ARCHITECTURE GUIDE

Amazon Security Token Service (STS) AssumeRole Activation Guide

VERSION 11 FEATURE RELEASE 18,19, 20

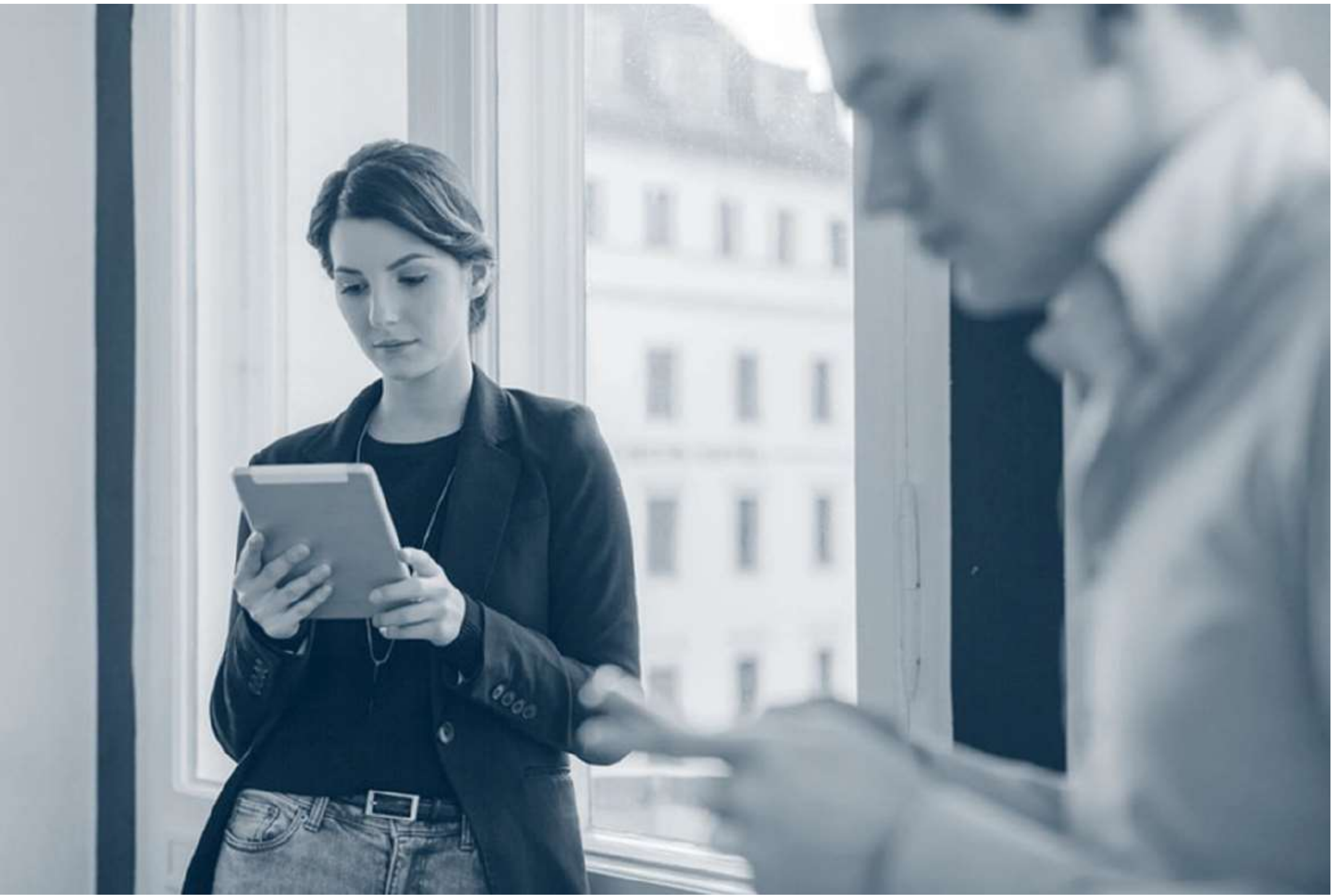
15th June 2020

As businesses adopt Cloud Services, the **security** of operational processes and business data is paramount. Amazon Identify and Access Management (IAM) provides next-generation security controls that allow a business to segment and control access to critical business information.

