

Day 1 Activity File: Build, Host, and Design Your Web Application Using a GoDaddy Low-Cost Domain

Today you will build, host, and design your own web application. Specifically, you will:

- (1) **Create an Azure web app.**
- (2a) **Choose a domain with GoDaddy.**
- (2b) **Map your custom domain to Azure's App Service.**
- (3) **Deploy a container on the web app.**
- (4) **Design your custom web application.**
- (5) **Answer review questions.**

Resources

- [Azure App Service Documentation](#)
 - [How to come up with a good domain name](#)
 - [Mapping existing DNS to Azure App Service](#)
 - If Microsoft Support is needed, visit [How to open a support ticket](#)
 - [Split-Half Search](#)
 - [Top CyberSecurity Blog Websites](#)
-

Getting Started/Prerequisites

Before you begin, you are required to have completed the following tasks from the Cloud unit:

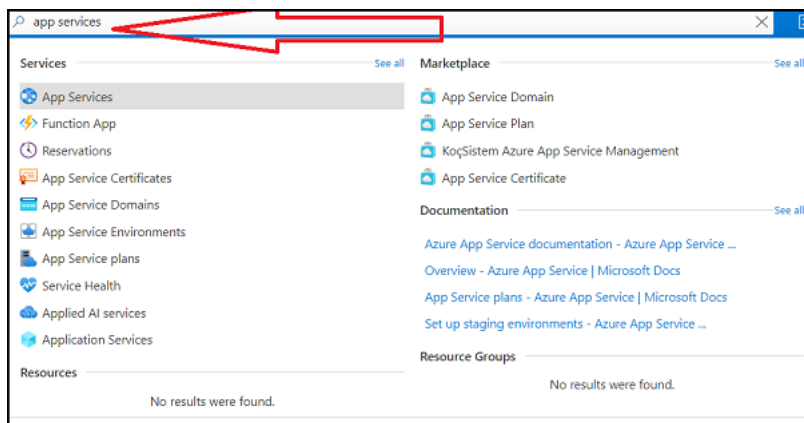
- Created your own Azure Account (not @Cyberxsecurity).
- Created a subscription.
- Created a resource group (RedTeam was recommended during the Cloud unit).

Instructions

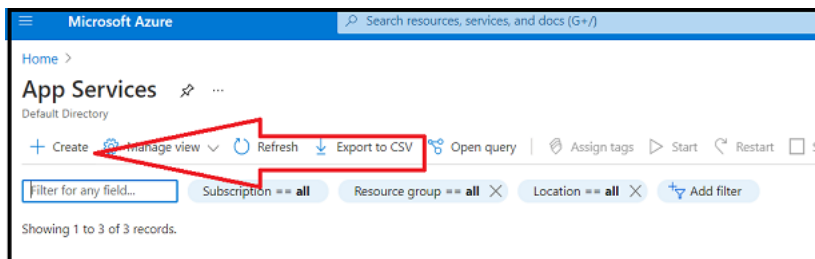
Part 1: Create an Azure Web App

In Part 1 of this activity, you will create your own Azure web application. You will name your application instance and select your back-end code and service plan. To do so, complete the following steps:

1. Begin by logging in to the Azure portal: <https://portal.azure.com>.
 - Make sure that you're logged in to your personal Azure account (not @Cyberxsecurity), where your Cloud Security-unit VMs are located.
2. Select "App Services" from the Azure search field at the top of the page, as the following image shows:



3. Select "+ Create" to create your application, as the following image shows:



4. Under the "Basics" tab, make the following selections:

- Subscription/Resource Group: Select the same subscription and resource group that you used during Cloud week.
- Name: Name your instance as you see fit; note that this will be the name of the Azure app.
 - For example: "Bobssecurityresume"
- Publish: Select "Code."
- Runtime Stack: Select "PHP 7.4."
- Operating System: Select "Linux."
- Region: Select the same region that you used during Cloud week.

The following image shows the completed "Basics" tab:

Home > App Services >

Create Web App

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * ⓘ Azure subscription 1

Resource Group * ⓘ

[Create new](#)

Instance Details

Name * bobssecurityblog ✓
.azurewebsites.net

Publish * ☒ Code ☐ Docker Container

Runtime stack * PHP 7.4

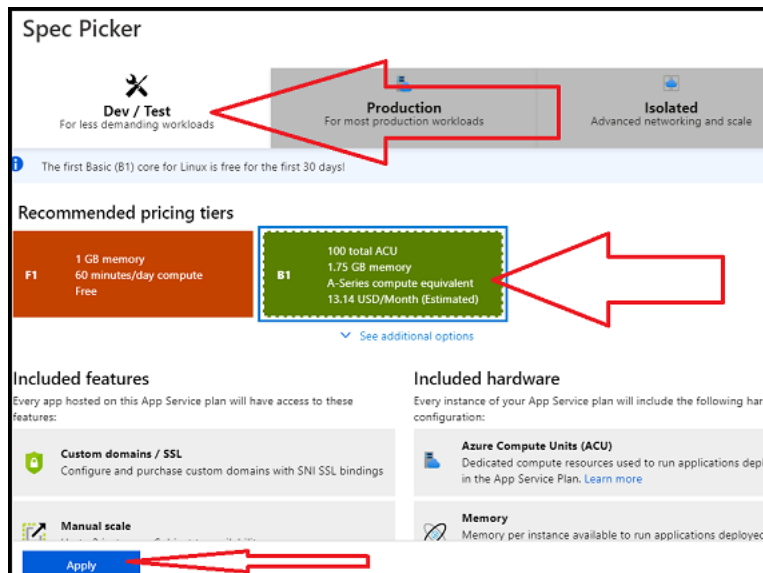
Operating System * ☒ Linux ☐ Windows

Region * Central US

[i](#) Not finding your App Service Plan? Try a different region.

