is the default answer file. If you run SysprepInstance.ps1 or choose **ShutdownWithSysprep** in the user interface, the system reads the setting from this file.

BeforeSysprep.cmd

Customize this batch file to run commands before EC2Launch runs Sysprep.

SysprepSpecialize.cmd

Customize this batch file to run commands during the Sysprep specialize phase.

#### Running Sysprep with EC2Launch

On the full installation of Windows Server 2016 and later (with a desktop experience), you can run Sysprep with EC2Launch manually or by using the **EC2 Launch Settings** application.

**To run Sysprep using the EC2Launch Settings application**

1. In the Amazon EC2 console, locate or create a Windows Server 2016 or later AMI.
2. Launch a Windows instance from the AMI.
3. Connect to your Windows instance and customize it.
4. Search for and run the **EC2LaunchSettings** application. It is located in the following directory by default: C:\ProgramData\Amazon\EC2-Windows\Launch\Settings.


       EC2 Launch Settings application
      

1. Select or clear options as needed. These settings are stored in the LaunchConfig.json file.
2. For **Administrator Password**, do one of the following:
   * Choose **Random**. EC2Launch generates a password and encrypts it using the user's key. The system disables this setting after the instance is launched so that this password persists if the instance is rebooted or stopped and started.
   * Choose **Specify** and type a password that meets the system requirements. The password is stored in LaunchConfig.json as clear text and is deleted after Sysprep sets the administrator password. If you shut down now, the password is set immediately. EC2Launch encrypts the password using the user's key.
   * Choose **DoNothing** and specify a password in the unattend.xml file. If you don't specify a password in unattend.xml, the administrator account is disabled.
3. Choose **Shutdown with Sysprep**.

**To manually run Sysprep using EC2Launch**

1. In the Amazon EC2 console locate or create a Windows Server 2016 or later Datacenter edition AMI that you want to duplicate.
2. Launch and connect to your Windows instance.
3. Customize the instance.
4. Specify settings in the LaunchConfig.json file. This file is located in the C:\ProgramData\Amazon\EC2-Windows\Launch\Config directory by default.

For adminPasswordType, specify one of the following values:

Random

EC2Launch generates a password and encrypts it using the user's key. The system disables this setting after the instance is launched so that this password persists if the instance is rebooted or stopped and started.

Specify

EC2Launch uses the password you specify in adminPassword. If the password does not meet the system requirements, EC2Lauch generates a random password instead. The password is stored in LaunchConfig.json as clear text and is deleted after Sysprep sets the administrator password. EC2Launch encrypts the password using the user's key.

DoNothing

EC2Launch uses the password you specify in the unattend.xml file. If you don't specify a password in unattend.xml, the administrator account is disabled.

1. (Optional) Specify settings in unattend.xml and other configuration files. If plan to attend to the installation, then you don't need to make changes in these files. The files are located in the following directory by default: C:\ProgramData\Amazon\EC2-Windows\Launch\Sysprep.
2. In Windows PowerShell, run ./InitializeInstance.ps1 -Schedule. The script is located in the following directory, by default: C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts. This script schedules the instance to initialize during the next boot. You must run this script before you run the SysprepInstance.ps1 script in the next step.
3. In Windows PowerShell, run ./SysprepInstance.ps1. The script is located in the following directory by default: C:\ProgramData\Amazon\EC2-Windows\Launch\Scripts.

You are logged off the instance and the instance shuts down. If you check the **Instances** page in the Amazon EC2 console, the instance state changes from Running to Stopping, and then to Stopped. At this point, it is safe to create an AMI from this instance.

# Create a custom Windows-based image

The process for creating a Windows-based image that meets all of the requirements to run in Strigo.



Written by Daniel Wolf  
Updated over a week ago

## **Overview**

This article describes the actions to take in order to create an image in general, and in order to make sure it meets the prerequisites for running in Strigo.

The process of creating an image for Strigo involves:

1. Launching an EC2 instance as a base instance, from which an image will be created.
2. Install all of the software required for your training.
3. Set some configurations for the image to be accessible to Strigo.
4. Create an image out of that base instance.

## **Before you begin**

It is advised to create the base instance, and then create the image based on it, already on the region where the Lab Resource in Strigo is configured to run (by default, the *eu-central-1 (Frankfurt)* region).

