Pentest Tools

Kubernetes Vulnerability Scanner Report

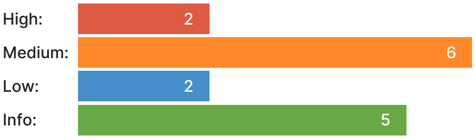
✓ **kubernetes.pentest-ground.com**

Summary

Overall risk level:

High

Risk ratings:



Scan information:

Start time:	Jun 20, 2024 / 15:10:46
Finish time:	Jun 20, 2024 / 15:11:37
Scan duration:	51 sec
Tests performed:	15/15
Scan status:	Finished

Findings

Anonymous Authentication

port 10250/tcp

CONFIRMED

We managed to detect this vulnerability using the following request:
HTTP Request
GET / HTTP/1.1
Host: kubernetes.pentest-ground.com:10250
HTTP Response
HTTP/1.1 200 OK

Details

Vulnerability description:
We found that the Kubelet is configured to allow anonymous (unauthenticated) requests to its HTTP API. This may expose certain information, and capabilities to an attacker with access to the Kubelet API.

Risk description:
The risk exists that an unauthenticated remote attacker could gain access to the Kubelet API.

Recommendation:
We recommend ensuring Kubelet is protected using `--anonymous-auth=false` Kubelet flag. Allow only legitimate users using `--client-ca-file` or `--authentication-token-webhook` Kubelet flags.

References:
https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet-authentication-authorization/

Exposed Run Inside Container

port 10250/tcp

CONFIRMED

We managed to detect this vulnerability using the following request:
HTTP Request
GET /run/test/test/test?cmd= HTTP/1.1
Host: kubernetes.pentest-ground.com:10250
HTTP Response
HTTP/1.1 404

Details

Vulnerability description:
We have detected that the Kubelet is leaking container logs via the `/run` endpoint. This endpoint is exposed as part of the kubelet's debug handlers.

Risk description:

The risk exists that a remote unauthenticated attacker can fully compromise the cluster to steal confidential information, install ransomware or create a reverse shell.

Recommendation:
We recommend disabling `--enable-debugging-handlers` Kubelet flag.

References:
https://github.com/kubernetes/kubernetes/blob/4a6935b31fcc4d1498c977d90387e02b6b93288f/pkg/kubelet/server/server.go
https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet/#options

Exposed Pods

port 10250/tcp

CONFIRMED

We managed to detect this vulnerability using the following request:
HTTP Request
GET /pods HTTP/1.1
Host: kubernetes.pentest-ground.com:10250
HTTP Response
HTTP/1.1 200 OK
Number of pods: 8

Details

Vulnerability description:
We found sensitive information about pods that are bound to a Node using the exposed `/pods` endpoint.

Risk description:
The risk exists that an unauthenticated remote attacker may gain access to sensitive information.

Recommendation:
We recommend ensuring that Kubelet is protected using `--anonymous-auth=false` Kubelet flag. Allow only legitimate users using `--client-ca-file` or `--authentication-token-webhook` Kubelet flags. Disable the read only port by using `--read-only-port=0` Kubelet flag.

References:
https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet/
https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet-authentication-authorization/

Exposed Running Pods

port 10250/tcp

CONFIRMED

We managed to detect this vulnerability using the following request:
HTTP Request
GET /runningpods HTTP/1.1
Host: kubernetes.pentest-ground.com:10250
HTTP Response
HTTP/1.1 200
Number of pods: 8

Details

Vulnerability description:
We have detected that the Kubelet is leaking container logs via the `/runningpdos` endpoint. This endpoint is exposed as part of the kubelet's debug handlers.

Risk description:
The risk exists that an unauthenticated remote attacker can gain insights into the Kubernetes cluster's architecture and active processes, potentially revealing vulnerabilities or leading to targeted attacks.

Recommendation:
We recommend disabling `--enable-debugging-handlers` Kubelet flag.