5. For the App Service Plan, complete the following steps:

- Under "Linux Plan," select "Create New" and then enter "project1plan."
- Under "Sku and size," select "Change size."
- The spec picker will pop up on the right-hand side of your screen.
 - Note that this allows you to choose the pricing structure of your web app.
 - Select "Dev/Test" and "Plan B1" (the green option), and then click "Apply," as the following image shows:



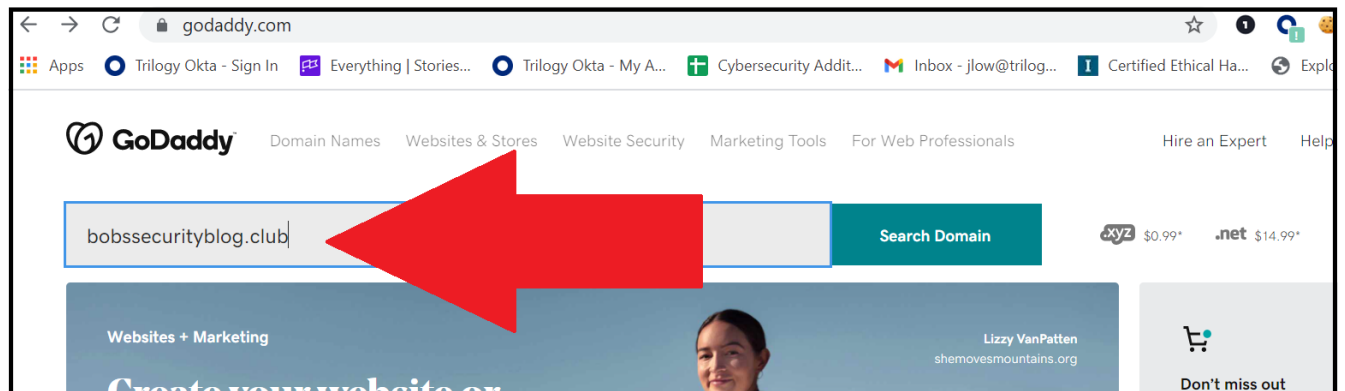
6. Leave the default options for all of the other tabs. Select the "Review + Create" tab.

7. Select "Create" at the bottom of the screen to create your web app.

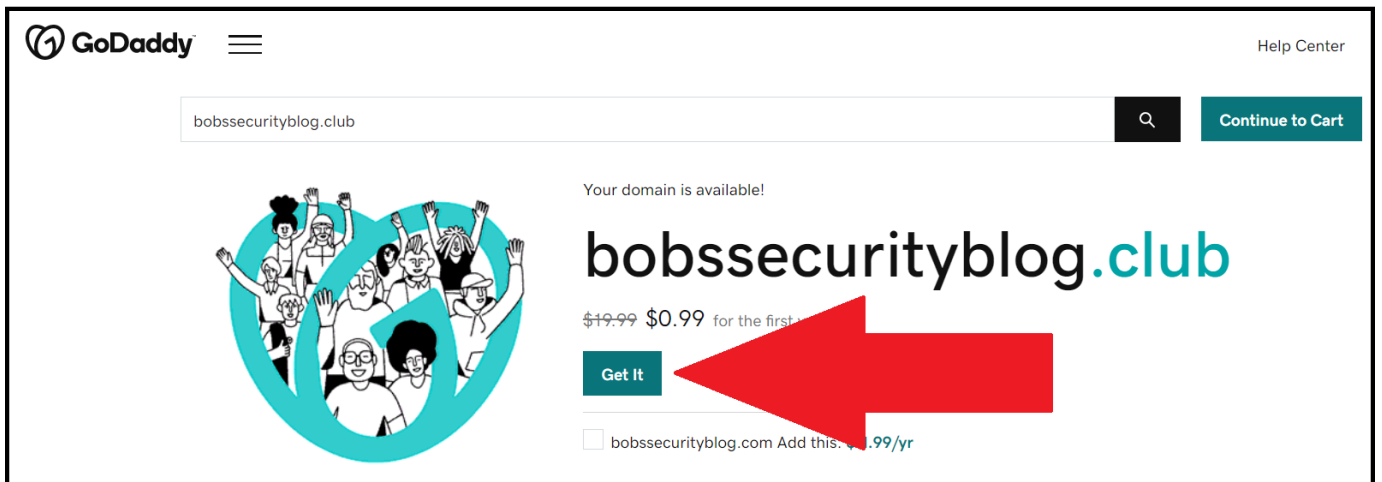
Part 2a: Choose a Domain with GoDaddy

In this part, you will select your own unique domain name using GoDaddy. To do so, complete the following steps:

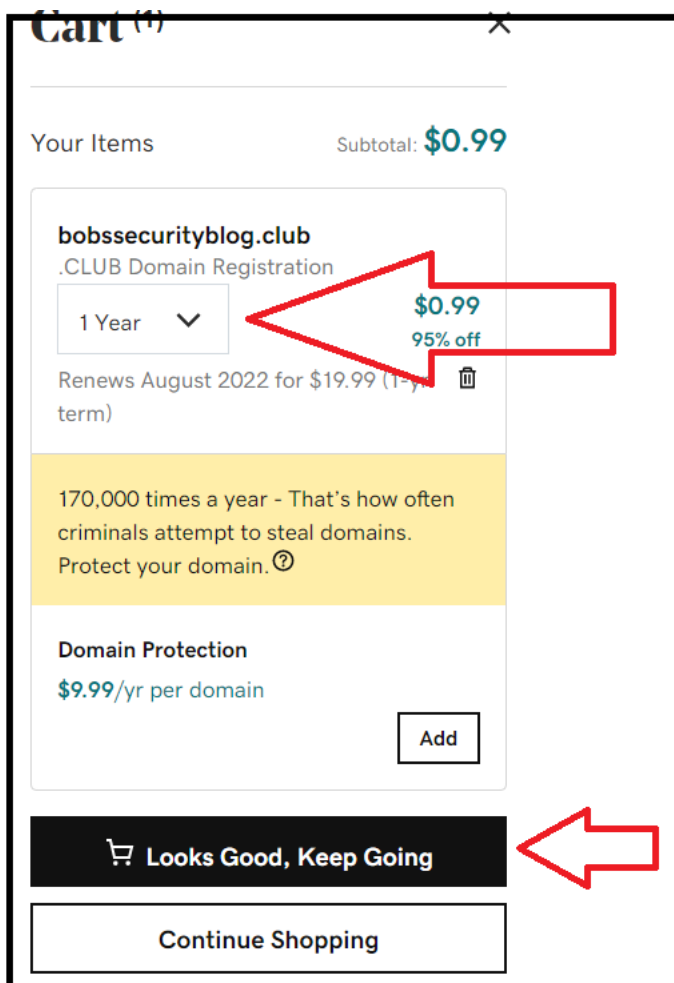
1. Navigate to [GoDaddy.com](https://www.godaddy.com).
2. Pick a website for your cyber-blog, but note the following:
 - The 99 cent domains are usually ".club" or ".xyz," and ".info" is \$1.99 a year.
 - After you pick the domain, the cost may increase after your first year—check carefully for this.
 - We recommend choosing a domain such as "bobssecurityblog.club," but you can pick any domain that you prefer:



- Refer to the following webpage for tips on picking a domain: [How to come up with a good domain name](#).
 - GoDaddy will provide you other domain options if your chosen domain is not available.
3. Once you have chosen an available domain, click "Get It," as the following image shows:



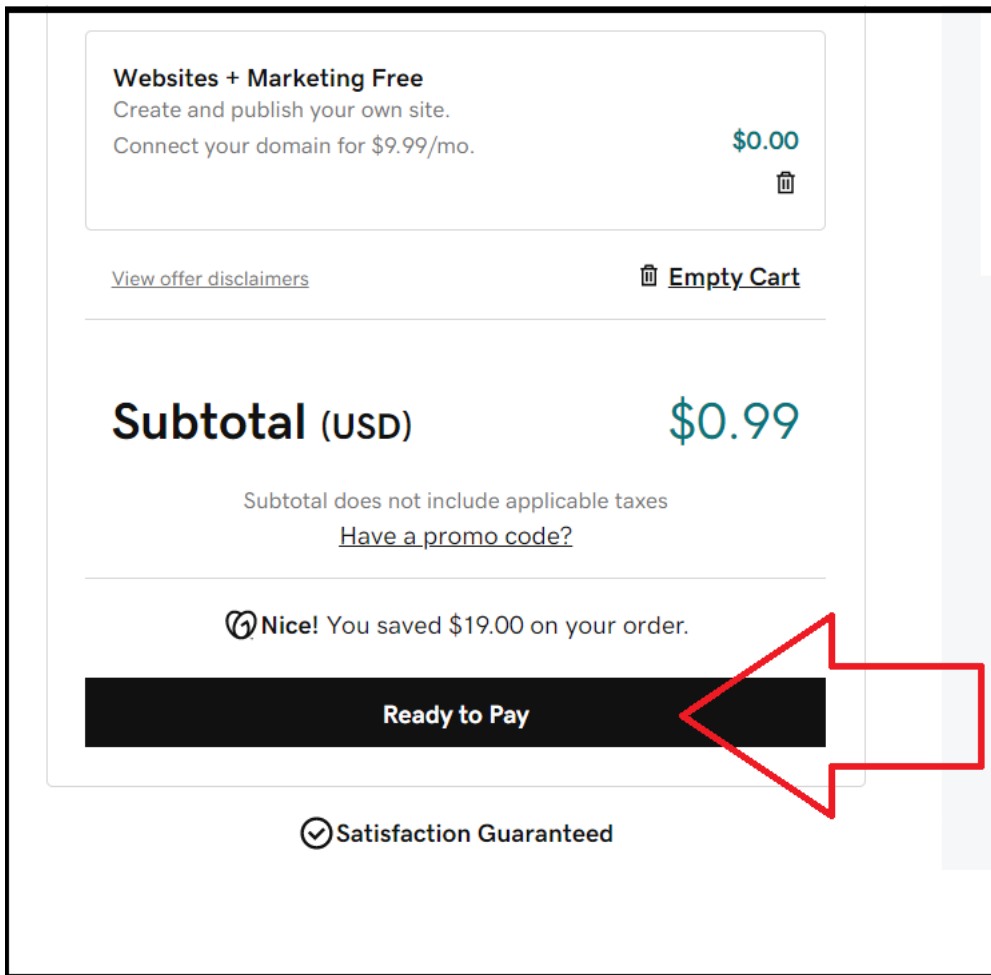
4. Select a duration of one year to get the 99 cent price, then select "Looks Good, Keep Going," as the following image shows:



