Course 3_Lab 3: Identify vulnerabilities and remediation techniques

Scenario

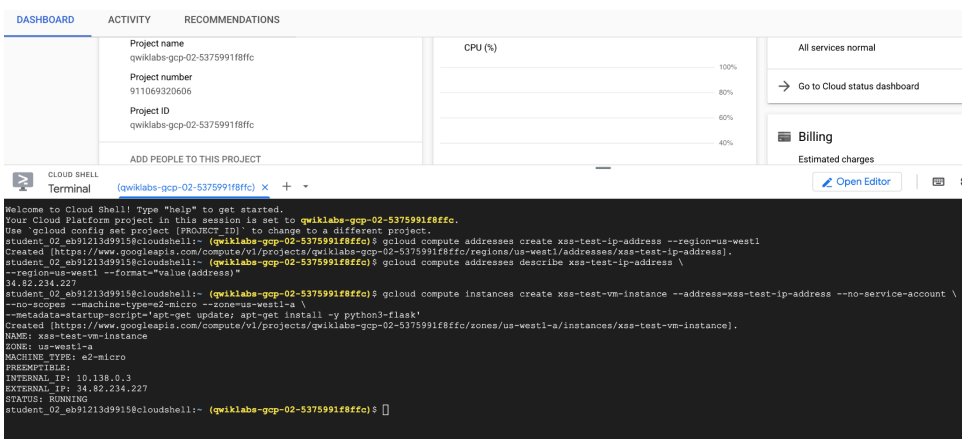
Cymbal Bank has developed a new banking application for its corporate clients that is set to be hosted and deployed on the new cloud infrastructure. The Chief Information Security Officer (CISO), Javier, wants to prioritize the security of this application before it is launched and customer-facing. Your team lead, Chloe, has tasked you with identifying and mitigating any application vulnerabilities for this new application. You'll use the Web Security Scanner in Google Cloud to scan the application for vulnerabilities pertaining to a top OWASP® web application vulnerability known as Cross-Site Scripting (XSS).

Here's how you'll do this task: **First**, you'll create a static IP address and launch a virtual machine. **Then**, you'll deploy the vulnerable application. **Next**, you'll set up and run the application. **Then**, you'll access and scan the application. **Finally**, you'll fix the vulnerabilities and re-scan the application.

MY WORK:

Task 1: Launch a virtual machine

In this task, I created a static IP (an IP address that doesn't change), in the us-east-1a region that will be used to scan the application. In addition, with the Google Cloud commands provided, I launched a VM (virtual machine), using Cloud Shell to run the vulnerable application.

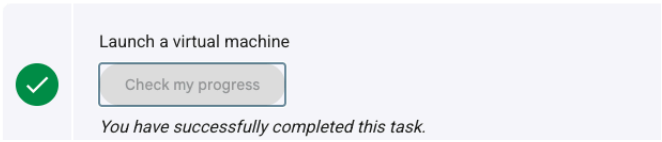


This command creates a VM instance to run the vulnerable application.

8. Press **ENTER**.

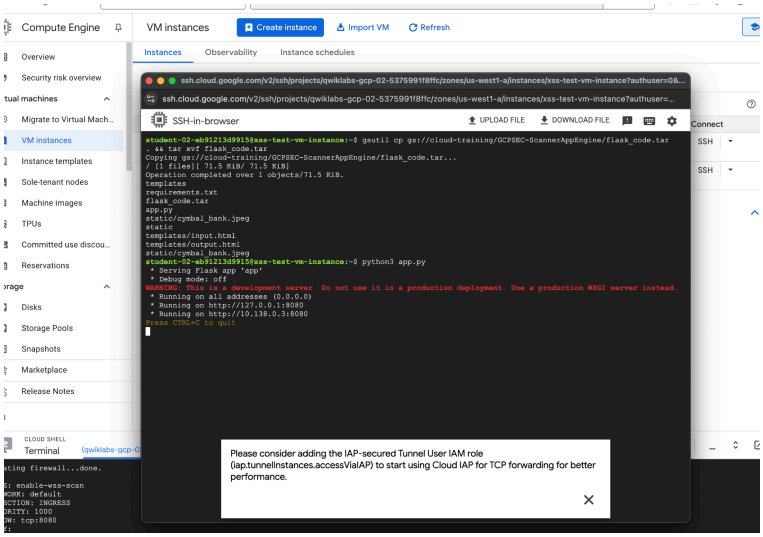
> ***Note:*** *The startup script will install* ***python-flask****, a* ***Web Application Framework****, which is used for running a simple Python application. This application demonstrates cross-site scripting (XSS) vulnerability, which is a common web application security vulnerability.*

Click **Check my progress** to verify that you have completed this task correctly.