One of the key technologies is providing granular role-based access controls within Amazon is the use of **IAM Roles** and temporary credential sharing, using Amazon Security Token Service (STS) **AssumeRole** capability. This capability allows your centralized IT team to perform data management activities on behalf of your individual business units or departments.

Customers are responsible for managing their data (including encryption options), classifying their assets, and using IAM tools to apply the appropriate permissions.

Amazon Shared Responsibility Model aws.amazon.com/compliance/shared-responsibility-model



NOTICES

This document is provided for informational purposes only. It represents Commvault's current product offerings and practices as of the date of issue of this document, of which are subject to change without notice. The responsibilities and liabilities of Commvault® to its customers are controlled by Commvault agreements, and this document is not part of, nor does it modify, any agreement between Commvault and its customers.

TABLE OF CONTENTS

Introduction

This document serves as a brief introduction and activation guide to using Amazon Security Token Service (STS) AssumeRole functionality with your Commvault Complete™ Backup and Recovery platform. Commvault allows customers to utilize STS:AssumeRole to perform cross-account data management activities for the following use-cases:

- Authentication and writing to Amazon S3 Cloud Library (**STS Assume Role >**, **STS Assume Role with IAM Role Policy >**)
- Performing Amazon native snapshot backup & recovery of EC2 instances and associated Elastic Block Store (EBS) volumes
- Performing Amazon native snapshot backup & recovery of RDS

This document will focus on the **Technology Preview** capability that provides AssumeRole functionality for EC2 and RDS native-snapshot based backups.

Note: This document has links to the latest version of Commvault software (**v11 Feature Release 20 >**). The capabilities covered in this document are supported for 11.18.26, 11.19.13, and 11.20.0.

Prerequisites

Commvault software

You must be running one of the following Commvault major revisions:

- Commvault v11, Feature Release 18 (**hotfix pack 26 >** or later)
- Commvault v11, Feature Release 19 (**hotfix pack 13 >** or later)
- **Commvault v11, Feature Release 20 >**

Commvault components

You will require the following components deployed to utilize Amazon STS:AssumeRole with Commvault:

- A Commvault **CommServe® >** to perform command and control of data protection actions
- At least one Commvault **MediaAgent >** to perform data storage and replication (may reside on CommServe)
- At least one Commvault Cloud **Access Node >** (also referred to as a **VSA** Proxy) to perform communication with Amazon API endpoints.
- Access Node(s) must reside within Amazon EC2 service to utilize a **IAM Role** for authentication (See **Adding an Amazon Hypervisor >**).

Note: Commvault will require access from the Access Node to the regional Security Token Service (STS) endpoint – for example: `https://sts.us-east-1.amazonaws.com` (see **AWS Security Token Service endpoints and quotas >**).

Amazon account configuration

There are prerequisites to utilizing STS:AssumeRole functionality with your Amazon account(s) and Commvault software. There are a number of common terms used (below):

- An **admin** account refers to a shared 'IT' account responsible for coordinating and protecting Amazon resources.
- A **tenant** account is typically an individual business unit that has resources requiring protection (i.e. EC2 instances, RDS instances).