5. The next page offers add-ons, such as domain protection. These are not required. Select "Continue to cart."

6. The next page should show your cart. Make sure that your total shows correctly (99 cents if you didn't select any add-ons).

- Then click "Ready to Pay," as the following image shows:



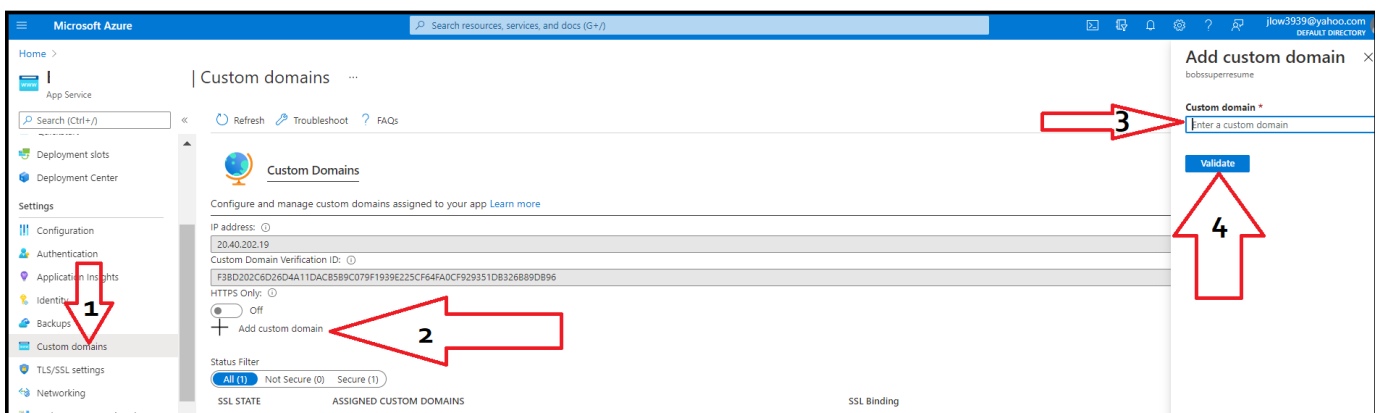
7. Create a GoDaddy account, or log in if you already have an account.

8. Follow the steps to complete your payment.

Part 2b: Map Your Custom Domain to Azure's App Service

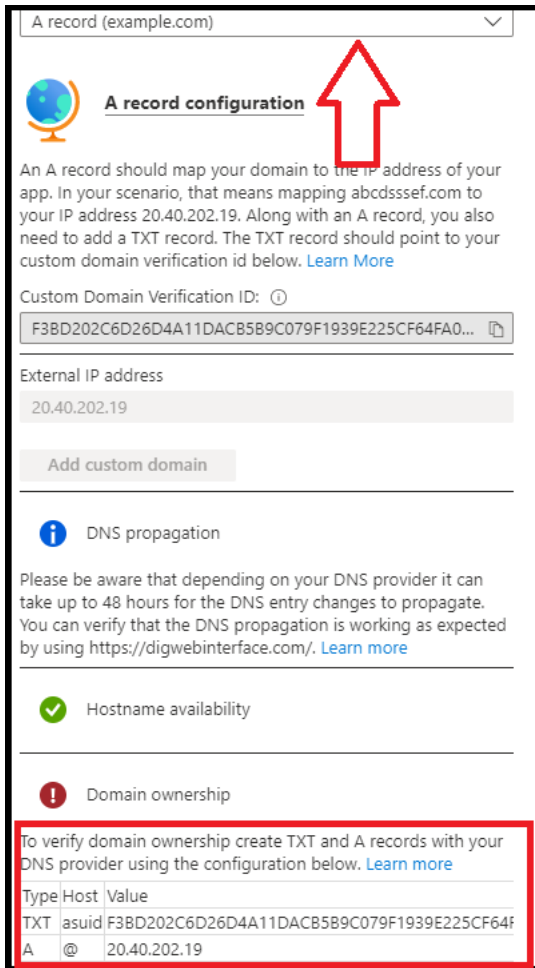
In this part, you will point the DNS to Azure's app service. To do so, complete the following steps:

1. Return to the app that you created within Azure App Service.
2. Select "Custom domains" from the left-hand menu, then click "Add custom domain." In the "Custom domain" text box, enter the new domain that you created on GoDaddy, and then click "Validate." The following image shows these steps:



3. After selecting "Validate," the page will confirm whether the hostname is available.

- Select the hostname record type "A Record," and notice the TXT and A data under "Domain ownership," as shown in the following image:



A record (example.com) ▾

A record configuration

An A record should map your domain to the IP address of your app. In your scenario, that means mapping abcdsssef.com to your IP address 20.40.202.19. Along with an A record, you also need to add a TXT record. The TXT record should point to your custom domain verification id below. [Learn More](#)

Custom Domain Verification ID: ⓘ
F3BD202C6D26D4A11DACB5B9C079F1939E225CF64FA0...