Task 2: Set up and run the vulnerable application

In this task, I was able to create a firewall rule that would allow Web Security Scanner to access the vulnerable application. Once complete, I was able to open the application in a new browser tab by connecting to the VM instance from an SSH in-browser (pictured below).



Quick tip: Review the prerequisites before you run the lab

End Lab 01:16:35

Caution: When you are in the console, do not deviate from the lab instructions. Doing so may cause your account to be blocked. Learn more.

Open Google Cloud console

Google Cloud username

student-02-eb91213d9915

Google Cloud password

k0KJIiJgB7Xg

Google Cloud project ID

qwiklabs-gcp-02-5375991

10. Press **ENTER**.

A message should indicate that the application is up and running.

Note: Since this is a web application that was installed for use in development, there may be vulnerabilities associated with the configuration file. It is important to test any application prior use on a public facing network.

Note: *Do not close the* ***SSH-in-browser*** *page when performing the next task, as the application must continue to run.*

Click **Check my progress** to verify that you have completed this task correctly.

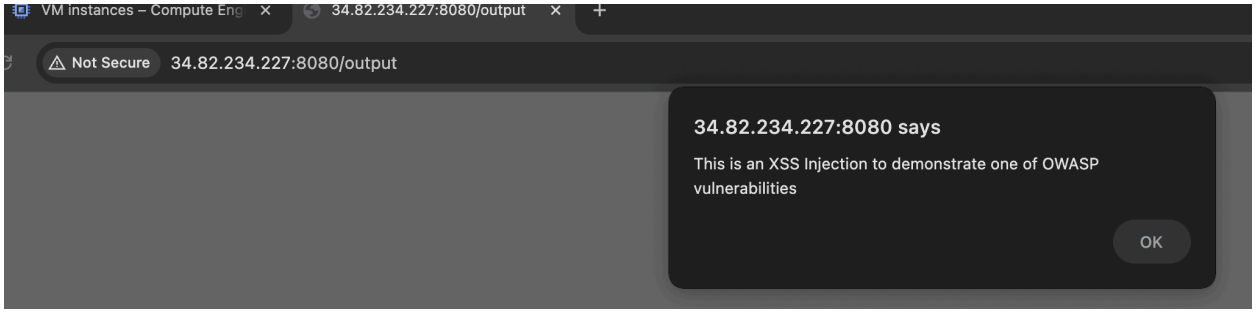
Set up and run the vulnerable application

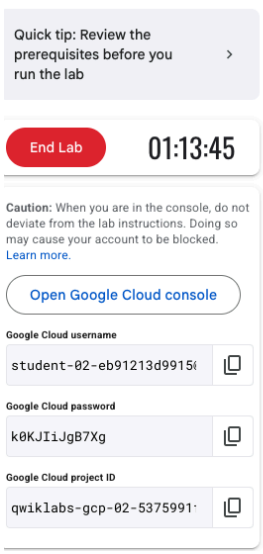
Check my progress

You have successfully completed this task.

Task 3: Access the vulnerable application

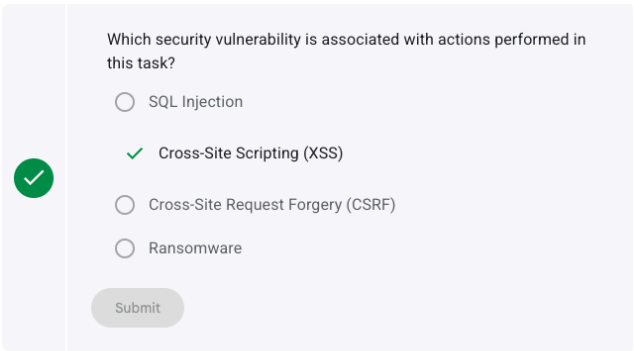
In this step, I was able to access the application with the XSS (cross-site scripting) vulnerability.