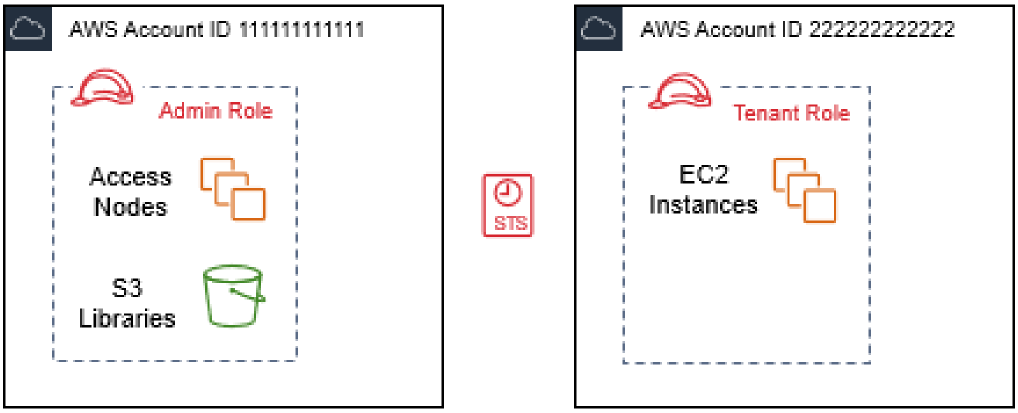
As a minimum you will require two Amazon account(s) to utilize STS:AssumeRole functionality.

IAM Roles, Permissions and Trust

These details are covered later in this document, at a minimum you will require:

- One account with **sts:AssumeRole >** permissions and Commvault **amazon_restricted_role_permissions.json >** policy (**Admin** account).
- One account with EC2/RDS instances to protect and **sts:AssumeRole >** permissions and Commvault **amazon_restricted_role_permissions.json >** permissions applied (**Tenant** account).
- **Tenant account** configured to **trust** the **Admin account**.

Resources are segregated between Admin and Tenant accounts



In this configuration:

- At least two (2) **Amazon Accounts** exist across your organization.
- Data protection resources are owned by an **Admin Account** (Commvault Access Nodes, Cloud Libraries).
- Per-department (Tenant) resources are owned by their own **Tenant Account**.
- The **Tenant Account** trusts the Admin Account to perform protection operations of its resources.

Resources from the Admin account will utilize Amazon **STS:AssumeRole >** capability to gain temporary credentials allowing the protection of Tenant resources (EC2 instances, depicted above).

Supported features per use-case

The initial release of STS:AssumeRole capabilities is limited to protection of EC2 and RDS instances and utilization of native Amazon snapshots from protection and recovery operations. Please refer to the table below to match AssumeRole capabilities to your protection policies.

Feature matrix per use-case

Feature	Multiple account protection
Streaming backups **docs >**	
Full Instance Restores from backup copy **docs >**	
Attach Volume to existing instance or New Instance **docs >**	
Live browse and Guest File Level Restores / Agentless Restores / Download of files **docs >**	
Conversions from VMware / Hyper-V to AWS using Import/Hot add Method **docs >**	
Conversion from Azure to AWS **docs >**	
Live Sync/Disaster Recovery from **VMWare to AWS >**	
Live Sync/DR from AWS to AWS within the Same/cross region **doc >**	
File Indexing for Virtual Machines **docs >**	
Auto Scaling of Amazon Access Nodes **docs >**	
IntelliSnap® Backups **docs >**	✓
Full Instance Restores **docs >**	✓
Attach Volume to **existing >** instance or **New Instance >**	
Live browse and Guest File Level Restores / Agentless Restores / Download of files **docs >**	
Snap Replication (copy) from one region to another region **docs >**	✓
Snap Replication and sharing to different account and making a copy of the snap **docs >**	

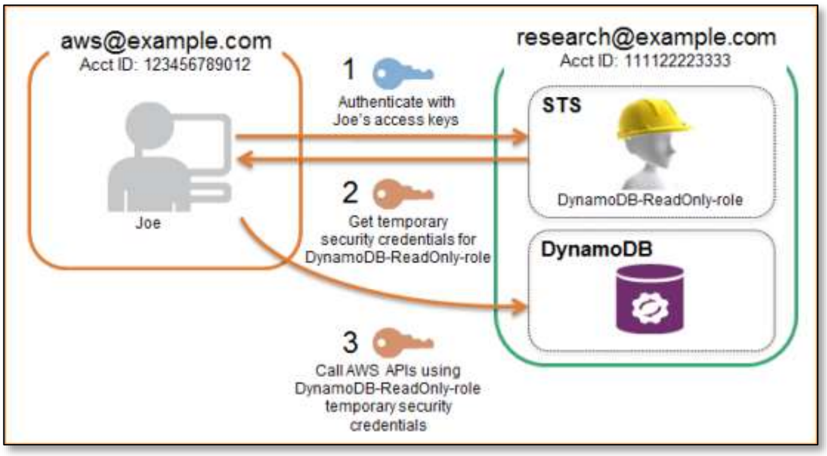
* Supported features are supported from both Admin and Tenant account

How does AssumeRole work?