External IP address
20.40.202.19

[Add custom domain](#)

i DNS propagation

Please be aware that depending on your DNS provider it can take up to 48 hours for the DNS entry changes to propagate. You can verify that the DNS propagation is working as expected by using <https://digwebinterface.com/>. [Learn more](#)

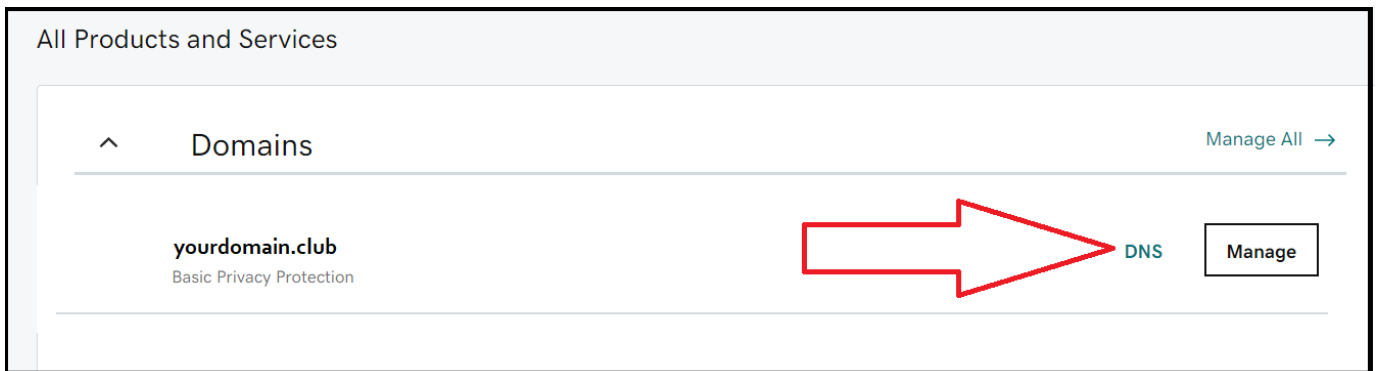
✓ Hostname availability

! Domain ownership

To verify domain ownership create TXT and A records with your DNS provider using the configuration below. [Learn more](#)

Type	Host	Value
TXT	asuid	F3BD202C6D26D4A11DACB5B9C079F1939E225CF64f
A	@	20.40.202.19




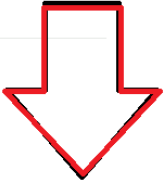
- You will update this data in the GoDaddy DNS record to prove that you own the domain.
4. Return to your GoDaddy.com products page: <https://account.godaddy.com/products>.
 5. On this page, find the domain that you just added (you may have to scroll down), and select "DNS," as the following image shows:



6. Below your existing DNS records, select "ADD," as the following image shows:

Records

Last updated 11/15/2019 10:40 AM

Type	Name	Value	TTL	
A	@	Parked	600 seconds	
CNAME	www	@	1 Hour	
CNAME	_domainconnect	_domainconnect.gd.domaincontrol.com	1 Hour	
NS	@	ns25.domaincontrol.com	1 Hour	
NS	@	ns26.domaincontrol.com	1 Hour	
				
				ADD

7. One at a time, add in the two "Domain ownership" records (TXT and A), that you saw in Step 3, and click "Save" after each one.
- The TTL can stay at 1 hour.

The following image shows this step:

Type *

TXT

Host *

asuid

TXT Value *

F3BD202C6D26D4A11DACB5B9C079F1939E...



TTL *

1 Hour

Save

Cancel

8. Once you've added both records, they should appear at the bottom of your DNS page, as the following image shows:

TXT	asuid	F3BD202C6D26D4A11DACB5B9C079F1939E...	1 Hour	
A	@	20.40.202.19	1 Hour	
				ADD

9. Delete the A record that is labeled as "Parked"
10. Return to Azure, and repeat Step 2. After clicking "Validate," green check marks should appear next to "Hostname availability" and "Domain ownership."
11. Select "Add custom domain," as the following image shows:

Validate

Hostname record type

A record (example.com)

A record configuration

An A record should map your domain to the IP address of your a In your scenario, that means mapping ladylow.com to your IP address 20.40.202.19. Along with an A record, you also need to a a TXT record. The TXT record should point to your custom doma verification id below. [Learn More](#)

Custom Domain Verification ID: ⓘ

F3BD202C6D26D4A11DACB5B9C079F1939E225CF64FA0CF92...

External IP address

20.40.202.19

Add custom domain

✓

Hostname availability

✓

Domain ownership

12. After a minute, your new domain will appear under "Assigned Custom Domains," as the following image shows:

Deployment

Quickstart

Deployment slots

Deployment Center

Settings

Configuration

Authentication

Application Insights

Identity

Backups

Custom domains

TLS/SSL settings

Networking

Custom Domains

Configure and manage custom domains assigned to your app [Learn more](#)

IP address: ⓘ

20.40.202.19

Custom Domain Verification ID: ⓘ

F3BD202C6D26D4A11DACB5B9C079F1939E225CF64FA0CF929351DB326B89DB96

HTTPS Only: ⓘ

On

+ Add custom domain

Status Filter

All (2)

Not Secure (1)

Secure (1)

SSL STATE	ASSIGNED CUSTOM DOMAINS	SSL Binding
Not Secure	New domain will appear here	Add binding

- If the domain doesn't appear, refresh your page and clear your cache.

