Red Hat OpenShift Service on AWS 4

Backing up and restoring applications

Backing up and restoring of applications data
Backing up and restoring of applications data
Abstract

This document provides information about backing up applications.
## Table of Contents

### CHAPTER 1. BACKING UP APPLICATIONS

1.1. PREPARING AWS CREDENTIALS ......................................................... 3  
1.2. INSTALLING THE OADP OPERATOR AND PROVIDING THE IAM ROLE  6  
1.3. KNOWN ISSUES ........................................................................... 9  
1.4. ADDITIONAL RESOURCES .............................................................. 9
CHAPTER 1. BACKING UP APPLICATIONS

You can employ OpenShift API for Data Protection (OADP) with Red Hat OpenShift Service on AWS (ROSA) clusters to backup and restore application data. Before installing OADP, you must set up role and policy credentials for OADP so that it can use the AWS API.

This is a two stage process:

1. Prepare AWS credentials.
2. Install the OADP Operator and provide it with the IAM role.

1.1. PREPARING AWS CREDENTIALS

An AWS account must be ready to accept an OADP installation.

Procedure

1. Create the following environment variables by running the following commands:

   ```
   $ export CLUSTER_NAME=my-cluster
   export ROSA_CLUSTER_ID=$(rosa describe cluster -c ${CLUSTER_NAME} --output json | jq -r .id)
   export REGION=$(rosa describe cluster -c ${CLUSTER_NAME} --output json | jq -r .region.id)
   export OIDC_ENDPOINT=$(oc get authentication.config.openshift.io cluster -o jsonpath='{.spec.serviceAccountIssuer}' | sed 's|^https://||')
   export AWS_ACCOUNT_ID=$(aws sts get-caller-identity --query Account --output text)
   export CLUSTER_VERSION=$(rosa describe cluster -c ${CLUSTER_NAME} -o json | jq -r .version.raw_id | cut -f -2 -d '.')
   export ROLE_NAME="${CLUSTER_NAME}-openshift-oadp-aws-cloud-credentials"
   export SCRATCH="/tmp/${CLUSTER_NAME}/oadp"
   mkdir -p ${SCRATCH}
   echo "Cluster ID: ${ROSA_CLUSTER_ID}, Region: ${REGION}, OIDC Endpoint: ${OIDC_ENDPOINT}, AWS Account ID: ${AWS_ACCOUNT_ID}"
   ```

   Replace **my-cluster** with your ROSA cluster name.

2. On the AWS account, create an IAM policy to allow access to S3.

   a. Check to see if the policy exists by running the following command:

   ```
   $ POLICY_ARN=$(aws iam list-policies --query "Policies[? PolicyName=='RosaOadpVer1'].ARN" --output text)
   ```

   Replace **RosaOadp** with your policy name.
Use the following command to create the policy JSON file and then create the policy in ROSA.

```bash
$ if [[ -z "${POLICY_ARN}" ]]; then
cat << EOF > ${SCRATCH}/policy.json
{
  "Version": "2012-10-17",
  "Statement": [
    { "Effect": "Allow",
    ],
    "Resource": "*"
  }
}
EOF

POLICY_ARN=$(aws iam create-policy --policy-name "RosaOadpVer1" --policy-document file:///${SCRATCH}/policy.json --query Policy.Arn --tags Key=rosa_openshift_version,Value=${CLUSTER_VERSION} Key=rosa_role_prefix,Value=ManagedOpenShift Key=operator_namespace,Value=openshift-oadp Key=operator_name,Value=openshift-
scratch is a name for a temporary directory created for the environment variables.

c. View the policy ARN by running the following command:

```bash
$ echo ${POLICY_ARN}
```

3. Create an IAM role trust policy for the cluster:

a. Create the trust policy file by running the following command:

```bash
$ cat <<EOF > ${SCRATCH}/trust-policy.json
{
  "Version": "2012-10-17",
  "Statement": [
    {"Effect": "Allow",
      "Principal": {"Federated": "arn:aws:iam::${AWS_ACCOUNT_ID}:oidc-provider/${OIDC_ENDPOINT}"},
      "Action": "sts:AssumeRoleWithWebIdentity",
      "Condition": {
        "StringEquals": {
        "${OIDC_ENDPOINT}:sub": [
        "system:serviceaccount:openshift-adp:velero"
        ]
        }
      }
    }
  ]
}
EOF
```

b. Create the role by running the following command:

```bash
$ ROLE_ARN=$(aws iam create-role --role-name
  "$ROLE_NAME" \
  --assume-role-policy-document file://${SCRATCH}/trust-policy.json \
  --tags Key=rosa_cluster_id,Value=${ROSA_CLUSTER_ID} \
  Key=rosa_openshift_version,Value=${CLUSTER_VERSION} \
  Key=rosa_role_prefix,Value=ManagedOpenShift \
  Key=operator_namespace,Value=openshift-adp \
  Key=operator_name,Value=openshift-oadp \
  --query Role.Arn --output text)
```

c. View the role ARN by running the following command:

```bash
$ echo ${ROLE_ARN}
```

4. Attach the IAM policy to the IAM role by running the following command:
1.2. INSTALLING THE OADP OPERATOR AND PROVIDING THE IAM ROLE

AWS Security Token Service (AWS STS) is a global web service that provides short-term credentials for IAM or federated users. Red Hat OpenShift Service on AWS (ROSA) with STS is the recommended credential mode for ROSA clusters. This document describes how to install OpenShift API for Data Protection (OADP) on (ROSA) with AWS STS.

IMPORTANT

Restic and Kopia are not supported in the OADP on ROSA with AWS STS environment. Make sure that the Restic/Kopia node agent is disabled. For backing up volumes, OADP on ROSA with AWS STS supports only native snapshots and CSI snapshots. See Known Issues for more information.

IMPORTANT

In an Amazon ROSA cluster using STS authentication, restoring backed-up data in a different AWS region is not supported.

The Data Mover feature is not currently supported in ROSA clusters. You can use native AWS S3 tools for moving data.

Prerequisites

- A cluster with the required access and tokens. For instructions, see the procedure in "Preparing AWS credentials". If you plan to use two different clusters for backing up and restoring, you need to prepare AWS credentials, including ROLE_ARN, for each cluster.

Procedure

1. Create an OpenShift secret from your AWS token file by entering the following commands.

   a. Create the credentials file:

   ```bash
   $ cat <<EOF > ${SCRATCH}/credentials
   [default]
   role_arn = ${ROLE_ARN}
   web_identity_token_file = /var/run/secrets/openshift/serviceaccount/token
   EOF
   
   $ oc create namespace openshift-adp
   $ oc create secret generic aws-token --from-file=aws-token=${SCRATCH}/credentials
   ```

   b. Create a namespace for OADP:

   ```bash
   $ oc create namespace openshift-adp
   ```

   c. Create the OpenShift secret:

   ```bash
   $ oc create secret generic aws-secret --from-literal=aws-secret=cluster
   ```

   d. Attach the role policy:

   ```bash
   $ aws iam attach-role-policy --role-name "${ROLE_NAME}" --policy-arn ${POLICY_ARN}
   ```

Next steps

- Continue to Installing the OADP Operator and providing the IAM role.
2. Install the OADP Operator.

   a. In the Red Hat OpenShift Service on AWS web console, navigate to Operators → OperatorHub.

   b. Search for the OADP Operator, then click Install.

3. Create AWS cloud storage using your AWS credentials:

   ```bash
   $ oc -n openshift-adp create secret generic cloud-credentials
   --from-file=${SCRATCH}/credentials
   $ cat << EOF | oc create -f -
   apiVersion: oadp.openshift.io/v1alpha1
   kind: CloudStorage
   metadata:
     name: ${CLUSTER_NAME}-oadp
     namespace: openshift-adp
   spec:
     creationSecret:
       key: credentials
       name: cloud-credentials
     enableSharedConfig: true
     name: ${CLUSTER_NAME}-oadp
     provider: aws
     region: $REGION
   EOF
   $ cat << EOF | oc create -f -
   apiVersion: oadp.openshift.io/v1alpha1
   kind: DataProtectionApplication
   metadata:
     name: ${CLUSTER_NAME}-dpa
     namespace: openshift-adp
   spec:
     backupLocations:
     - bucket:
         cloudStorageRef:
           name: ${CLUSTER_NAME}-oadp
           credential:
             key: credentials
   EOF
   ```

4. Create the **DataProtectionApplication** resource, which is used to configure the connection to the storage where the backups and volume snapshots are stored:

   ```bash
   $ cat << EOF | oc create -f -
   apiVersion: oadp.openshift.io/v1alpha1
   kind: DataProtectionApplication
   metadata:
     name: ${CLUSTER_NAME}-dpa
     namespace: openshift-adp
   spec:
     backupLocations:
     - bucket:
         cloudStorageRef:
           name: ${CLUSTER_NAME}-oadp
           credential:
             key: credentials
   EOF
   ```
name: cloud-credentials
prefix: velero
default: true
cfg:
  region: ${REGION}
configuration:
  velero:
    defaultPlugins:
    - openshift
    - aws
nodeAgent:
  enable: false
  uploaderType: restic
snapshotLocations:
  - velero:
    config:
      credentialsFile: /tmp/credentials/openshift-adp/cloud-credentials-credentials
    enableSharedConfig: "true"
    profile: default
    region: ${REGION}
    provider: aws
EOF

1. See the first note below.

2. The credentialsFile field is the mounted location of the bucket credential on the pod.

3. The enableSharedConfig field allows the snapshotLocations to share or reuse the credential defined for the bucket.

4. Use the profile name set in the AWS credentials file.

5. Specify region as your AWS region. This must be the same as the cluster region.

You are now ready to backup and restore OpenShift applications, as described in the OADP documentation.

NOTE

The enable parameter of restic is set to false in this configuration because OADP does not support Restic in ROSA environments.

If you are using OADP 1.2, replace this configuration:

nodeAgent:
  enable: false
  uploaderType: restic

with the following:

restic:
  enable: false
NOTE

If you want to use two different clusters for backing up and restoring, the two clusters must have identical AWS S3 storage names in both the cloudstorage CR and the OADP DataProtectionApplication configuration.

Additional resources

- Preparing AWS credentials

1.3. KNOWN ISSUES

Restic, Kopia, and DataMover are not supported or recommended

- CloudStorage: openshift-adp-controller-manager crashloop seg fault with Restic enabled
- (Affects OADP 1.1.x_ only): CloudStorage: bucket is removed on CS CR delete, although it doesn’t have "oadp.openshift.io/cloudstorage-delete": "true"

1.4. ADDITIONAL RESOURCES

- Understanding ROSA with STS
- Getting started with ROSA STS
- Creating a ROSA cluster with STS
- About installing OADP
- Configuring CSI volumes
- ROSA storage options