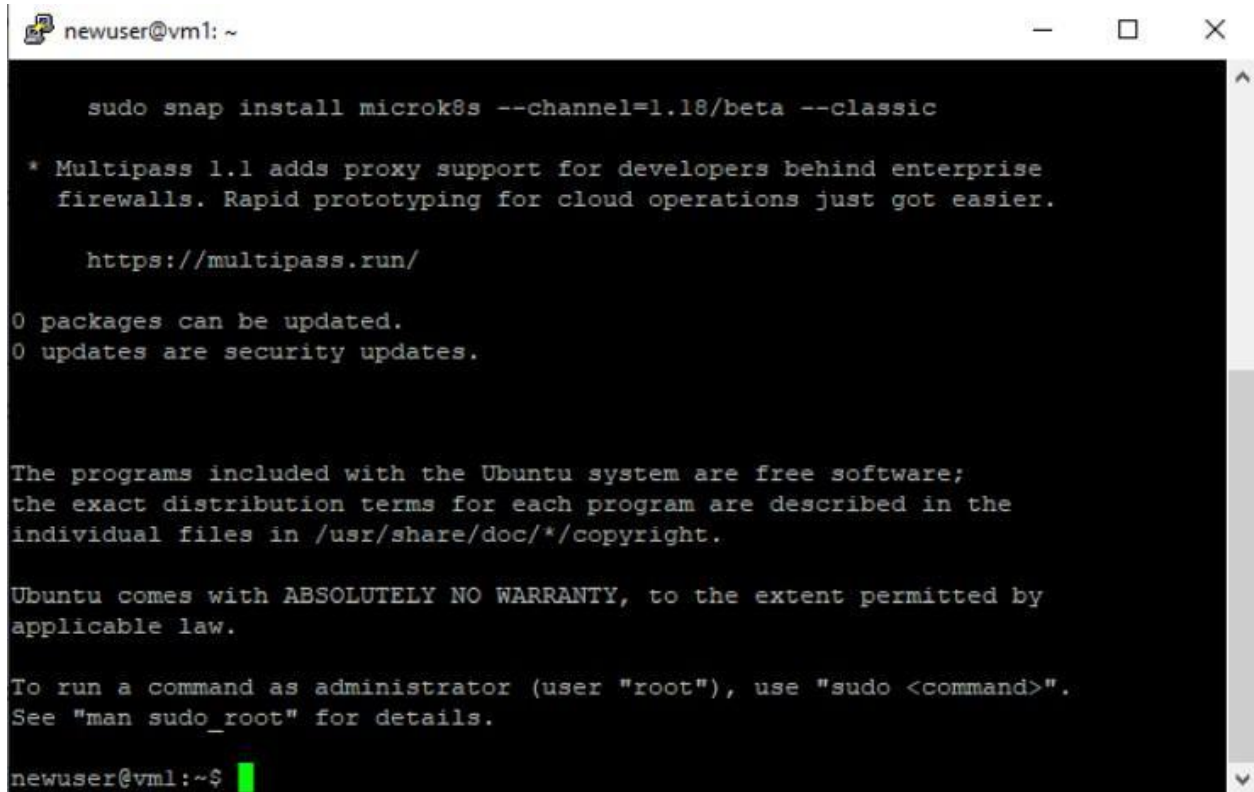


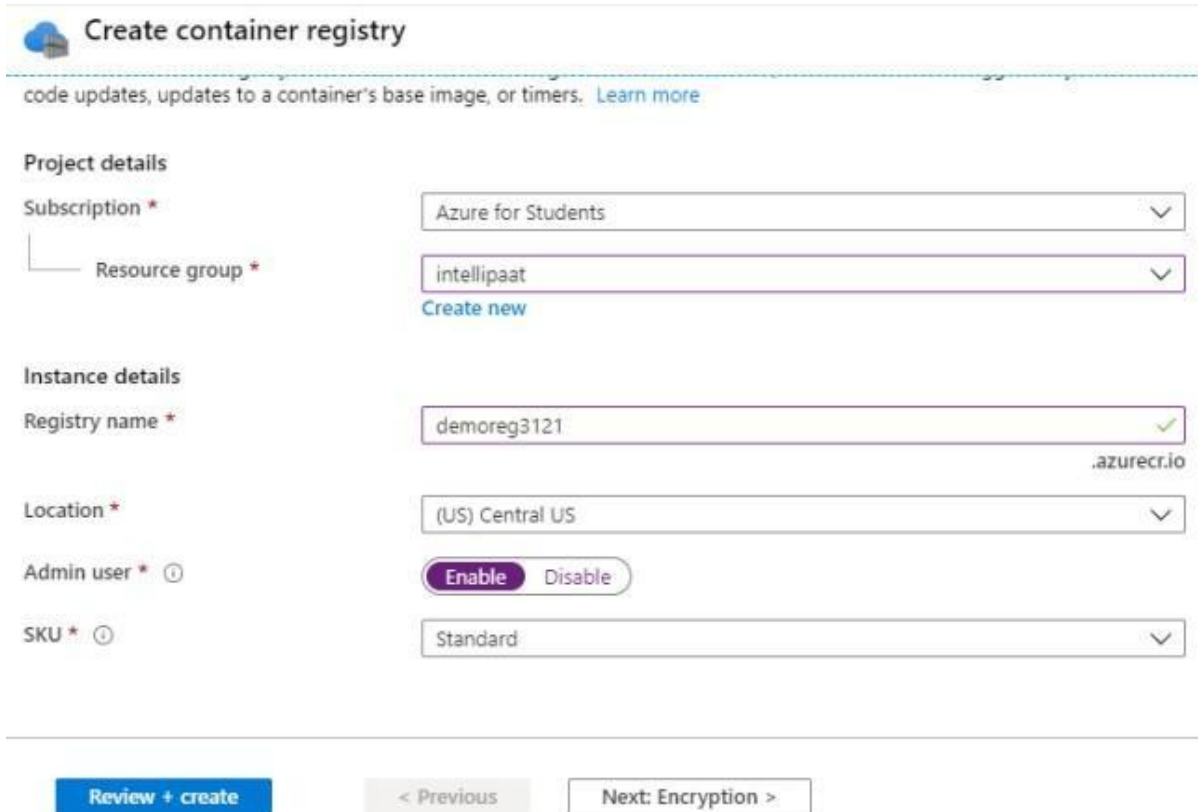


Module 5: Hands-On: Publishing and Automating Image Deployment to Azure Container Registry

Step 1: Deploy an Ubuntu virtual machine and install the Azure CLI and Docker

```
newuser@vm1: ~  
  
sudo snap install microk8s --channel=1.18/beta --classic  
  
* Multipass 1.1 adds proxy support for developers behind enterprise  
  firewalls. Rapid prototyping for cloud operations just got easier.  
  
https://multipass.run/  
  
0 packages can be updated.  
0 updates are security updates.  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
newuser@vm1:~$
```

Step 2: Search for container registries and click on it. Then, click on Add. Enter details such as the resource group and the name of the registry. Switch the Admin user to Enable. Then click on Next: Encryption



