|  |  |
| --- | --- |
| Name: | Pratyusha Ampolu |
| Lab User ID: | 23SEK3324\_U09 |
| Date: | 09-01-2024 |
| Application Name: | OWASP-wrong secrets |

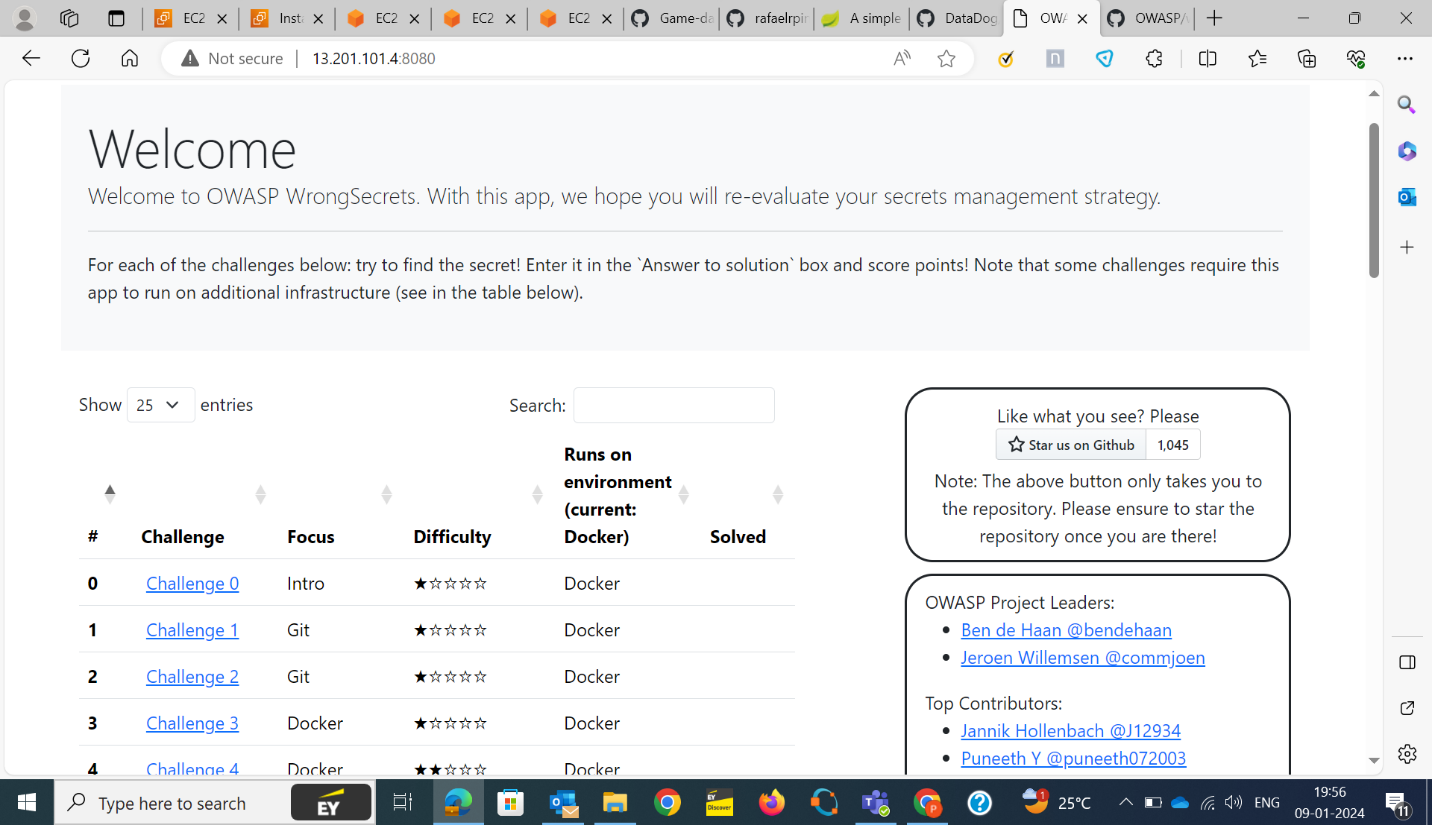
**Follow the below guidelines:**





System Architecture:

(Understand the system and document the physical and logical architecture of the system, use the shapes and icons to capture the system architecture)



My VM

My Docker

Browser

IP:8080

Docker container

jjj

Docker image

Docker image

kjj

k

­

Define system’s normal behavior:

(Define the steady state of the system is defined, thereby defining some measurable outputs which can indicate the system’s normal behavior)

* The web server starts listening on defined ports like IP address:8080.
* User can access a website hosted on this server via web browser.
* While opening the page, we can see the challenges and its state of difficulty.
* We can also able to see the environment where it is running.

Hypothesis:

(During an experiment, we need a hypothesis for comparing to a stable control group, and the same applies here too. If there is a reasonable expectation for a particular action according to which we will change the steady state of a system, then the first thing to do is to fix the system so that we accommodate for the action that will potentially have that effect on the system. For eg: "If one of our database servers fails, our service will automatically switch to a backup server, and users will not experience any downtime or data loss.")



* Hardcoded Passwords
* Exposed API keys
* Insecure storage of secrets
* Lack of encryption for secrets
* Automated tools may miss certain patterns.
* Effectiveness of existing detection tools.

**Known**

Things we are aware of but don’t understand.

Through load testing, engineers discover a particular microservice experiences significant latency only when multiple users concurrently access a specific feature.

* Unauthorized actions
* Data breaches
* Service disruption.

**Unknown**

**Unknown**

**Known**

Things we are neither aware of nor understand.

**Overview**: OWASP wrong secrets is an app started as a bad example app for a talk for AllDayDevops in 2020, "DevSecOps — Our Secret Management Journey from Code to Vault". How an organization handles its secrets reflects its security maturity.

Their goal is to educate people about secrets management and its pitfalls while they have a good time learning.

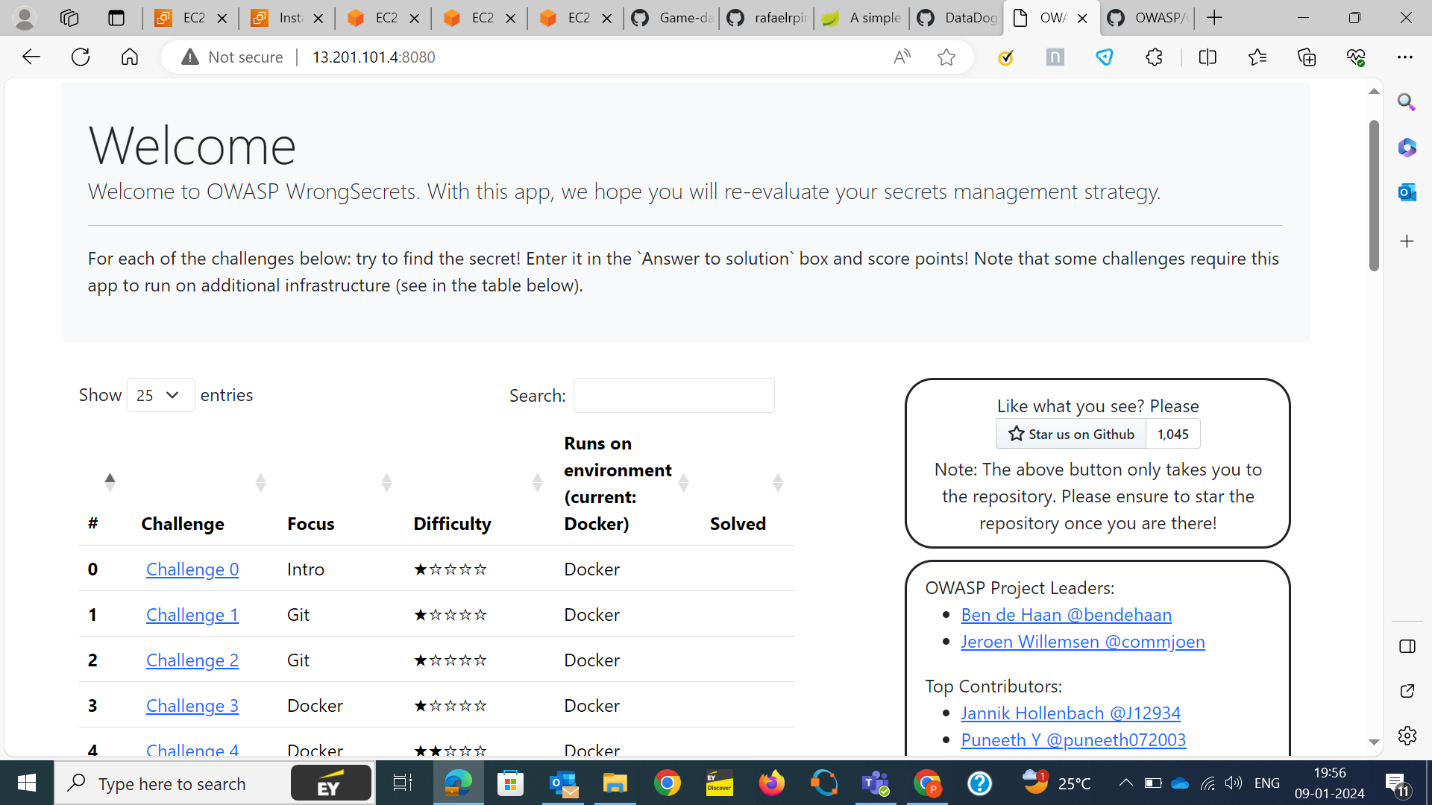
Step-1: installing docker in the VM.