*Note: As AWS EC2* [*doesn't officially support Windows Desktop*](https://aws.amazon.com/windows/?analyst-reports-windows.sort-by=item.additionalFields.datePublished&analyst-reports-windows.sort-order=desc) *(such as Windows 10), Strigo doesn't support it either. You may be able to get Windows 10 working with Strigo, but we may not be able to support you.*

## **Configuring the instance**

### Launch an AWS instance as a base for the eventual AMI

1. Launch an AWS EC2 instance on your AWS account, based on any Windows AMI as a starting point (whether from the AWS Marketplace, or an AMI on your account).
2. For a comprehensive guide on how to launch a Windows instance in AWS, see [here](https://aws.amazon.com/getting-started/tutorials/launch-windows-vm/).

### Connect to the instance

Connect to the instance using an RDP client.

### Prepare the instance to your liking

This is where you install all of the software required for training, change all of the settings that are required for the machine to run as you like, and so on.

Once you are ready with this step, move on to make last configuration preparations before the instance can be created as an image.

### Update the AWS drivers (recommended)

*Note: While this step isn't mandatory, we'e found it solves a lot of machine startup issues as Amzon updates its infrastructure (in particular, old versions of the PV and NVMe drivers tend to break with newer volume types).*

The easiest - and recommended - way to keep your AWS drivers is to execute [Amazon's update automation](https://docs.aws.amazon.com/AWSEC2/latest/WindowsGuide/migrating-latest-types.html#auto-upgrade) on your running instance, but if you prefer a manual or selective update, the above link extensively documents the available options.

### Run the AWS settings utility

* On Windows Server 2016 or later, run the EC2Launch utility.
* On Windows Server 2012 or lower, run the EC2Config utility.

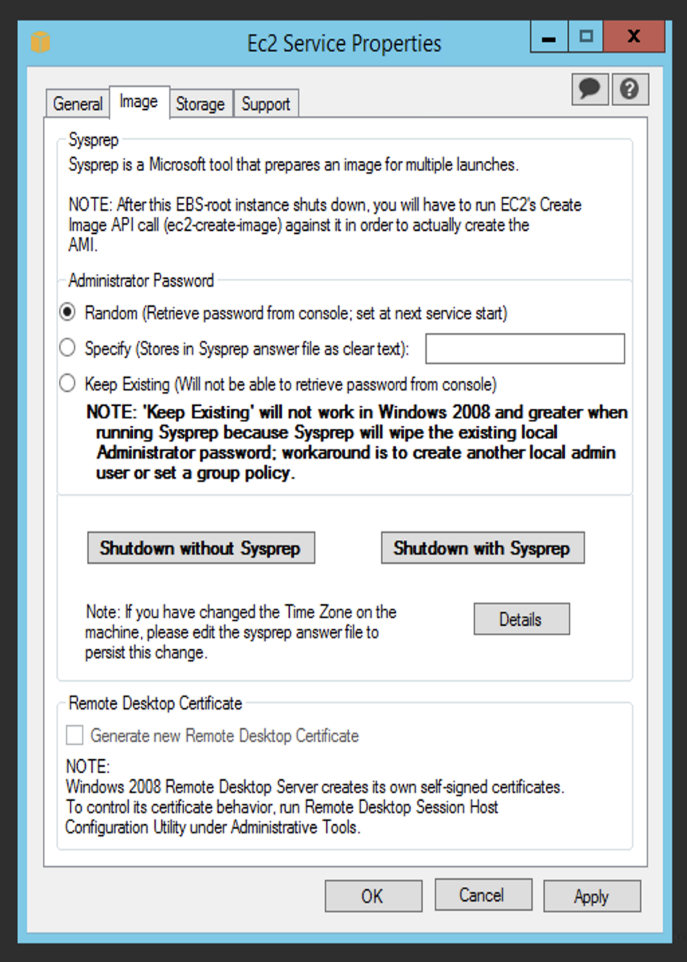
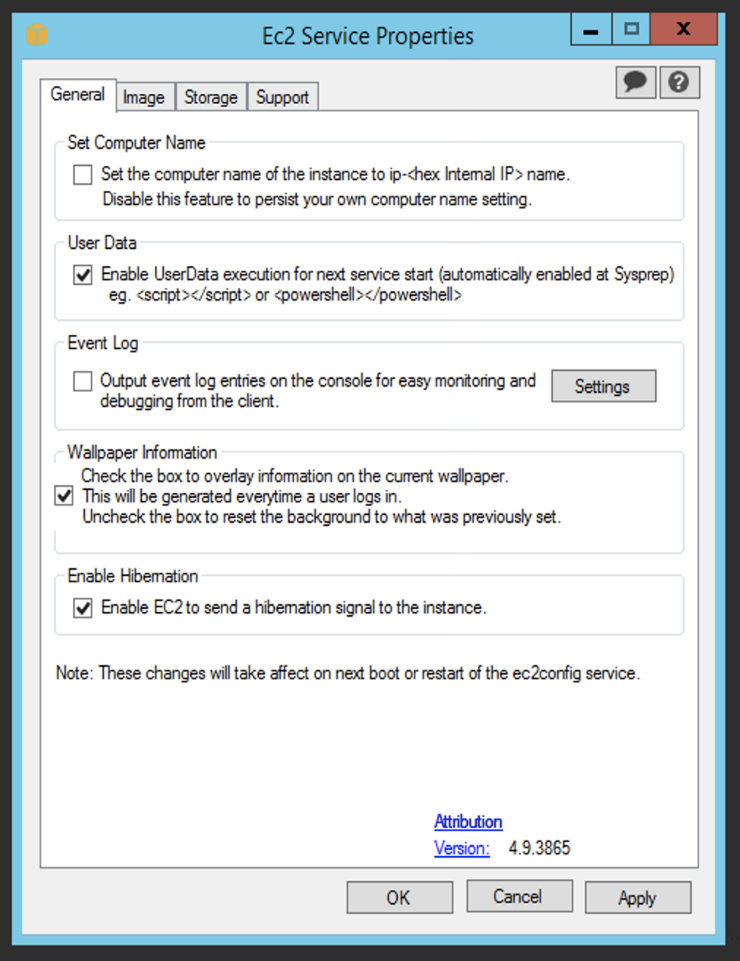
Open the Start menu, type in either "EC2Launch" or "EC2Config" and run the settings utility you find there.