Before starting the configuration of your STS:AssumeRole setup it is important to understand how AssumeRole functions. Amazon explains the process of using IAM Roles to delegates permissions to your resources here:

Creating a Role to Delegate Permissions to an IAM User >

It is always easier to understand concepts with a picture, *Delegating API Access to AWS Services Using IAM Roles* takes the explanation one step further by explaining how a single user account can access the resources of another (see below)


aws@example.com
Acct ID: 123456789012
Joe
1 Authenticate with Joe's access keys
2 Get temporary security credentials for DynamoDB-ReadOnly-role
3 Call AWS APIs using DynamoDB-ReadOnly-role temporary security credentials
research@example.com
Acct ID: 111122223333
STS
DynamoDB-ReadOnly-role
DynamoDB

Delegating API Access to AWS Services Using IAM Roles >

In the example above, the following occurs:

1. research@example.com is the **trusting entity** and has granted ReadOnly access to its owned DynamoDB instance.
2. aws@example.com is the **trusted entity** which has been trusted to use STS to temporarily perform ReadOnly actions on the DynamoDB account (with the permissions granted in the DynamoDB-ReadOnly-Role **Role**).
3. Joe calls **STS:AssumeRole** with his access key/secret key. STS service grants Joe a temporary access key (for 1 hour) with DynamoDB-ReadOnly-role permissions.
4. Joe then uses these temporary credentials to make API calls against the DynamoDB instance.

It is important to note that:

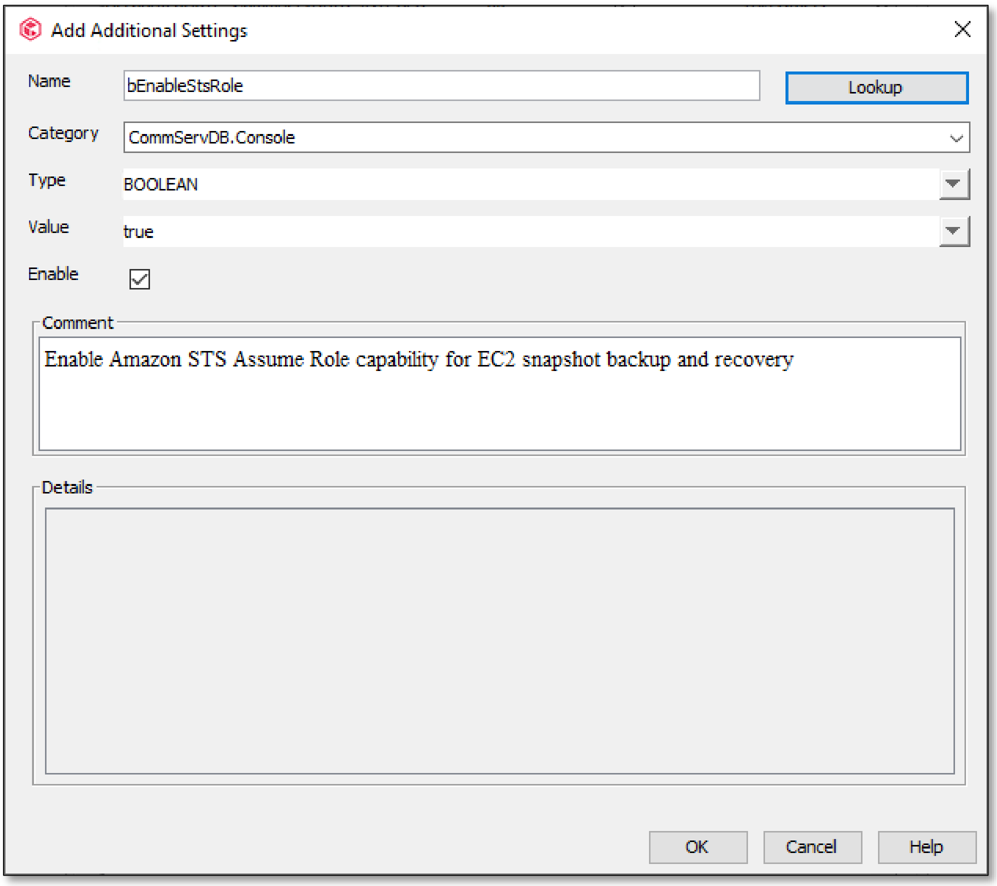
- The ***trusting entity*** (Acct ID 111122223333) must explicitly trust the account(s) it will issue AssumeRole credentials to.
- The credentials returns by STS:AssumeRole are **temporary** and expire after one hour (default, may be changed).
- The **permissions** that are granted are defined by the ***trusting entity***, allowing granular control over what ***trusted entities*** can perform.