**Cmd: apt install docker.io -y**

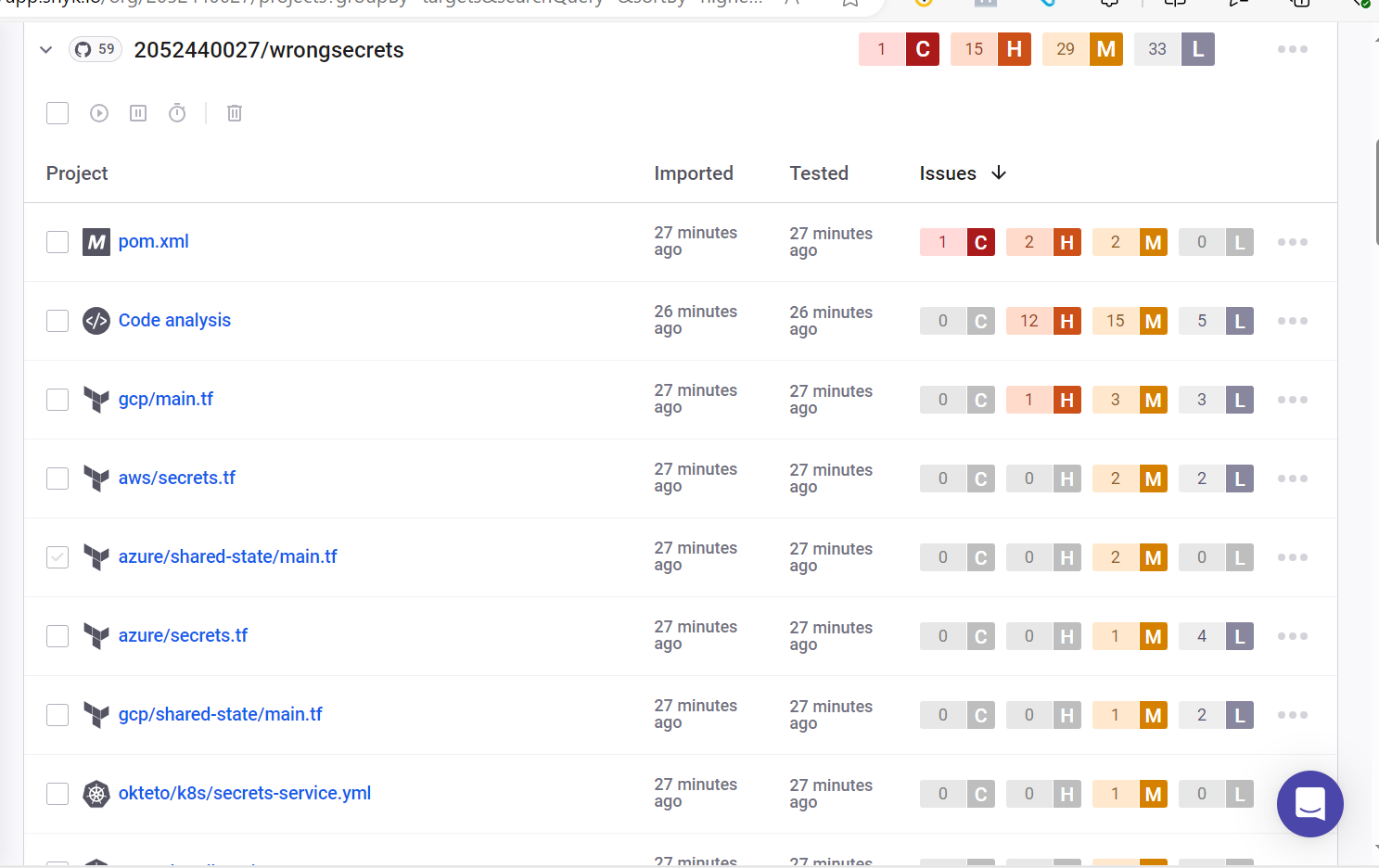
Step-2: Creating container by pulling the image jeroenwillemsen/wrongsecrets:latest-no-vault