Create container registry

code updates, updates to a container's base image, or timers. [Learn more](#)

Project details

Subscription *

Resource group *
[Create new](#)

Instance details

Registry name * .azurecr.io

Location *


Admin user * ☒ Enable ☐ Disable

SKU *

[Review + create](#) [< Previous](#) [Next: Encryption >](#)

Step 3: Here, enter the Encryption details, if any, and then click on Review + create

[All services](#) > [Container registries](#) > Create container registry

 **Create container registry**

[Basics](#) * [Encryption](#) [Tags](#) [Review + create](#)

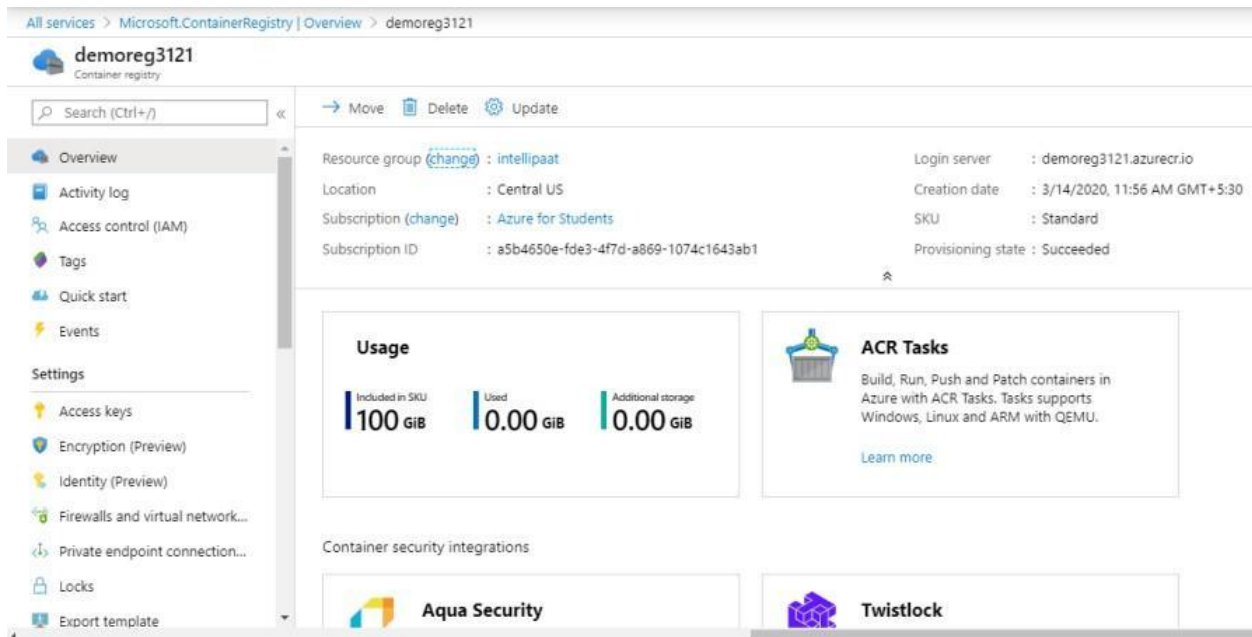
Azure Container Registry service encryption protects your data at rest. Azure Container Registry encrypts your images and other artifacts when they're pushed to your registry and automatically decrypts when you pull them. [Learn more](#)

