Lab Exercise 22- Docker Image Vulnerability Scanning Using Trivy (Windows)

Objective

By the end of this lab, you will be able to:

- Install and configure **Trivy** on Windows
- Scan **Docker images** for vulnerabilities
- Interpret scan reports and take remediation actions

Prerequisites

- Windows 10/11 (with **Docker Desktop** installed and running)
- Internet access (Trivy downloads vulnerability databases)
- Basic familiarity with Docker CLI commands

Step 1: Verify Docker Setup

Before using Trivy, make sure Docker is working correctly.

```
docker --version

docker run hello-world
```

Expected Output:

Docker runs successfully and displays the “Hello from Docker!” message.

Step 2: Install Trivy on Windows

Manual Installation

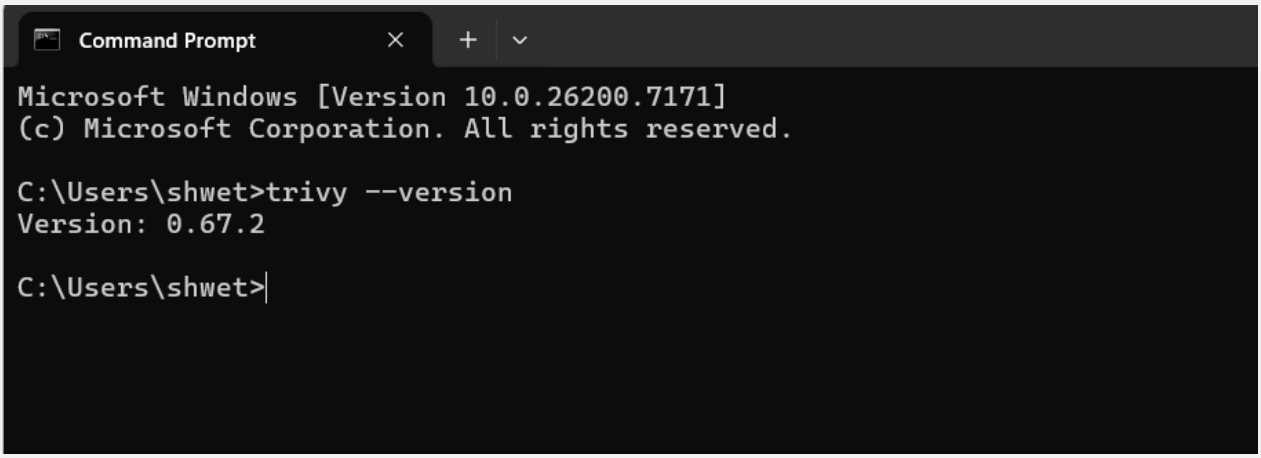
1. Go to the official GitHub releases page:

   https://github.com/aquasecurity/trivy/releases
2. Download the Windows ZIP file (trivy_x.x.x_windows_amd64.zip)
3. Extract it (e.g., to C:\trivy)
4. Add that folder to your **System PATH** environment variable

Verify Installation

Open **PowerShell** and run:

```
trivy –version
```



Expected Output: Trivy version and build information.

Step 3: Pull a Docker Image

Let's pull an image that we'll scan:

```
docker pull nginx:latest
```

```
C:\Users\shwet>docker pull nginx:latest
latest: Pulling from library/nginx
Digest: sha256:553f64aecdc31b5bf944521731cd70e35da4faed96b2b7548a3d8e2598c52a42
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest
```

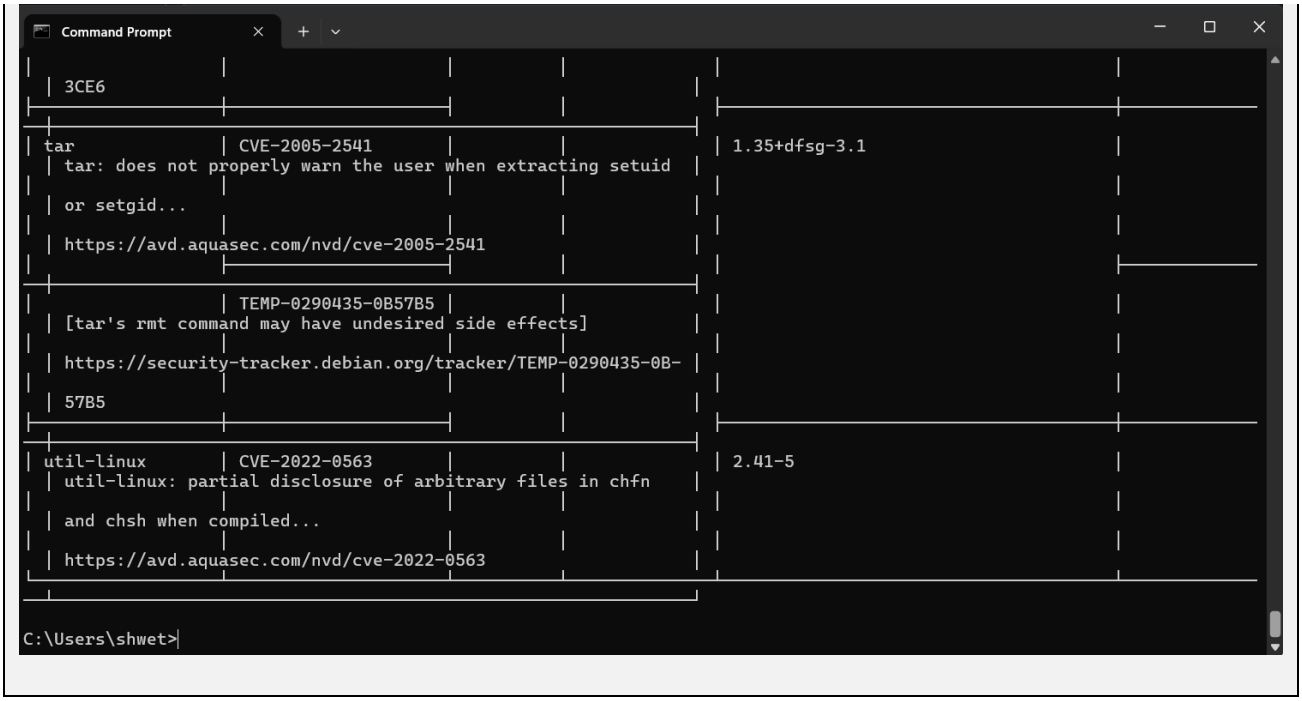
Check it's downloaded:

```
docker images
```