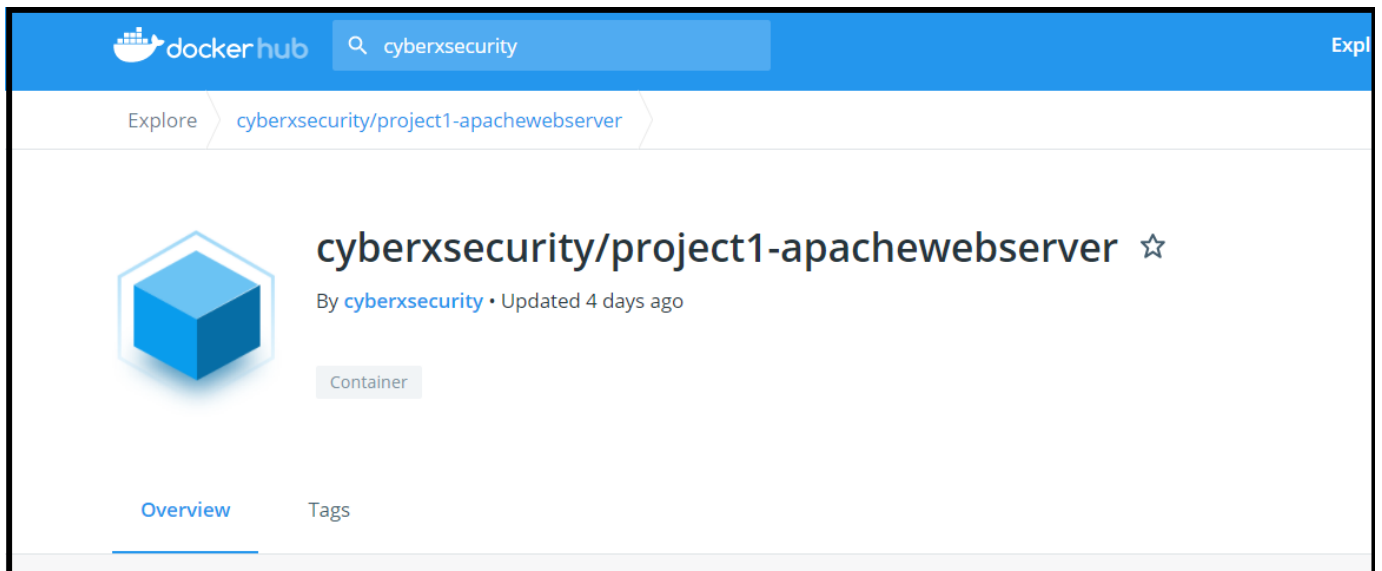
Congratulations! You now own your own unique domain, accessible on the internet!

:warning: Checkpoint :warning:
Before continuing, make sure that you have completed the following critical tasks:
:heavycheckmark: Your Azure web app has been created.
:heavycheckmark: A unique IP has been assigned to your web app.
:heavycheckmark: You have selected and created your own unique domain with GoDaddy.
:heavycheckmark: You have mapped your domain to Azure's app service.

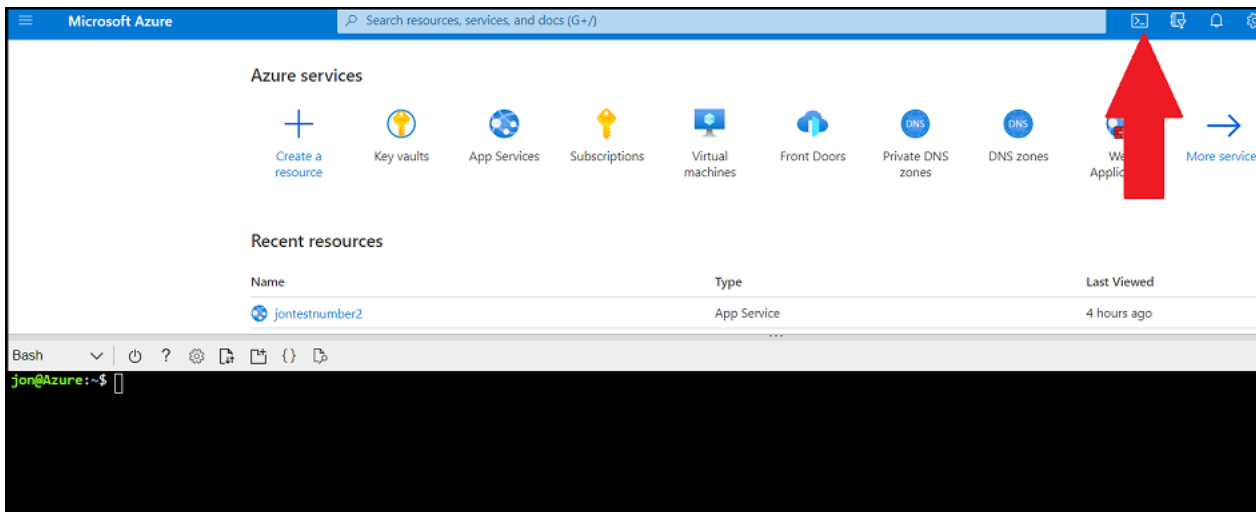
Part 3: Deploy a Container on the Web App

In Part 3, you will use the Azure Cloud Shell to deploy a Docker container on your web application. This container contains the framework for your cyber blog webpage.