References:
https://github.com/kubernetes/kubernetes/blob/4a6935b31fcc4d1498c977d90387e02b6b93288f/pkg/kubelet/server/server.go
https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet/#options

Exposed Kubelet Cmdline

CONFIRMED

port 10250/tcp

We managed to detect this vulnerability using the following request:

HTTP Request

```
GET /debug/pprof/cmdline HTTP/1.1
Host: kubernetes.pentest-ground.com:10250
```

HTTP Response

```
HTTP/1.1 200
/var/lib/minikube/binaries/v1.22.3/kubelet--bootstrap-kubeconfig=/etc/kubernetes/bootstrap-kubelet.conf--config=/var/lib/kubelet/config.yaml--container-runtime=docker--hostname-override=steamcloud--kubeconfig=/etc/kubernetes/kubelet.conf--node-ip=10.10.11.133
```

Details

Vulnerability description:

We have detected an exposed Kubelet Cmdline. When the Kubelet is run in debug mode, a Pod running in the cluster is able to access the Kubelet's debug/pprof/cmdline endpoint and examine how the Kubelet was executed on the node, specifically the command line flags that were used, which tells the attacker about what capabilities the Kubelet has which might be exploited.

Risk description:

The risk exists that an unauthenticated remote attacker may gain access to sensitive information.

Recommendation:

We recommend disabling `--enable-debugging-handlers` Kubelet flag.

References:

https://github.com/kubernetes/kubernetes/blob/4a6935b31fcc4d1498c977d90387e02b6b93288f/pkg/kubelet/server/server.go#L327
https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet/#options

Exposed Container Logs

CONFIRMED

port 10250/tcp

We managed to detect this vulnerability using the following request:

HTTP Request

```
GET /containerLogs/default/nginx/nginx HTTP/1.1
Host: kubernetes.pentest-ground.com:10250
```

HTTP Response

```
HTTP/1.1 200
```

Details

Vulnerability description:

We have detected that the Kubelet is leaking container logs via the `/containerLogs` endpoint. This endpoint is exposed as part of the kubelet's debug handlers.

Risk description:

The risk exists that an unauthenticated remote attacker may gain access to sensitive information contained within the logs. Such information might include application details, system configurations, or credentials, which could be exploited to gain further unauthorized access or to perform malicious activities within the system.

Recommendation:

We recommend disabling `--enable-debugging-handlers` Kubelet flag.

References:

https://github.com/kubernetes/kubernetes/blob/4a6935b31fcc4d1498c977d90387e02b6b93288f/pkg/kubelet/server/server.go
https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet/#options

Exposed System Logs

CONFIRMED

port 10250/tcp

We managed to detect this vulnerability using the following request:

HTTP Request

```
GET /logs/ HTTP/1.1
Host: kubernetes.pentest-ground.com:10250
```

HTTP Response

```
HTTP/1.1 200
```

Details

Vulnerability description:
We have detected that the Kubelet is leaking system logs via the /logs endpoint. This endpoint is exposed as part of the kubelet's debug handlers.

Risk description:
The risk exists that an unauthenticated remote attacker may gain access to sensitive information contained within the logs. Such information might include application details, system configurations, or credentials, which could be exploited to gain further unauthorized access or to perform malicious activities within the system.

Recommendation:
We recommend disabling `--enable-debugging-handlers` Kubelet flag.

References:
https://github.com/kubernetes/kubernetes/blob/4a6935b31fcc4d1498c977d90387e02b6b93288f/pkg/kubelet/server/server.go
https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet/#options