Customer-Managed Key ☐ Enabled ☒ Disabled

Customer-Managed Key is only available for Premium SKU.

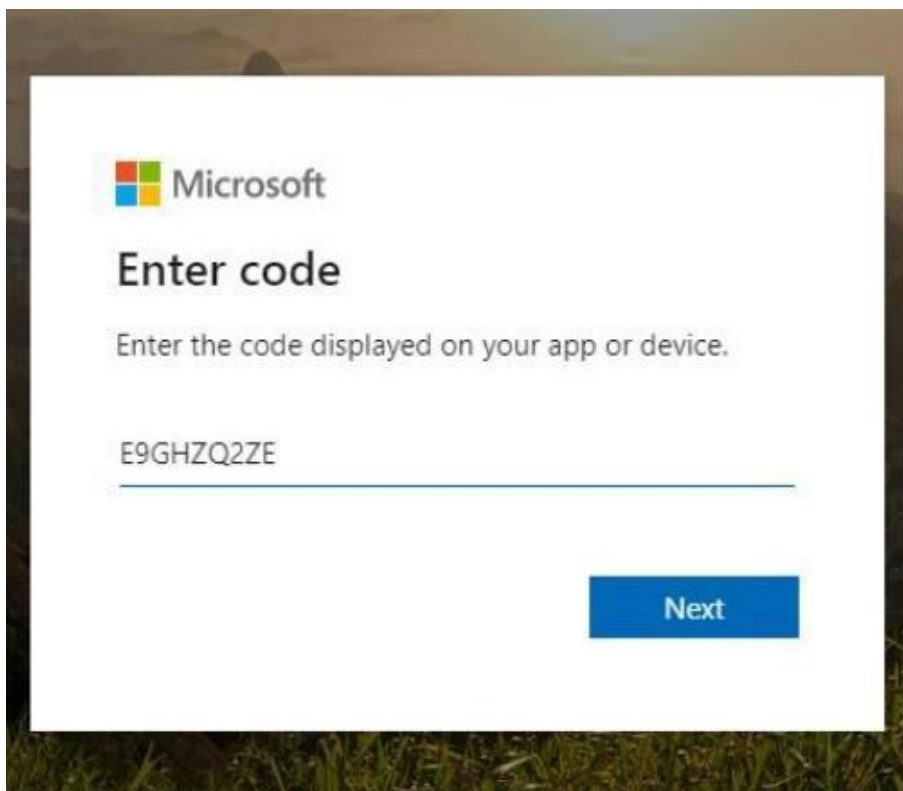
[Review + create](#) [< Previous](#) [Next: Tags >](#)

Step 4: Once the resource has been deployed, copy the name of the container registry



Step 5: Go to the Ubuntu machine and type the command `az login` to log into Azure Portal. It will provide you with a code to authenticate

```
root@vml:/home/newuser# az login
To sign in, use a web browser to open the page https://microsoft.com/devicelogin
and enter the code E9GHZQ2ZE to authenticate.
root@vml:/home/newuser#
```

Step 6: Enter the code to authenticate**Step 7:** Connect to the Azure container registry with the command:

```
az acr login --name <name of the container registry>
```

```
root@vml:/home/newuser# az acr login --name demoreg3121
Login Succeeded
root@vml:/home/newuser#
```

Step 8: Pull an image 'hello-world' using the command `docker pull hello-world` and tag it using the `docker tag` command:

```
docker tag <image name><destination address of the image>
```

Now, follow these commands to push the image into the container registry and run it

```
> docker push demoreg3121.azurecr.io/hello-world:v1
```

```
> docker run demoreg3121.azurecr.io/hello-world:v1
```

Step 9: Go to the Azure Container Registry in Azure Portal and click on Repositories. Refresh it to be able to see the repositories that have been pushed by you

