Prometheus Metrics Collection on Amazon EKS and Kubernetes clusters

Setting Up IAM Roles

The first step is to set up the necessary IAM role in the cluster. There are two methods:

- Set up an IAM role for a service account, also known as a *service role*. This method works for both the EC2 launch type and the Fargate launch type.
- Add an IAM policy to the IAM role used for the cluster. This works only for the EC2 launch type.

Create your OIDC identity provider for your cluster

```
eksctl utils associate-iam-oidc-provider --cluster eksworkshop-eksctl
--app
```

Set up a service role (EC2 launch type and Fargate launch type)

To set up a service role, enter the following command. Replace MyCluster with the name of the cluster.

```
eksctl create iamserviceaccount \
 --name cwagent-prometheus \
--namespace amazon-cloudwatch \
 --cluster MyCluster \
--attach-policy-arn arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
\
--approve \
--override-existing-serviceaccounts
```

Add the necessary policy to the IAM role for your worker nodes

In order for CloudWatch to get the necessary monitoring info, we need to install the CloudWatch Agent to our EKS Cluster.

First, we will need to ensure the Role Name our workers use is set in our environment:

```
test -n "$ROLE_NAME" && echo ROLE_NAME is "$ROLE_NAME" || echo ROLE_NAME
is not set
```

We will attach the policy to the nodes IAM Role:

```
aws iam attach-role-policy \
  --role-name $ROLE_NAME \
  --policy-arn arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
```

Finally, let's verify that the policy has been attached to the IAM ROLE:

```
aws iam list-attached-role-policies --role-name $ROLE_NAME | grep
CloudWatchAgentServerPolicy || echo 'Policy not found'
```

Output

```
"PolicyName": "CloudWatchAgentServerPolicy",
"PolicyArn": "arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy"
```

Installing the CloudWatch Agent to Collect Prometheus Metrics

create it by entering the following command:

```
kubectl create namespace amazon-cloudwatch
```

To deploy the agent with the default configuration and have it send data to the AWS Region that it is installed in, enter the following command

```
kubectl apply -f
https://raw.githubusercontent.com/aws-samples/amazon-cloudwatch-containe
r-insights/latest/k8s-deployment-manifest-templates/deployment-mode/serv
ice/cwagent-prometheus/prometheus-eks.yaml
```

Installing the CloudWatch Agent on a Kubernetes Cluster

To install the CloudWatch agent with Prometheus support on a cluster running Kubernetes, enter the following command:

```
curl
https://raw.githubusercontent.com/aws-samples/amazon-cloudwatch-containe
r-insights/latest/k8s-deployment-manifest-templates/deployment-mode/serv
ice/cwagent-prometheus/prometheus-k8s.yaml |
sed "s/{{cluster_name}}/MyCluster/;s/{{region_name}}/region/" |
kubectl apply -f -
```

Replace MyCluster with the name of the cluster. This name is used in the log group name that stores the log events collected by the agent, and is also used as a dimension for the metrics collected by the agent.

Replace region with the name of the AWS Region where you want the metrics to be sent. For example, us-west-1.

Verify that the Agent is Running

On both Amazon EKS and Kubernetes clusters, you can enter the following command to confirm that the agent is running.

```
kubectl get pod -l "app=cwagent-prometheus" -n amazon-cloudwatch
```

Viewing Your Prometheus Metrics

1. **To see all your Prometheus metrics**
    - Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/
    - In the navigation pane, choose **Metrics**.
    - In the list of namespaces, choose **ContainerInsights/Prometheus** or **ECS/ContainerInsights/Prometheus**.
    - Choose one of the sets of dimensions in the following list. Then select the checkbox next to the metrics that you want to see.

2. **To see pre-built reports on your Prometheus metrics**
    - Open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/
    - In the navigation pane, choose **Performance Monitoring**.
    - In the drop-down box near the top of the page, choose any of the Prometheus options.
      In the other drop-down box, choose a cluster to view
3. We have also provided custom dashboards for NGINX, App Mesh, Memcached, HAProxy, and Java/JMX.
4. **To use a custom dashboard that Amazon has provided**
    - Open the CloudWatch console at ttps://console.aws.amazon.com/cloudwatch/
5. In the navigation pane, choose **Dashboards**.
6. Choose **Create Dashboard**. Enter a name for the new dashboard, and choose **Create dashboard**.
7. In **Add to this dashboard**, choose **Cancel**.
8. Choose **Actions**, **View/edit source**.
9. Download one of the following JSON files:
    - NGINX custom dashboard source on Github
10. App Mesh custom dashboard source on Github
11. Memcached custom dashboard source on Github
12. HAProxy-Ingress custom dashboard source on Github
    Java/JMX custom dashboard source on Github
13. Open the JSON file that you downloaded with a text editor, and make the following changes:
    - Replace all the {{YOUR_CLUSTER_NAME}} strings with the exact name of your cluster. Make sure not to add whitespaces before or after the text.
    - Replace all the {{YOUR_AWS_REGION}} strings with the AWS Region where your cluster is running. For example, us-west-1 Make sure not to add whitespaces before or after the text.
    - Replace all the {{YOUR_NAMESPACE}} strings with the exact namespace of your workload.
    - Replace all the {{YOUR_SERVICE_NAME}} strings with the exact service

name of your workload. For example, haproxy-haproxy-ingress-controller-metrics

14. Copy the entire JSON blob and paste it into the text box in the CloudWatch console, replacing what is already in the box.
15. Choose **Update**, **Save dashboard**.