1. For your web application, you will use a Docker container that has been added to Docker Hub. View the Docker container at the following location: [Cyber Blog Framework - Docker Container](#).
2. Note that the Docker container image name is `cyberxsecurity/project1-apachewebserver`, as the following image shows:



3. Next, you will use the Azure Cloud Shell to deploy this container to your web application.
 - Azure Cloud Shell takes user input from a command line to manage Azure's cloud resources.
 - While we will use Bash, you can also use Powershell to administer your commands.
 - For additional resources on Azure's Cloud Shell, refer to the following pages:
 - [Azure Cloud Shell Overview](#)
 - [Azure Web App Container commands](#)
 - To open Azure Cloud Shell, click the shell logo in the tool bar at the top of the screen, as indicated by the red arrow in the following image:



- Once you've clicked this icon, the Cloud Shell will be accessible at the bottom of your page.
 - When using Shell, you may receive the following prompts:
 - Select which shell to use (Bash or Powershell): Select "Bash."
 - Create Storage: If a window appears, select "Create Storage."
4. Next, from the command line, you'll enter a command to configure your container.
 - There are three types of commands that manage your web app container settings:

- (1) `az webapp config container delete` - This will delete your web app container's settings.
- (2) `az webapp config container set` - This will set your web app container's settings.
- (3) `az webapp config container show` - This will display the current details of your web app container's settings.

- To configure your web app with your provided container, run the following:

```
az webapp config container set --name <name of your webapp> --resource-group <name of your resource group> --docker-custom-image-name <docker custom image name>
```

- For example:

```
az webapp config container set --name bobswebapp --resource-group redteamRG --docker-custom-image-name cyberxsecurity/project1-apachewebserver
```

- After pressing enter, an output similar to the image below should appear:

```
[
  {
    "name": "WEBSITES_ENABLE_APP_SERVICE_STORAGE",
    "slotSetting": false,
    "value": "true"
  },
  {
    "name": "DOCKER_CUSTOM_IMAGE_NAME",
    "value": "DOCKER|cyberxsecurity/project1-apachewebserver"
  }
]
```

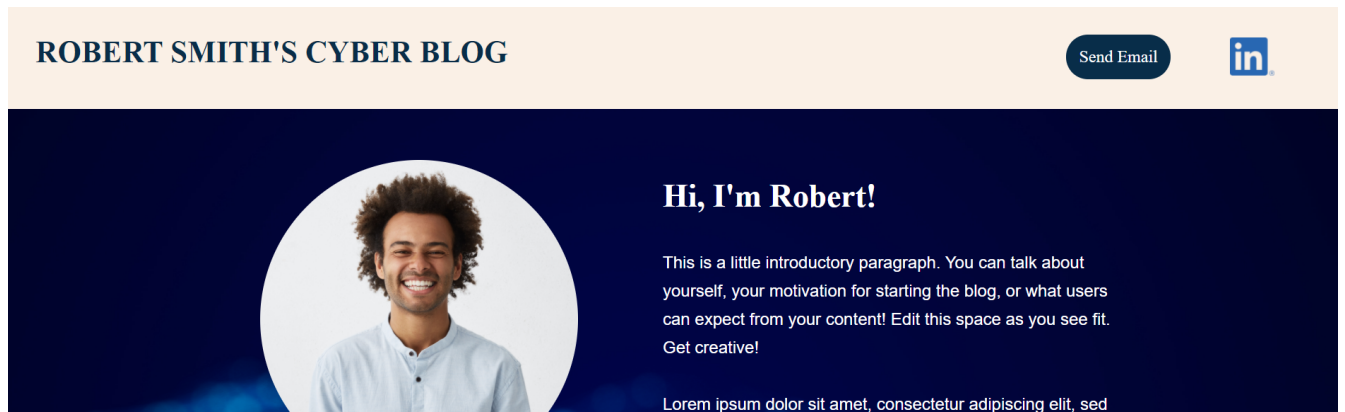
5. To verify that the container has been added correctly, run the following command to **show** the container for your web app:

```
az webapp config container show --name <name of webapp> --resource-group <name of your resource group>
```

- For example: `az webapp config container show --name bobswebapp --resource-group redteamRG`

6. Now, check the unique domain that you selected to verify that the container has been successfully deployed.

- A cyber blog webpage that looks like the following image should appear (note that it may take five to eight minutes to load):



Now, you are ready to customize your web application!