Activating AssumeRole Functionality in Commvault

Amazon STS:AssumeRole functionality is still within **Technology Preview** (at the time of writing) and requires activation through the use of a Commvault **additional setting >**.

Perform the following to activate AssumeRole capability

1. Login to your **CommServe** as an administrator.
2. Open the **CommCell Console** and login as a Commvault administrator.
3. Right-click your CommServe name (root of CommCell Brower) > **Properties.**
4. Click the **Additional Settings** tab.
5. Click **Add** button.
6. Enter **bEnableStsRole**, BOOLEAN (True) for the **CommServDB.Console** category (see below).
7. Click **OK** to save the new additional setting.



Add Additional Settings

Name: bEnableStsRole

Category: CommServDB.Console

Type: BOOLEAN

Value: true

Enable: ☑

Comment: Enable Amazon STS Assume Role capability for EC2 snapshot backup and recovery

Details:

8. Click **OK** to close the CommCell properties window
9. Logout of **CommCell Console**, you will complete the remaining tasks in **Commvault Command Center**.

Note: This configuration step is temporary and will be removed after **General Availability** release of AssumeRole functionality in a later release.

Configuring AssumeRole for multiple accounts

To configure AssumeRole for Commvault protection operations where the ***trusting entity*** and ***trusted entity*** resources are in separate Amazon accounts, perform the following.

1. Configure **trusted entity** account to perform AssumeRole (**Admin** Account)
2. Configure **trusting entity** account(s) to allow trusted actions (**Tenant** Accounts)
3. Configure a Commvault Hypervisor and VM Group for protection of **trusting entity** resources.

The following steps assume you have met **software prerequisites >** and have enabled AssumeRole.

Configure trusted entity account

Your ***trusted entity*** account will own your S3 Cloud Libraries, MediaAgents and Access Nodes. This account will perform all of its data access actions with the temporary credentials received from the **trusting entity** resources. Additionally, this account is utilized to discover **regions**, **availability zones**, and perform data protection write activities so **amazon_restricted_role_permissions.json >** permissions are required as well.

To configure your ***trusted entity*** (**Admin**) account:

1. Login to the AWS Management Console -or- open the **aws cli**
2. Create a new **IAM Policy** (i.e. sts-only) with a single permission – **sts:AssumeRole** (see policy JSON below)

amazon_sts_assumerole.json

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "sts:AssumeRole",
            "Resource": "*"
        }
    ]
}
```

For example, to create the policy using the **aws command line tools**

```
aws iam create-policy --policy-name sts-only --policy-document
file://amazon_sts_assumerole.json
{
    "Policy": {
        "PolicyName": "sts-only",
        "PolicyId": "ANPAQZ4OJ6HUUFAZZEALF",
        "Arn": "arn:aws:iam::xxxxxxxxxxxx:policy/sts-only",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 0,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2020-06-24T01:46:30+00:00",
        "UpdateDate": "2020-06-24T01:46:30+00:00"
    }
}
```

Note: xxxxxxxxxxxx will contain the unique Amazon account Id of your **Admin** account.

3. Create a new **IAM Policy** with the Commvault **restricted backup & recovery permissions** (See **amazon_restricted_role_permissions.json >** for syntax – omitted here due to length)