Step 4: Scan Docker Image with Trivy

Now, run a vulnerability scan on the image:

```
trivy image nginx:latest
```



Explanation:

Trivy will:

- Fetch the latest vulnerability database
- Analyze all OS packages and libraries inside the image
- Display severity levels (LOW, MEDIUM, HIGH, CRITICAL)

Sample Output

nginx:latest (debian 12.2)

==========

Total: 12 (LOW: 2, MEDIUM: 4, HIGH: 5, CRITICAL: 1)

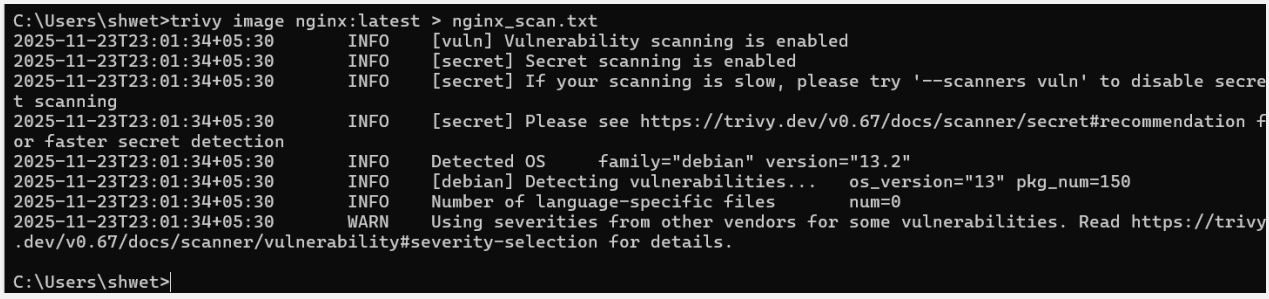
PACKAGE	VULNERABILITY ID	SEVERITY	INSTALLED VERSION	FIXED VERSION
openssl	CVE-2023-0464	HIGH	3.0.9-1	3.0.9-2
zlib	CVE-2022-37434	MEDIUM	1.2.11-5	1.2.12

Step 5: Save Report to a File

You can export the results in different formats.

Save as a text file:

```
trivy image nginx:latest > nginx_scan.txt
```



Save as a JSON report:

```
trivy image --format json -o nginx_scan.json nginx:latest
```

```
C:\Users\shwet>trivy image --format json -o nginx_scan.json nginx:latest
2025-11-23T23:01:59+05:30       INFO    [vuln] Vulnerability scanning is enabled
2025-11-23T23:01:59+05:30       INFO    [secret] Secret scanning is enabled
2025-11-23T23:01:59+05:30       INFO    [secret] If your scanning is slow, please try '--scanners vuln' to disable secre
t scanning
2025-11-23T23:01:59+05:30       INFO    [secret] Please see https://trivy.dev/v0.67/docs/scanner/secret#recommendation f
or faster secret detection
2025-11-23T23:01:59+05:30       INFO    Detected OS     family="debian" version="13.2"
2025-11-23T23:01:59+05:30       INFO    [debian] Detecting vulnerabilities...    os_version="13" pkg_num=150
2025-11-23T23:01:59+05:30       INFO    Number of language-specific files       num=0
2025-11-23T23:01:59+05:30       WARN    Using severities from other vendors for some vulnerabilities. Read https://trivy
.dev/v0.67/docs/scanner/vulnerability#severity-selection for details.
```

Tip: JSON format is useful for automation or CI/CD integration.

Step 6: Scan a Local Image

If you've built your own Docker image:

```
docker build -t myapp:1.0 .
```

```
 | 3CE6
tar              | CVE-2005-2541       | 1.35+dfsg-3.1
 | tar: does not properly warn the user when extracting setuid
 | or setgid...
 | https://avd.aquasec.com/nvd/cve-2005-2541
                 | TEMP-0290435-0B57B5 |
 | [tar's rmt command may have undesired side effects]
 | https://security-tracker.debian.org/tracker/TEMP-0290435-0B-
 | 57B5
util-linux       | CVE-2022-0563       | 2.41-5
 | util-linux: partial disclosure of arbitrary files in chfn
 | and chsh when compiled...
 | https://avd.aquasec.com/nvd/cve-2022-0563
```

```
trivy image myapp:1.0
```

Step 7: Update Vulnerability Database

Keep Trivy's database up-to-date:

```
trivy image --download-db-only
```

Step 8: Clean Up

Remove images (optional):

```
docker rmi nginx:latest
```