### Set the AWS configuration

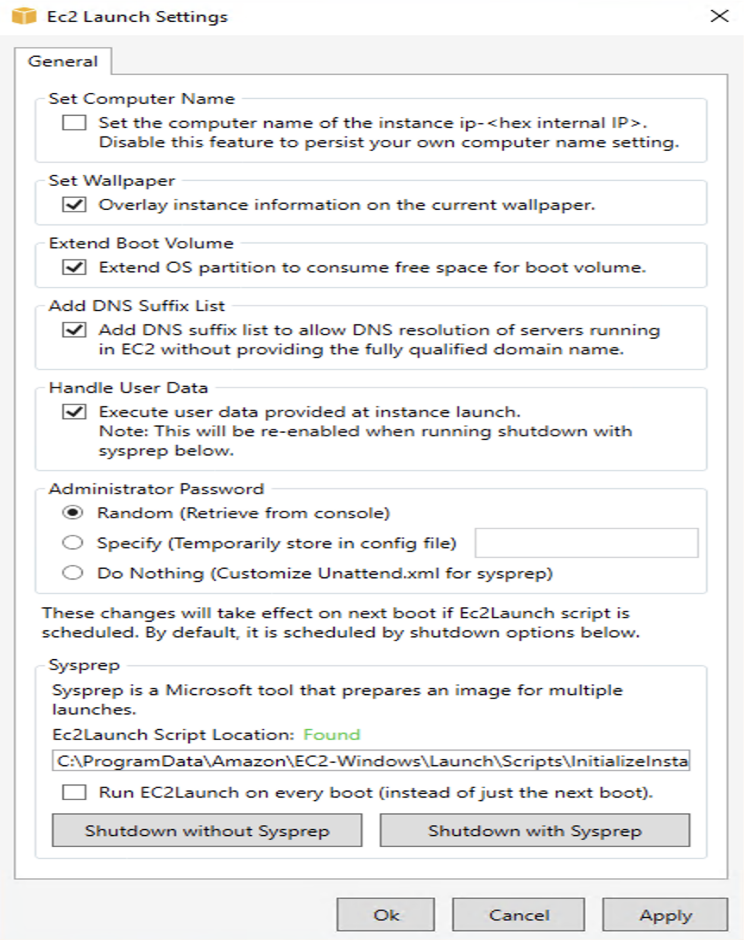
These are the configurations to set in the settings utility:

1. Make sure Administrator Password is set to Random.
2. Make sure User Data is enabled.
3. Apply the changes, then hit OK.

With EC2Config:



With EC2Launch:



## **Creating the image**

### Shut-down the instance

1. Run the EC2Launch/EC2Config settings utility again, as before.
2. Hit the *Shutdown without Sysprep* option.

*Note: shutting down without Sysprep is not mandatory, but is recommended by Strigo for most use-cases. Please contact us* [*here*](https://strigo.io/support) *to get a better understanding of this.*

### Create the image

Go back to the EC2 console, and wait for the instance to be stopped.  
Once it has stopped, you can use this instance as a base for creating an AMI usable in Strigo. You can use [this](https://docs.aws.amazon.com/toolkit-for-visual-studio/latest/user-guide/tkv-create-ami-from-instance.html) guide for help.