```
aws iam create-policy --policy-name amazon_restricted_role --policy-document
file://amazon_restricted_role_permissions.json
{
    "Policy": {
        "PolicyName": "amazon_restricted_role",
        "PolicyId": "ANPAQZ4OJ6HUYC3ZTJSXN",
        "Arn": "arn:aws:iam::xxxxxxxxxxxx:policy/amazon_restricted_role",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 0,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2020-06-24T01:51:17+00:00",
        "UpdateDate": "2020-06-24T01:51:17+00:00"
    }
}
```

Note: xxxxxxxxxxxx will contain the unique Amazon account Id of your **Admin** account.

4. Create a new **IAM Role** and supplying an initial **Assume Role Policy** (json below)

admin-iam-role-policy.json

```
{
    "Version": "2012-10-17",
    "Statement": {
        "Effect": "Allow",
        "Action": "sts:AssumeRole",
       "Principal": {
           "Service": "ec2.amazonaws.com",
           "AWS": "arn:aws:iam::xxxxxxxxxxxx:root"
           }
     }
}
```

Note: Replace xxxxxxxxxxxx with the Account ID of your Amazon **Admin account**.

```
aws iam create-role --role-name vsa-assume-role --assume-role-policy-document
file:// admin-iam-role-policy.json
```

```
{
    "Role": {
        "Path": "/",
        "RoleName": "vsa-assume-role",
        "RoleId": "AROAQZ4OJ6HUZDMWSHUBQ",
        "Arn": "arn:aws:iam::xxxxxxxxxxxx:role/vsa-assume-role",
        "CreateDate": "2020-06-24T01:52:43+00:00",
        "AssumeRolePolicyDocument": {
            "Version": "2012-10-17",
            "Statement": {
                "Effect": "Allow",
                "Action": "sts:AssumeRole",
                "Principal": {
                    "Service": "ec2.amazonaws.com",
                    "AWS": "arn:aws:iam::xxxxxxxxxxxx:root"
                }
            }
        }
    }
}
```

Note: xxxxxxxxxxxx will contain the unique Amazon Account ID of your **Admin account**.

Now attach your newly created policies to allow the ***trusted entity*** to call STS:AssumeRole and perform basic discovery actions within the Amazon region.

```
aws iam attach-role-policy --role-name vsa-assume-role --policy-arn
"arn:aws:iam::055596151273:policy/sts-only"
aws iam attach-role-policy --role-name vsa-assume-role --policy-arn
"arn:aws:iam::055596151273:policy/amazon_restricted_role"
```

5. Create an **Instance Profile >** from your newly created IAM Role. This step is not required when utilizing the Amazon Management Console as the instance profile is created automatically.

```
aws iam create-instance-profile --instance-profile-name vsa-assume-role
aws iam add-role-to-instance-profile --instance-profile-name vsa-assume-role --
role-name vsa-assume-role
```

6. Attach the new role you each of the **Amazon Access Nodes** that well be performing protection operations.

```
aws ec2 associate-iam-instance-profile --instance-id i-0d6a2d710af044a19 --iam-instance-profile Name=vsa-assume-role
{
    "IamInstanceProfileAssociation": {
        "AssociationId": "iip-assoc-03a0a1f6409b05ca5",
        "InstanceId": "i-0d6a2d710af044a19",
        "IamInstanceProfile": {
            "Arn": "arn:aws:iam::xxxxxxxxxxxx:instance-profile/vsa-assume-role",
            "Id": "AIPAQZ4OJ6HU74Y26MDSV"
        },
        "State": "associating"
    }
}
```

It may take a couple of seconds for Amazon to associate the instance profile to your running **Access Node**. You can check status with the command below, when completed the status will show **associated**.