**Cmd: docker run -p 8080:8080 jeroenwillemsen/wrongsecrets:latest-no-vault**

Step-3: Then check IP using the port 8080 to live the application.

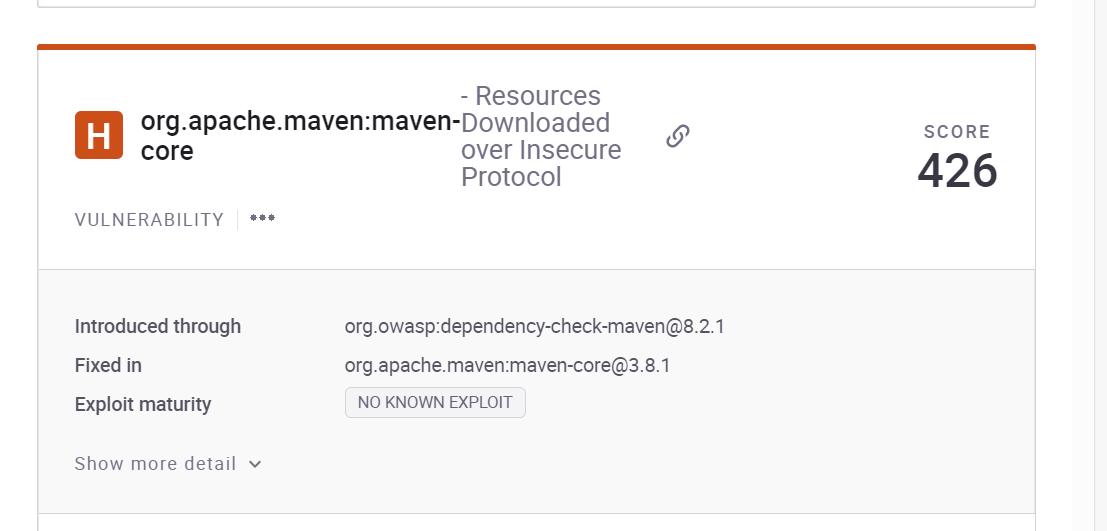


Tools: SYNK



A screenshot of a computer

Description automatically generated



**Issue**: Resources downloaded over insecure protocol.

**Solution**: A repository manager is a dedicated server application designed to manage repositories of binary components.

