

Create Azure Resource Locks on a Web App

Understand the scenario

You are an Azure® administrator. You need to create an Azure web app that includes resource locks. First, you will create a web app that uses Linux®. Next, you will deploy the source code for the web app from a Docker Hub container image. Finally, you will enable resource locks, and then you will test the configuration of the web app.

Understand your environment

You will be using an Azure resource group named corp-datalod26434248 that contains no resources.

# **Create a web app**

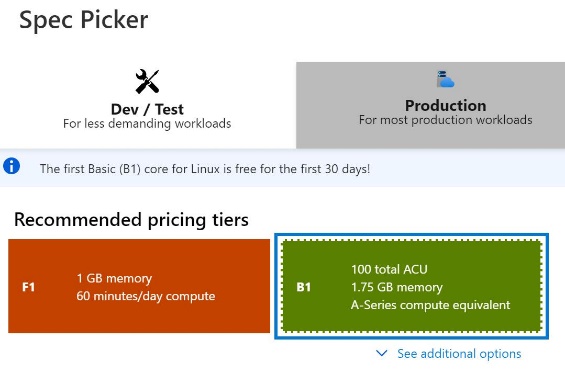
* Sign in to the Azure portal
* Create an Azure **web app** by using the values in the following table. For any property that is not specified, use the default value.

| **Property** | **Value** |
| --- | --- |
| Resource Group | **corp-datalod26434248** |
| Name | wa26434248 |
| Publish | **Docker Container** |
| Operating System | **Linux** |
| Region | **East US** |
| Linux Plan (East US) | AppPlan1 |
| Sku and size | **Basic B1** |
| Options | **Single Container** |
| Image Source | **Docker Hub** |
| Access Type | **Public** |
| Image and tag | nginx:latest |

* Expand this hint for guidance on creating an Azure web app.
  + On the Azure portal home page, select **Create a resource** to open the Azure Marketplace.
  + In the Popular offers list, select **Web App**, and then select **Create**.



* + On the Create Web App blade, on the Basics page, in Resource Group, select **corp-datalod26434248**, and then in Name, enter wa26434248.
  + In Publish, select **Docker Container**, in Operating System, ensure that **Linux** is selected, and then in Region, select **East US**.
  + In Linux Plan (East US), select **Create new**, in Name, enter AppPlan1, and then select **OK**.
  + In Sku and size, select **Change size**.
  + On the Spec picker page, select **Dev/Test**, in Recommended pricing tiers, select **B1**, and then select **Apply**.



* + On the Docker page, in Options, ensure that **Single Container** is selected, in Image Source, select **Docker Hub**, in Access Type, ensure that **Public** is selected, and then in Image and tag, enter nginx:latest.
  + Select **Review + create**, review the configuration, and then select **Create**.

You use an Azure [App Service](https://docs.microsoft.com/en-us/azure/app-service/overview) web app to host your website in the cloud in a fully managed environment. An [App Service plan](https://docs.microsoft.com/en-us/azure/app-service/overview-hosting-plans) defines the capacity and scalability of the web servers that support your website.

* In a new browser window, go to the URL for the new web app at https://wa26434248.azurewebsites.net to verify that it is up and running.

You should see the nginx default home page.



If you do not see the nginx default home page, refresh the browser. It may take approximately 1–2 minutes for the nginx container to fully deploy after the web app has deployed.

## Check your work

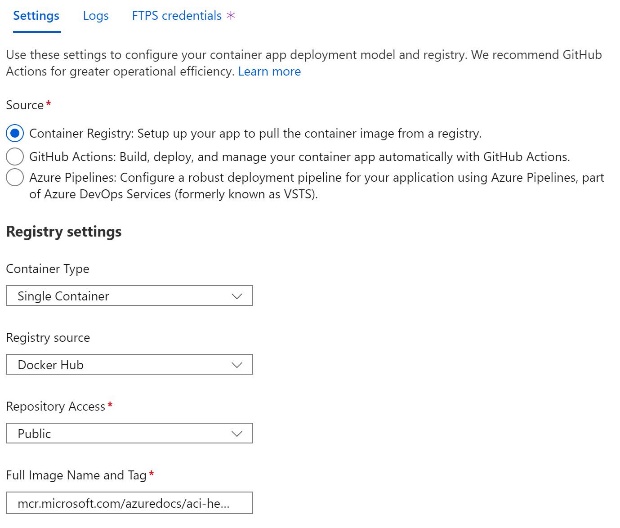
* Confirm that you created a web app named wa26434248.
* Confirm that you displayed the new web app in a browser.