Exposed Existing Privileged Container(s) Via Secure Kubelet Port

CONFIRMED

port 10250/tcp

```
{
"pod_namespace": "kube-system",
"pod_id": "kube-proxy-7pwmt",
"container_name": "kube-proxy",
"service_account_token": "eyJhbGciOiJSUzI1NiIsImtpZCI6IjdwQ3NjcWFCTGVvMFdrVUowWEstbVRuSEZVR2c2RU5LUzVKTE5ka0VoT3cif
Q.eyJhdWQiOlsiaHR0cHM6Ly9rdWJlcm5ldGVzLmRlZmF1bHQuc3ZjLmNsdXN0ZXIubG9jYWwiXSwiZXhwIjoxNzUwNDE5Njk2LCJpYXQiOj
E3MTg4ODM2OTYsImlzcyI6Imh0dHBzOi8va3ViZXJuZXRlcy5kZWZhdWx0LnN2Yy5jbHVzdGVyLmxvY2FsIiwia3ViZXJuZXRlcy5pbyI6eyJu
YW1lc3BhY2UiOiJrdWJlLXN5c3RlbSIsInBvZCI6eyJuYW1lIjoia3ViZS1wcm94eS03cHdtdCIsInVpZCI6Ijg5NzRkNGU5LWJiZTUtNDJkYi05OD
AzLWVhNDgzMThjNmJlNyJ9LCJzZXJ2aWNlYWNjb3VudCI6eyJuYW1lIjoia3ViZS1wcm94eSIsInVpZCI6ImU0ODA2YzIwLWYwZjQtNDdjYS
04Y2VjLWQ1YjBiM2U2YzUwNSJ9LCJ3YXJuYWZ0ZXIiOjE3MTg4ODczMDN9LCJuYmYiOjE3MTg4ODM2OTYsInN1YiI6InN5c3RlbTpzZXJ2
aWNlYWNjb3VudDprdWJlLXN5c3RlbTprdWJlLXByb3h5In0.i6iOsYo3nmUQGFnbeIMGYP0UrHJepyWpHnuOVmtYc0mIjZScRFb5T6ea23g
mA0ifIwiR_7via33NhyI3-z1uUIrtnWWFyDM7sLU0AcsNb2oxijqSYSgUhejZGgMMxw3qebZPwGU1NaUj5YF_6EMcrAVRBqSVfGJYrBfbAE05
9Uw9EEZYpbKPPsM0OFkA3PNvyq3OI-Vk4wGDoMIE89VHOOUXYk8vUgFhuFA5ig4EHf_W3VVFj5wGQZP9n9ODqeBIU72PBIeSDCwjm4H
376Iv5Iu6fAOPB2qDELp07aYYbUVOa9gYyZCHN84uwHucOmy2Qon_13MbI1I5aO1FOPygcQ",
"environment_variables": "PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin\nHOSTNAME=steamcloud\nNODE_NAME=stea
mcloud\nKUBE_DNS_PORT_9153_TCP_ADDR=10.96.0.10\nKUBERNETES_PORT_443_TCP_PORT=443\nKUBE_DNS_SERVICE_PORT=53\n
KUBE_DNS_SERVICE_PORT_DNS=53\nKUBE_DNS_PORT_53_UDP_PORT=53\nKUBE_DNS_PORT_53_UDP_ADDR=10.96.0.10\nKUBE_DNS_
PORT_53_TCP_PORT=53\nKUBE_DNS_PORT_9153_TCP_PORT=9153\nKUBERNETES_SERVICE_PORT=443\nKUBERNETES_SERVICE_POR
T_HTTPS=443\nKUBERNETES_PORT_443_TCP_PROTO=tcp\nKUBE_DNS_SERVICE_PORT_DNS_TCP=53\nKUBE_DNS_PORT_53_TCP_PR
OTO=tcp\nKUBE_DNS_PORT_9153_TCP=tcp://10.96.0.10:9153\nKUBE_DNS_PORT_9153_TCP_PROTO=tcp\nKUBERNETES_PORT_443_TC
P=tcp://10.96.0.1:443\nKUBERNETES_PORT_443_TCP_ADDR=10.96.0.1\nKUBE_DNS_SERVICE_HOST=10.96.0.10\nKUBE_DNS_PORT=ud
p://10.96.0.10:53\nKUBE_DNS_PORT_53_UDP=udp://10.96.0.10:53\nKUBE_DNS_PORT_53_TCP=tcp://10.96.0.10:53\nKUBERNETES_SER
VICE_HOST=10.96.0.1\nKUBERNETES_PORT=tcp://10.96.0.1:443\nKUBE_DNS_SERVICE_PORT_METRICS=9153\nKUBE_DNS_PORT_53_U
DP_PROTO=udp\nKUBE_DNS_PORT_53_TCP_ADDR=10.96.0.10\nHOME=/root"
}
```

Details

Vulnerability description:
We found that the Kubelet is configured to allow anonymous (unauthenticated) requests to its HTTPS API. This may expose certain information and capabilities to an attacker with access to the Kubelet API. A privileged container is given access to all devices on the host and can work at the kernel level. It is declared using the `Pod.spec.containers[].securityContext.privileged` attribute. This may be useful for infrastructure containers that perform setup work on the host but is a dangerous attack vector. Furthermore, if the Kubelet and the API server authentication mechanisms are (mis)configured such that anonymous requests can execute commands via the API within the containers (specifically privileged ones), a malicious actor can leverage such capabilities to do way more damage in the cluster than expected: e.g., start/modify process on the host.

Risk description:
The risk exists that an unauthenticated remote attacker can gain access to a privileged container and leverage its permissions to do more damage in the cluster.