### Use the image in Strigo

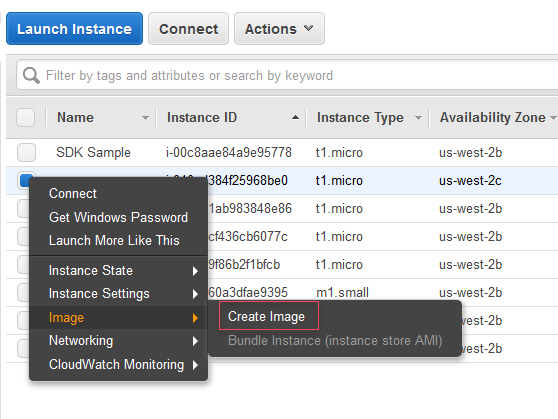
Please refer to [this](http://help.strigo.io/en/articles/1941452-using-custom-lab-images) article in order to use the image in Strigo.

Create an AMI from an Amazon EC2 Instance

From the **Amazon EC2 Instances** view, you can create Amazon Machine Images (AMIs) from either running or stopped instances.

*To create an AMI from an instance*

1. Right-click the instance you want to use as the basis for your AMI, and choose **Create Image** from the context menu.

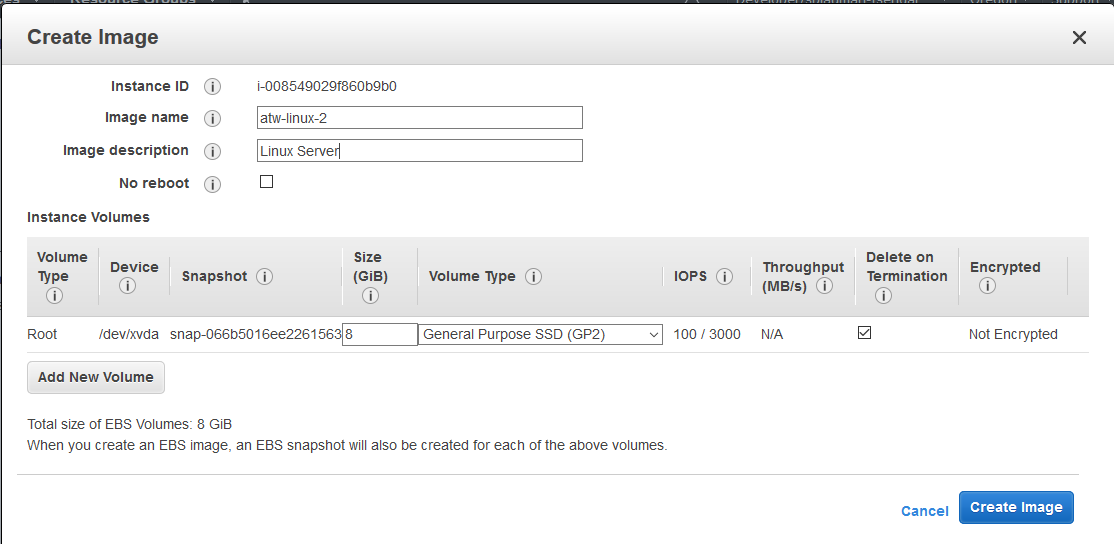


**Create Image** context menu

1. In the **Create Image** dialog box, type a unique name and description, and then choose **Create Image**. By default, Amazon EC2 shuts down the instance, takes snapshots of any attached volumes, creates and registers the AMI, and then reboots the instance. Choose **No reboot**if you don't want your instance to be shut down.

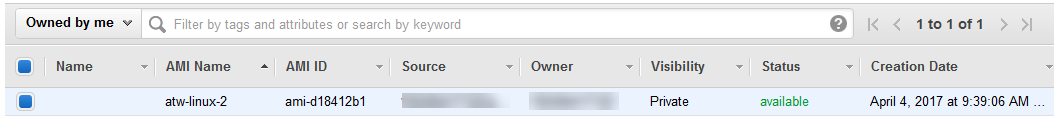
**Warning**

If you choose **No reboot**, we can't guarantee the file system integrity of the created image.



**Create Image** dialog box

It may take a few minutes for the AMI to be created. After it is created, it will appear in the **AMIs** view in AWS Explorer. To display this view, double-click the **Amazon EC2 | AMIs** node in AWS Explorer. To see your AMIs, from the **Viewing** drop-down list, choose **Owned By Me**. You may need to choose **Refresh** to see your AMI. When the AMI first appears, it may be in a pending state, but after a few moments, it transitions to an available state.



List of created AMIs