# **Deploy a Docker Hub container image**

* Deploy a sample container image to the **wa26434248** web app by using the values in the following table. For any property that is not specified, use the default value.

| **Property** | **Value** |
| --- | --- |
| Source | **Container Registry** |
| Container Type | **Single Container** |
| Registry source | **Docker Hub** |
| Repository Access | **Public** |
| Full Image Name and Tag | mcr.microsoft.com/azuredocs/aci-helloworld:latest |

* Expand this hint for guidance on deploying a sample container image to a web app.
  + On the Azure portal home page, select **All resources**, and then select the **wa26434248** web app.
  + On the wa26434248 resource menu, select **Deployment Center**.
  + On the Deployment Center page, on the Settings page, in Source, select **Container Registry**.
  + In Container Type, ensure that **Single Container** is selected.
  + In Registry source, ensure that **Docker Hub** is selected, in Repository Access, ensure that **Public** is selected, and then in Full Image Name and Tag, enter mcr.microsoft.com/azuredocs/aci-helloworld:latest.



* + On the command bar, select **Save** to initiate the deployment.
* In a new browser window, go to the URL for the web app at https://wa26434248.azurewebsites.net to verify that the web app was updated.

**Welcome to Azure Container Instances!** should be displayed.



If you do not see the updated page, refresh the browser. It may take approximately 1–2 minutes for the container to fully deploy.

## Check your work

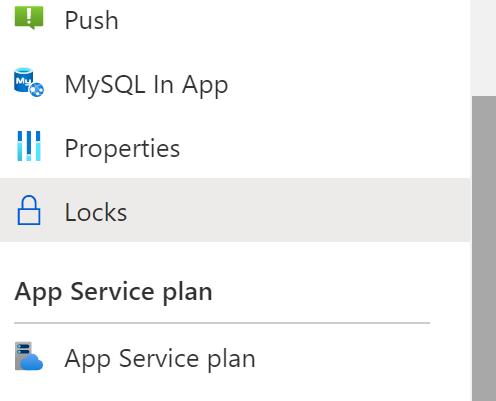
* Confirm that you deployed a sample container image to the wa26434248 web app.
* Confirm that you tested the updated wa26434248 web app.

# **Enable resource locks**

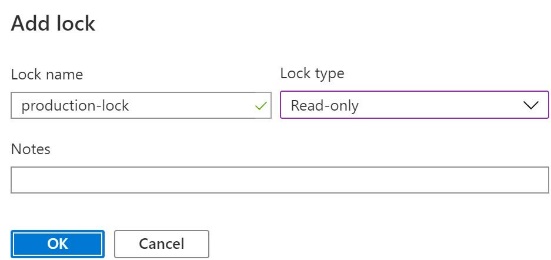
* Add a **read-only** resource lock named production-lock to the wa26434248 web app to prevent all unexpected changes in production.

Expand this hint for guidance on adding a read-only resource lock to a web app.

* + On the Azure portal home page, select **All resources**, and then select the **wa26434248** App Service.
  + On the wa26434248 resource menu, in Settings, select **Locks**.



* + On the Locks page, on the command bar, select **Add**.
  + On the Add lock blade, in Lock name, enter production-lock, in Lock type, select **Read-only**, and then select **OK**.

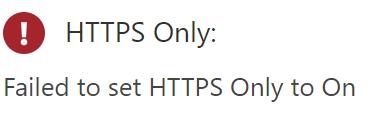


* Attempt to change HTTPS Only to **On** in the TLS/SSL settings for the web app.

Expand this hint for guidance on changing the TLS/SSL settings for a web app.

* + On the wa26434248 resource menu, in Settings, select **TLS/SSL settings**.
  + In HTTPS Only, attempt to select **On**.

You should see an error message that indicates that the resource is read-only.



* Edit the **production-lock** resource lock to prevent only an unexpected delete in production.

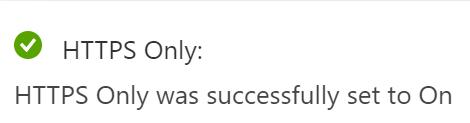
Expand this hint for guidance on configuring a delete resource lock to a web app.

* + On the wa26434248 resource menu, select **Locks**, and then in the production-lock row, select **Edit**.

The Edit option

* + On the Edit lock blade, in Lock type, select **Delete**, and then select **OK**.
* Attempt to change HTTPS Only to **On** in the TLS/SSL settings for the web app.

You should see a success message this time as the lock now restricts only a deletion of the resource itself.



## Check your work

* Confirm that you added a resource lock to prevent all unexpected changes.
* Confirm that you edited a resource lock to prevent only an unexpected delete.

# **Summary**

Congratulations, you have completed the **Create Azure Resource Locks on a Web App** challenge.

You have accomplished the following:

* Created a Linux web app.
* Deployed a Docker Hub container image.
* Enabled resource locks.