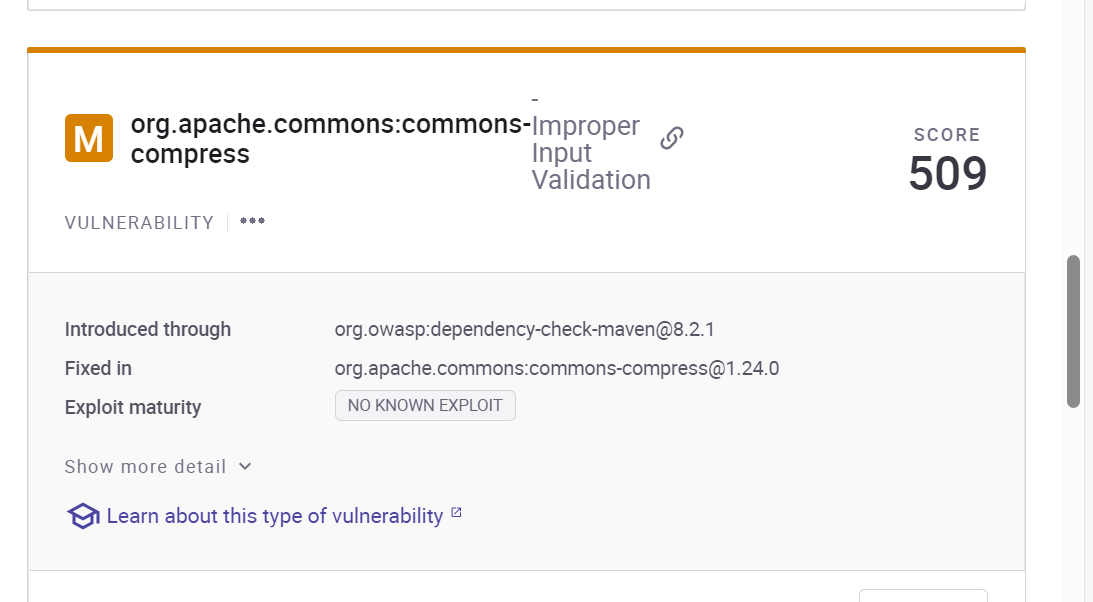
Using a repository manager provides the following benefits and features:

significantly reduced number of downloads off remote repositories, saving time and bandwidth resulting in increased build performance.

improved build stability due to reduced reliance on external repositories

increased performance for interaction with remote SNAPSHOT repositories

potential for control of consumed and provided artifacts.

