Lesson 08 Demo 01

Implementing Kubernetes Container Security

Objective: To implement Kubernetes container security by enhancing network, node, pod, and container security for containerized applications

Tools required: kubeadm, kubectl, kubelet, containerd, and Trivy

Prerequisites: A Kubernetes cluster (refer to Demo 01 from Lesson 01 for setting up a cluster)

Steps to be followed:

1. Implement network security
2. Secure nodes
3. Enhance pod security
4. Scan the application container image using the Trivy CLI

Step 1: Implement network security

1.1 Open the terminal and run the following command to create a YAML file:
nano network-policy.yaml

```
ravitulsianisim@ip-172-31-22-127:~$ nano network-policy.yaml
```

1.2 Add the following configurations into the file:

```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: allow-nginx
  namespace: default
spec:
  podSelector:
    matchLabels:
      app: nginx
  policyTypes:
  - Ingress
```

- Egress
ingress:
- from:
- podSelector:
matchLabels:
role: frontend

```
  GNU nano 6.2                                                    network-policy.yaml *
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: allow-nginx
  namespace: default
spec:
  podSelector:
    matchLabels:
      app: nginx
  policyTypes:
  - Ingress
  - Egress
  ingress:
  - from:
    - podSelector:
        matchLabels:
          role: frontend
```

Note: Save it by pressing ctrl+s and exit by pressing ctrl+x

1.3 After saving the file, apply it to your Kubernetes cluster using the following command:
kubect1 apply -f network-policy.yaml

```
ravitulsianisim@ip-172-31-22-127:~$ kubectl apply -f network-policy.yaml
```

Step 2: Secure nodes

2.1 Open the kubelet configuration file with the following command:
sudo nano /var/lib/kubelet/config.yaml

```
ravitulsianisim@ip-172-31-22-127:~$ sudo nano /var/lib/kubelet/config.yaml
```

2.2 Add the following code to the **config.yaml** file:

```
authentication:
  anonymous:
    enabled: false
  webhook:
    enabled: true
  x509:
    clientCAFile: "/etc/kubernetes/pki/ca.crt"
authorization:
  mode: Webhook
```

```
  GNU nano 6.2                /var/lib/kubelet/config.yaml
authentication:
  anonymous:
    enabled: false
  webhook:
    enabled: true
  x509:
    clientCAFile: "/etc/kubernetes/pki/ca.crt"
authorization:
  mode: Webhook
```

2.3 Restart the kubelet service by running the following command:
sudo systemctl restart kubelet

```
ravitulsianisim@ip-172-31-22-127:~$ sudo systemctl restart kubelet
```

Step 3: Enhance pod security

3.1 Create a YAML file using the following command:
nano psp.yml

```
ravitulsianisim@ip-172-31-22-127:~$ nano psp.yml
```

3.2 Add the following code into the file:

```
apiVersion: policy/v1beta1
kind: PodSecurityPolicy
metadata:
  name: restricted
spec:
  privileged: false
  runAsUser:
    rule: MustRunAsNonRoot
  fsGroup:
    rule: MustRunAs
    ranges:
    - min: 1
      max: 65535
```

```
File  Edit  View  Search  Terminal  Help
  GNU nano 6.2                          psp.yml *
apiVersion: policy/v1beta1
kind: PodSecurityPolicy
metadata:
  name: restricted
spec:
  privileged: false
  runAsUser:
    rule: MustRunAsNonRoot
  fsGroup:
    rule: MustRunAs
    ranges:
    - min: 1
      max: 65535
```

3.3 Save the YAML file and apply it to your Kubernetes cluster using the following command:
kubectl apply -f psp.yaml

```
ravitulsianisim@ip-172-31-22-127:~$ nano psp.yml
ravitulsianisim@ip-172-31-22-127:~$ kubectl apply -f psp.yaml
```

Step 4: Scan the application container image using the Trivy CLI

4.1 Install Trivy using the following commands:

wget https://github.com/aquasecurity/trivy/releases/download/v0.31.0/trivy_0.31.0_Linux-64bit.tar.gz tar zxvf trivy_0.31.0_Linux-64bit.tar.gz