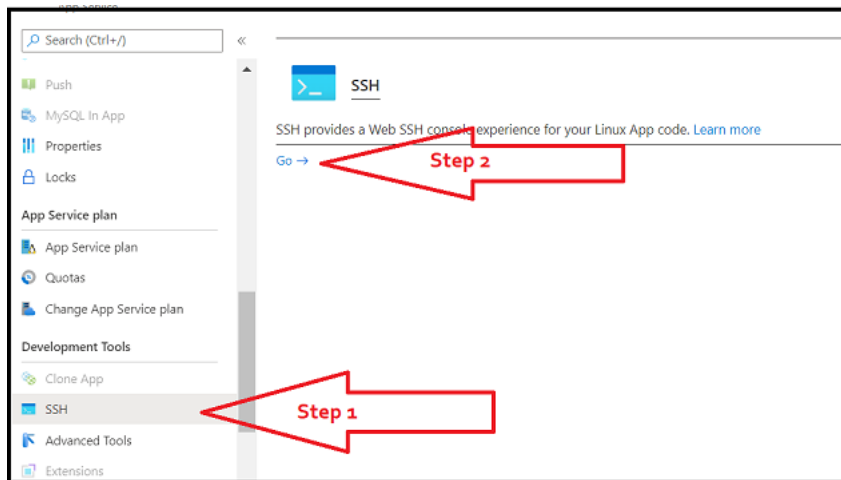
Part 4: Design Your Custom Web Application

The container that you just loaded onto your web application is a framework for a cyber-blog page that you can customize.

You will now customize the following elements of the webpage: - Your name - Your email - Your LinkedIn profile link - Your introduction - Your picture - Two custom blog posts on topics of your choice

1. To design and customize your webpage, you'll need to access the HTML pages of your new web application.

- To access these pages, you need to SSH over to your container and access the HTML files.
- Return to your web app in Azure, select "SSH" from the left-hand toolbar, and then select "GO," as shown in the following image:



2. This will SSH you right into the container.

- Once you have access, change directories to the location where the HTML files are located by running `cd /var/www/html`, as the following image shows:

```
Last login: Tue Aug  3 17:43:09 2021 from 172.16.0.2

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@66ef03f74935:~# cd /var/www/html
```

3. This directory contains the `index.html` file that makes up your webpage. To customize your webpage, complete the following steps:

- To change your name:
 - Run: `nano index.html`
 - Replace "ROBERT SMITH'S CYBER BLOG" with your name/text.
 - Replace "Hi, I'm Robert!" with your name/text.
- To change your email:
 - In the same `index.html` file, replace "aaggarwal@2u.com" with your email address.
- To change your LinkedIn profile link:
 - In the same `index.html` file, replace "https://www.linkedin.com/" with the link to your LinkedIn profile.
- To change your introduction:
 - In the same `index.html` file, replace the paragraph beginning "This is a little introductory paragraph" with your own introduction.
- To change your picture, follow these [instructions](#).
 - Note that if you prefer not to use a photo of yourself, you can replace it with a stock photo. To do so, replace `` with this: ``

4. Next, write and edit two blog posts on any cybersecurity topics of your choice. (Note: you can continue writing outside of class time if necessary.)

- Each blog post should be 10 sentences minimum.
- For inspiration, review the following cyber blogs (but do not copy directly from these or other blogs):
 - <https://www.troyhunt.com/>
 - <https://www.lastwatchdog.com/>
 - <https://krebsonsecurity.com/>
- You can select any topic to write about from any of the domains that we have covered, including the following:

- GRC, networking, network security, cloud, cryptography
- Feel free to use online resources to help you research and write your blog posts.

◦ Here are some possible ideas that you could use for blog topics:

- ***Ransomware: Should organizations pay or not?***
- ***Who should have the final say on product security decisions, the business or the security department?***
- ***Are humans really the weakest link in security?***
- ***How could quantum computing affect cybersecurity?***
- ***Should organizations try to utilize open source security software?***

5. Once you've written your blog posts, add your posts to your cyber blog webpage by completing the following steps:

◦ Blog Topic 1

- Change "Blog Post 1 Title" to the title of your first blog post.
- Change "Add Keywords" to relevant keywords for your post (e.g., ransomware, cryptography).
- Change the section beginning "Add a short description here" to the text of your blog post.

◦ Blog Topic 2

- Change "Blog Post 2 Title" to the title of your second blog post.
- Change "Add Keywords" to relevant keywords for your post (e.g., ransomware, cryptography).
- Change the section beginning "Add a short description here" to the text of your blog post.

⚠ Important - Backing up your HTML ⚠

- Restarting your virtual machine will often clear out any updates to your HTML files. Therefore, it is important to back them up everytime you make an update!
- After each update to your webpage, run the following command to backup your index.html file to your /home directory, which stays persistent across reboots.

◦ `cp /var/www/html/index.html /home`

- In case you need to restore your `index.html` file, run the following command:

◦ `cp /home/index.html /var/www/html/`

After you have made your changes and backed them up, return to your browser and refresh your webpage. - Congratulations, you now have your own cloud-hosted web blog!

Part 5: Answer Review Questions

- Open up the [review questions](#), make a copy of the document, and answer the Day 1 review questions.
- Note that you will submit this document as one of your deliverables at the end of the project.

Day 1 Milestone

In today's class, you:

- (1) - **Created an Azure web app.**
- (2a) - **Chose a domain.**
- (2b) - **Mapped your custom domain to Azure's app service.**
- (3) - **Deployed a container on the web app.**
- (4) - **Designed your custom web application.**
- (5) - **Answered review questions.**

Completing these steps required you to leverage your terminal, systems administration, cloud, and automation skills. This is an impressive set of tools to have in your toolkit!