Recommendation:
We recommend ensuring Kubelet is protected using `--anonymous-auth=false` Kubelet flag. Allow only legitimate users using `--client-ca-file` or `--authentication-token-webhook` Kubelet flags. This is usually done by the installer or cloud provider. Minimize the use of privileged containers. Use Pod Security Policies to enforce using `privileged: false` policy. Review the RBAC permissions to the Kubernetes API server for the anonymous and default service account, including bindings. Remove `AlwaysAllow` from `--authorization-mode` in the Kubernetes API server config. Alternatively, set `--anonymous-auth=false` in the Kubernetes API server config; this will depend on the API server version running.

References:
https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet-authentication-authorization/
https://kubernetes.io/docs/concepts/workloads/pods/pod/#privileged-mode-for-pod-containers
https://kubernetes.io/docs/concepts/policy/pod-security-policy/#privileged
https://kubernetes.io/docs/reference/access-authn-authz/rbac/

Kubernetes Version Disclosure

CONFIRMED

port 8443/tcp

We managed to detect the version using the following request:

HTTP Request

```
GET /version HTTP/1.1
Host: kubernetes.pentest-ground.com:8443
```

HTTP Response

```
HTTP/1.1 200 OK
...
v1.22.3
...
```

Details

Vulnerability description:
We have detected the specific version of the Kubernetes cluster.

Risk description:
The risk exists that an unauthenticated remote attacker that can execute commands on a pod in the cluster may be able to query the metadata service and discover additional information about the environment.

Recommendation:
We recommend disabling `--enable-debugging-handlers` Kubelet flag.

References:
https://github.com/kubernetes/kubernetes/blob/4a6935b31fcc4d1498c977d90387e02b6b93288f/pkg/kubelet/server/server.go
https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet/#options

Cluster Health Disclosure

CONFIRMED

port 10250/tcp

We managed to detect this vulnerability using the following request:

HTTP Request

```
GET /healthz HTTP/1.1
Host: kubernetes.pentest-ground.com:10250
```

HTTP Response

```
HTTP/1.1 200
ok
```

Details

Vulnerability description:
We found that the Kubelet is leaking its health information, which may contain sensitive information, via the `/healthz` endpoint. This endpoint is exposed as part of the kubelet's debug handlers.

Risk description:
The risk exists that an unauthenticated remote attacker can gain access to sensitive information about the cluster.

Recommendation:
We recommend disabling `--enable-debugging-handlers` Kubelet flag.

References:
https://github.com/kubernetes/kubernetes/blob/4a6935b31fcc4d1498c977d90387e02b6b93288f/pkg/kubelet/server/server.go
https://kubernetes.io/docs/reference/command-line-tools-reference/kubelet/#options

Open ports discovery

CONFIRMED

Port	State	Service	Product	Product Version
2379	open	etcd-client		
8443	open	https	Golang net/http server	
10250	open	https	Golang net/http server	

Details

Vulnerability description:
This is the list of ports that have been found on the target host.

Risk description:
This is the list of ports that have been found on the target host. Having unnecessary open ports may expose the target to more risks because those network services and applications may contain vulnerabilities.

Recommendation:
We recommend reviewing the list of open ports and closing the ones which are not necessary for business purposes.

Etcd service found.

port 2379/tcp

Node/Master cluster component found.

Kubelet API service found.

port 10250/tcp

API Server service found.

port 8443/tcp

Scan coverage information

List of tests performed (15/15)

- ✓ Scanning target ports...
- ✓ Checking for Etcd...
- ✓ Checking for Node/Master cluster component...
- ✓ Checking for Kubelet API...
- ✓ Checking for Anonymous Authentication on port 10250...
- ✓ Checking for API Server...
- ✓ Checking for Kubernetes Version Disclosure on port 8443...
- ✓ Checking for Exposed Pods on port 10250...
- ✓ Checking for Cluster Health Disclosure on port 10250...
- ✓ Checking for Exposed Running Pods on port 10250...
- ✓ Checking for Exposed Kubelet Cmdline on port 10250...
- ✓ Checking for Exposed Container Logs on port 10250...
- ✓ Checking for Exposed Run Inside Container on port 10250...
- ✓ Checking for Exposed System Logs on port 10250...
- ✓ Checking for Exposed Existing Privileged Container(s) Via Secure Kubelet Port on port 10250...

Scan parameters

Target:	kubernetes.pentest-ground.com
Preset:	Custom
Ports to scan:	10250,2379,8443
Authentication:	False
Check alive:	False
Active detections:	True