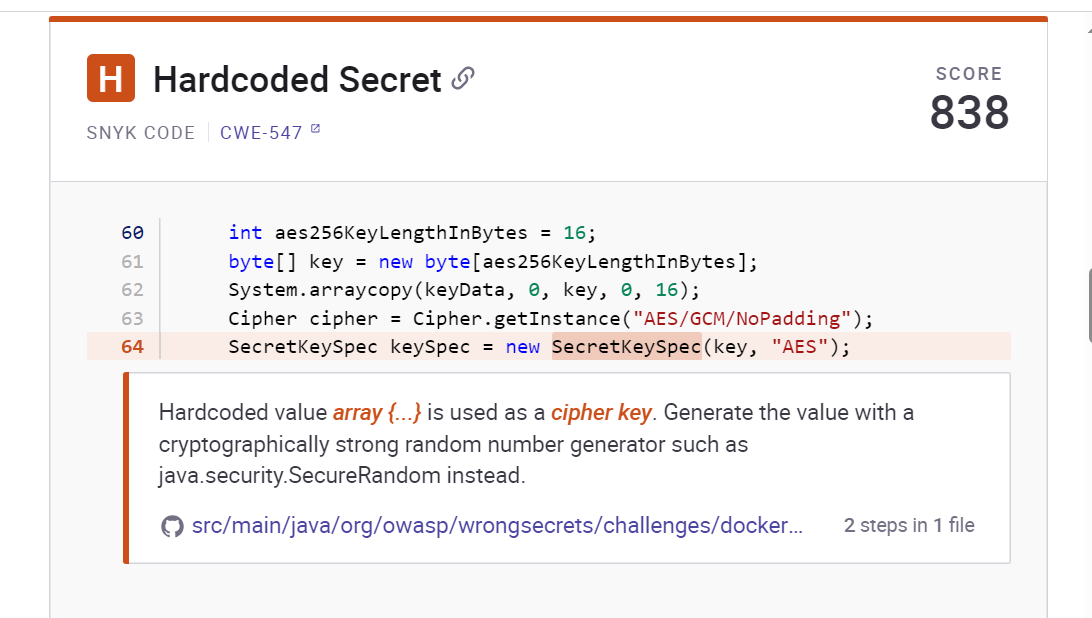


Issue: Improper input validation

Solution: Improper Input validation occurs when untrusted input is not validated for Syntactical and Semantic correctness.

Syntactical validation ensures that the input data is in the correct format (or syntax) that the application expects.

Semantic validation ensures that the input data is correct in its business context.



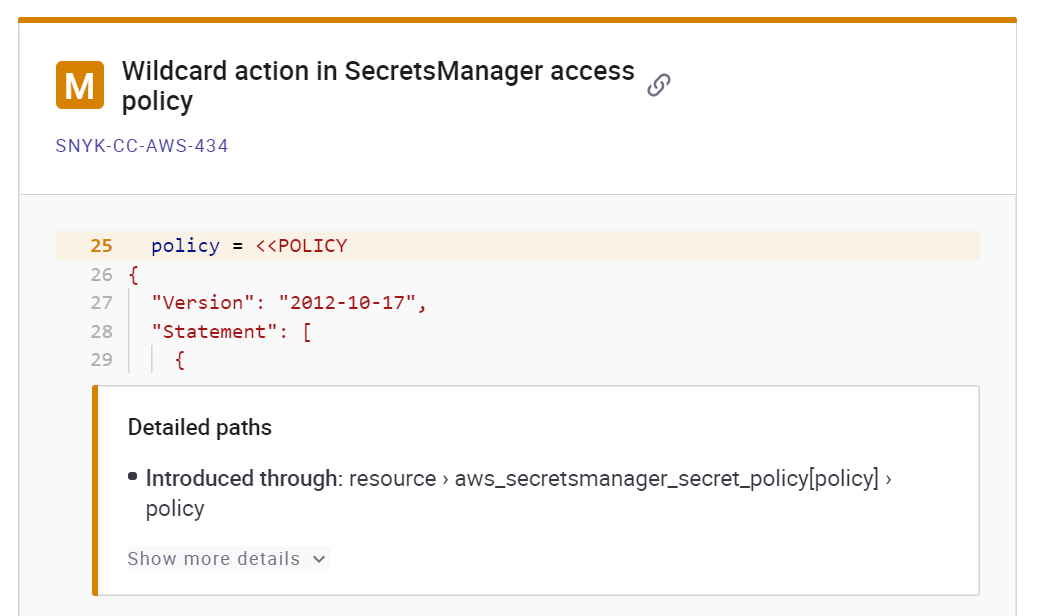
Issue: Hardcoded Secret

Solution:

The existence of hardcoded constants could cause unexpected behavior and the introduction of weaknesses during code maintenance or when making changes to the code if all occurrences are not modified. The use of hardcoded constants is an indication of poor quality.

The usage of symbolic names instead of hard-coded constants is preferred.

Avoid using hard-coded constants. Configuration files offer a more flexible solution.



Issue: Wildcard action in SecretsManager access policy.

Solution:

The impact of this issue is Wildcard permissions grant broad permissions.

The best practice recommends to providing only required permissions explicitly.

And this can be resolved by Remove \* values from Action in policy document. Add specific permissions only for example secretsmanager:DescribeSecret.