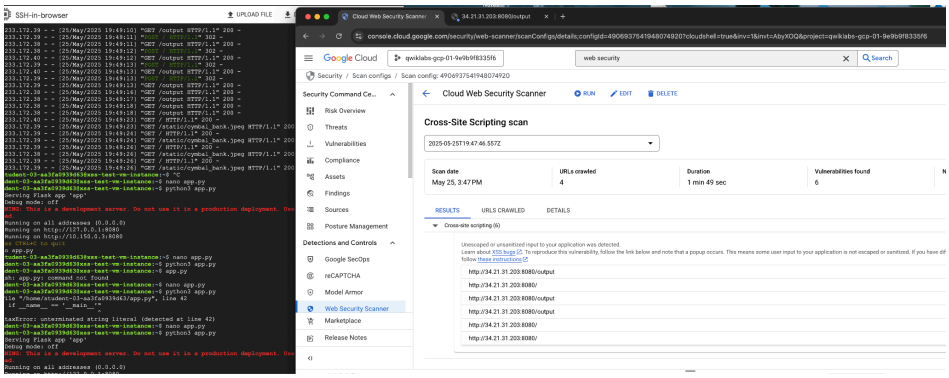
The injected code displayed a message back to the browser. This action by itself is not malicious, however attackers can introduce malicious code into an exploitable application to either steal data from it or implant malware onto the user's device.

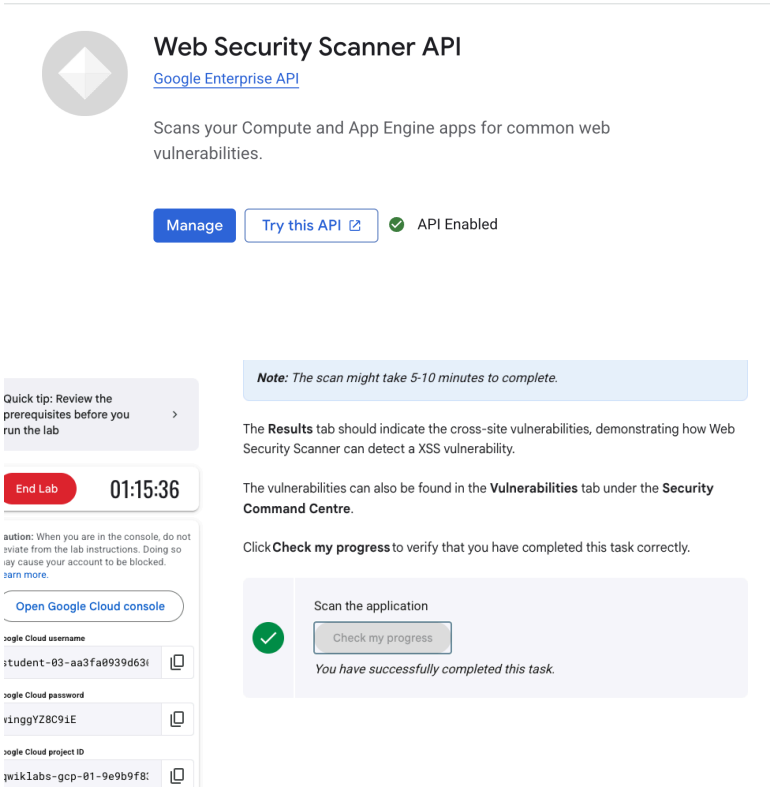
The alert window opens with the following message: "This is an XSS Injection to demonstrate one of OWASP vulnerabilities".



Task 4: Scan the application

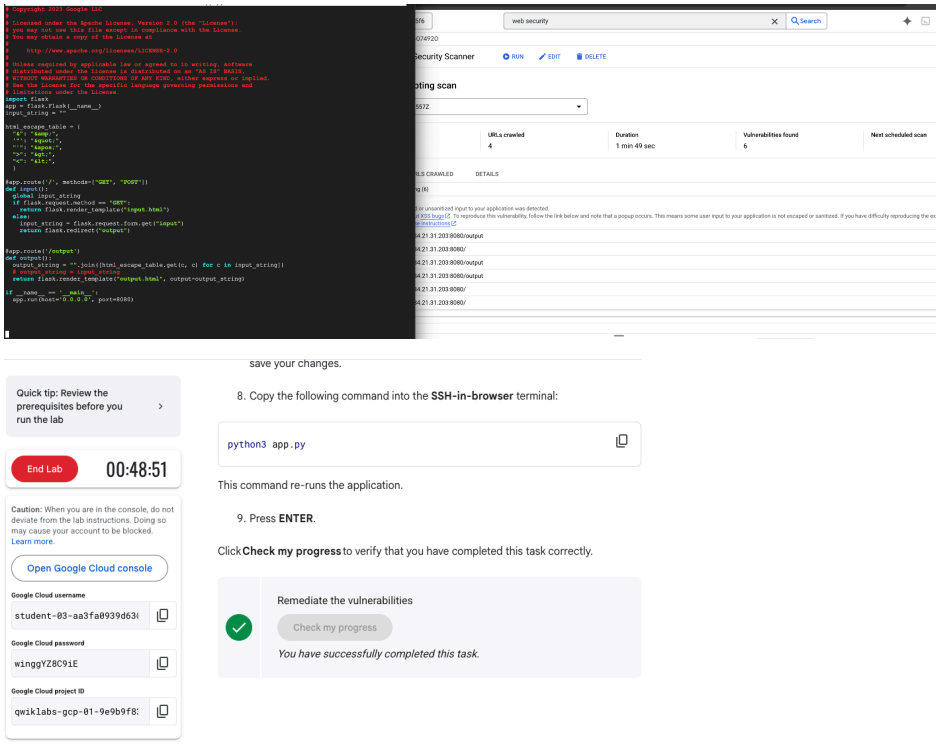
In this task, I was able to scan the application using Web Security Scanner to learn how the service can detect an XSS vulnerability, while also reviewing the results of the scan.