sudo mv trivy /usr/local/bin/

```
ravitulsianisim@ip-172-31-22-127:~$ wget https://github.com/aquasecurity/trivy/releases/download/v0.31.0/trivy_0.31.0_Linux-64bit.tar.gz
tar zxvf trivy_0.31.0_Linux-64bit.tar.gz
--2024-08-08 07:49:09--  https://github.com/aquasecurity/trivy/releases/download/v0.31.0/trivy_0.31.0_Linux-64bit.tar.gz
Resolving github.com (github.com)... 140.82.112.3
Connecting to github.com (github.com)|140.82.112.3|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/180687624/cf2a514c-c7e4-4834-931c-e2cd3aa099a2?X-Amz-Algorithm
=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20240808%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20240808T074910Z&X-Amz-Expires=300&X-
Amz-Signature=3f96e022dd1fbd474c2e1260effe4040ff1895e4fe9cfb48d1aeaa45737363b2&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=180687624&respons
e-content-disposition=attachment%3B%20filename%3Dtrivy_0.31.0_Linux-64bit.tar.gz&response-content-type=application%2Foctet-stream [following]
--2024-08-08 07:49:10--  https://objects.githubusercontent.com/github-production-release-asset-2e65be/180687624/cf2a514c-c7e4-4834-931c-e2cd3aa099a2?
X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20240808%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20240808T074910Z&X-Amz
-Expires=300&X-Amz-Signature=3f96e022dd1fbd474c2e1260effe4040ff1895e4fe9cfb48d1aeaa45737363b2&X-Amz-SignedHeaders=host&actor_id=0&key_id=0&repo_id=18
0687624&response-content-disposition=attachment%3B%20filename%3Dtrivy_0.31.0_Linux-64bit.tar.gz&response-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.108.133, 185.199.110.133, 185.199.109.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.108.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 49725659 (47M) [application/octet-stream]
Saving to: ‘trivy_0.31.0_Linux-64bit.tar.gz’

trivy_0.31.0_Linux- 100%[===================>]  47.42M  83.2MB/s    in 0.6s

2024-08-08 07:49:10 (83.2 MB/s) - ‘trivy_0.31.0_Linux-64bit.tar.gz’ saved [49725659/49725659]

LICENSE
README.md
contrib/asff.tpl
contrib/gitlab-codequality.tpl
contrib/gitlab.tpl
```

```
contrib/gitlab.tpl
contrib/html.tpl
contrib/junit.tpl
trivy
ravitulsianisim@ip-172-31-22-127:~$ sudo mv trivy /usr/local/bin/
ravitulsianisim@ip-172-31-22-127:~$
```

4.2 Scan the Docker container image using the following command:
trivy image <image_name>

```
ravitulsianisim@ip-172-31-22-127:~$ trivy image nginx:latest
2024-08-08T07:53:07.001Z        INFO    Need to update DB
2024-08-08T07:53:07.001Z        INFO    DB Repository: ghcr.io/aquasecurity/trivy-db
2024-08-08T07:53:07.001Z        INFO    Downloading DB...
51.02 MiB / 51.02 MiB [----------------------------------------------------------------------------------------] 100.00% 26.83 MiB
2024-08-08T07:53:09.205Z        INFO    Vulnerability scanning is enabled
2024-08-08T07:53:09.205Z        INFO    Secret scanning is enabled
2024-08-08T07:53:09.205Z        INFO    If your scanning is slow, please try '--security-checks vuln' to disable secret scanning
2024-08-08T07:53:09.205Z        INFO    Please see also https://aquasecurity.github.io/trivy/v0.31.0/docs/secret/scanning/#recommendation for
secret detection
2024-08-08T07:53:13.018Z        INFO    Detected OS: debian
2024-08-08T07:53:13.018Z        INFO    Detecting Debian vulnerabilities...
2024-08-08T07:53:13.036Z        INFO    Number of language-specific files: 0

nginx:latest (debian 12.6)

Total: 151 (UNKNOWN: 0, LOW: 89, MEDIUM: 44, HIGH: 15, CRITICAL: 3)

┌──────────────┬────────────────┬──────────┬────────────────┬───────────────┬──────────────────────────────────────────────────────────────┐
│   Library    │ Vulnerability  │ Severity │ Installed Version │ Fixed Version │                              Title                           │
├──────────────┼────────────────┼──────────┼────────────────┼───────────────┼──────────────────────────────────────────────────────────────┤
│ apt          │ CVE-2011-3374  │ LOW      │ 2.6.1          │               │ It was found that apt-key in apt, all versi
not       │
│              │                │          │                │               │ correctly...
          │
│              │                │          │                │               │ https://avd.aquasec.com/nvd/cve-2011-3374
          │
```

```
nginx:latest (debian 12.6)

Total: 151 (UNKNOWN: 0, LOW: 89, MEDIUM: 44, HIGH: 15, CRITICAL: 3)

┌──────────────┬────────────────┬──────────┬────────────────┬───────────────┬──────────────────────────────────────────────────────────────┐
│   Library    │ Vulnerability  │ Severity │ Installed Version │ Fixed Version │                              Title                           │
├──────────────┼────────────────┼──────────┼────────────────┼───────────────┼──────────────────────────────────────────────────────────────┤
│ apt          │ CVE-2011-3374  │ LOW      │ 2.6.1          │               │ It was found that apt-key in apt, all versions, do
not       │
│              │                │          │                │               │ correctly...
          │
│              │                │          │                │               │ https://avd.aquasec.com/nvd/cve-2011-3374
          │
├──────────────┼────────────────┼──────────┼────────────────┼───────────────┼──────────────────────────────────────────────────────────────┤
│ bash         │ TEMP-0841856-B18BAF │          │ 5.2.15-2       │               │ [Privilege escalation possible to other user than r
oot]      │
│              │                │          │                │               │ https://security-tracker.debian.org/tracker/TEMP-08
41856-B1-  │
│              │                │          │                │               │ 8BAF
          │
├──────────────┼────────────────┼──────────┼────────────────┼───────────────┼──────────────────────────────────────────────────────────────┤
│ bsdutils     │ CVE-2022-0563  │          │ 2.38.1-5+deb12u1 │               │ util-linux: partial disclosure of arbitrary files i
n chfn    │
│              │                │          │                │               │ and chsh when compiled...
          │
│              │                │          │                │               │ https://avd.aquasec.com/nvd/cve-2022-0563
```

This will perform the static vulnerability scan for the existing application container image before deploying it to the Kubernetes cluster.

By following these steps, you have successfully implemented Kubernetes container security by enhancing network, node, pod, and container security for your Kubernetes cluster and containerized applications, reducing vulnerabilities and unauthorized access.