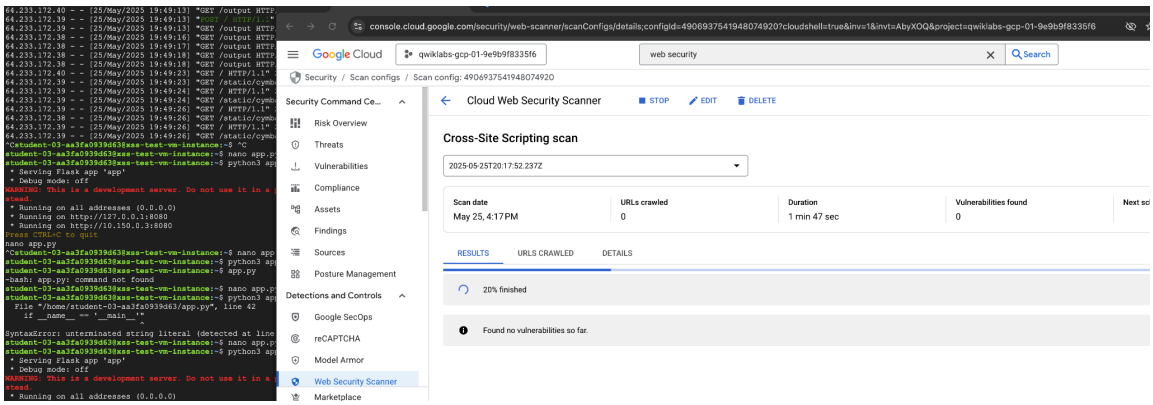
Task 5: Remediate the vulnerabilities

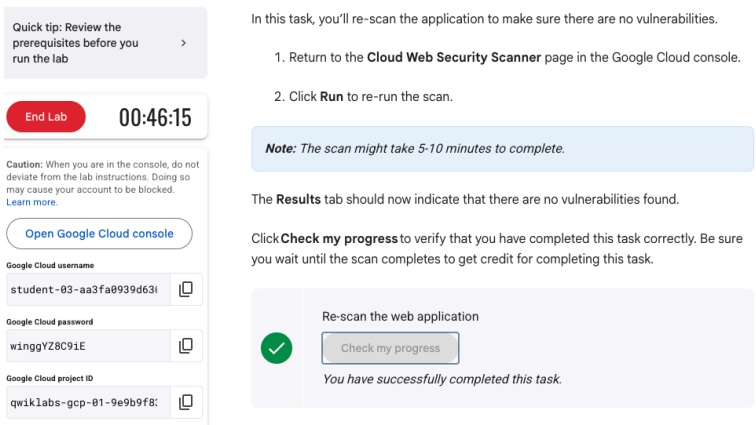
For this task, remediation was necessary to remove the XSS vulnerability. There was a simple change implemented in the python script, provided by Google. Once the change happened, I was able to rerun the application to verify remediation.



Task 6: Re-scan the web application

To verify the vulnerability was remediated, I was able to re-run the scan and learned no vulnerabilities were found.





My Assessment

The major highlight in this lab was the ability to see how cross-site scripting (XSS) works.. Google provided an HTML code that allowed the injection, while on the backend I was able to review the python script of the application to make the necessary changes and remediate this vulnerability. Enabling secure settings strengthens the security posture of the organization while also ensuring the organization is in compliance and protecting sensitive data.