```
aws ec2 describe-iam-instance-profile-associations --association-ids  iip-assoc-03a0a1f6409b05ca5
{
    "IamInstanceProfileAssociations": [
        {
            "AssociationId": "iip-assoc-03a0a1f6409b05ca5",
            "InstanceId": "i-0d6a2d710af044a19",
            "IamInstanceProfile": {
                "Arn": "arn:aws:iam::xxxxxxxxxxxx:instance-profile/vsa-assume-role",
                "Id": "AIPAQZ4OJ6HU74Y26MDSV"
            },
            "State": "associated"
        }
    ]
}
```

If your EC2 instance already has an activate association, use the following syntax to perform a **reassociation** of your new instance profile with your EC2 based Access Node.

Note: You can obtain the required **AssociationId** using the **aws ec2 describe-iam-instance-profile-associations >** command.

```
aws ec2 replace-iam-instance-profile-association  --iam-instance-profile
Name=vsa-assume-role --association-id iip-assoc-02111dd45ec6f7743

{
    "IamInstanceProfileAssociation": {
        "AssociationId": "iip-assoc-03a0a1f6409b05ca5",
        "InstanceId": "i-0d6a2d710af044a19",
        "IamInstanceProfile": {
            "Arn": "arn:aws:iam::xxxxxxxxxxxx:instance-profile/vsa-assume-role",
            "Id": "AIPAQZ4OJ6HU74Y26MDSV"
        },
        "State": "associating"
    }
}
```

Note: An Amazon Access Node is the host or host(s) that you utilize within your Admin account hypervisor within the Command Command Center (more below).

Configure trusting entity account(s)

Now that you have completed the setup of your **trusted entity**, you can configure each of the accounts that have resources that require data management. Each of these accounts needs to explicitly trust the ***trusted entity*** account to request temporary access credentials via STS:AssumeRole.

To configure your ***trusting entity*** account(s):

1. Login to the AWS Management Console -or- open the aws cli
2. Create a new **IAM Policy** with a single permission – **sts:AssumeRole** (see policy JSON below)

amazon_sts_assumerole.json

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "sts:AssumeRole",
            "Resource": "*"
        }
    ]
}
```

For example to create the policy using the **aws command line tools**

```
aws iam create-policy --policy-name sts-only --policy-document
file://amazon_sts_assumerole.json
{
    "Policy": {
        "PolicyName": "sts-only",
        "PolicyId": "ANPA6RPGBN5MXOUSCZVBI",
        "Arn": "arn:aws:iam::xxxxxxxxxxxx:policy/sts-only",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 0,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2020-06-24T02:12:55+00:00",
        "UpdateDate": "2020-06-24T02:12:55+00:00"
    }
}
```

Note: xxxxxxxxxxxx will contain the unique Amazon Account ID of your **Tenant account**.

3. Create a new **IAM Policy** with the Commvault **restricted backup & recovery permissions** (See **amazon_restricted_role_permissions.json >** for syntax – omitted here due to length)

```
aws iam create-policy --policy-name amazon_restricted_role --policy-document
file://amazon_restricted_role_permissions.json
{
    "Policy": {
        "PolicyName": "amazon_restricted_role",
        "PolicyId": "ANPA6RPGBN5MYST2DYKZZ",
        "Arn": "arn:aws:iam::xxxxxxxxxxxx:policy/amazon_restricted_role",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 0,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2020-06-24T02:14:18+00:00",
        "UpdateDate": "2020-06-24T02:14:18+00:00"
    }
}
```

Note: xxxxxxxxxxxx will contain the unique Amazon Account ID of your **Tenant account**.

4. Create a new **IAM Role** (i.e. **vsa-role**) and attach the previously created IAM polices (x2)

amazon_sts_assumerole_tenant.json

```
{
    "Version": "2012-10-17",
    "Statement": {
        "Effect": "Allow",
        "Action": "sts:AssumeRole",
       "Principal": {
           "Service": "ec2.amazonaws.com",
           "AWS": "arn:aws:iam::xxxxxxxxxxxx:root"
           }
     }
}
```

Note: The highlighted section xxxxxxxxxxxxx must be replaced with the **Account ID** of the AWS **Admin** account you want to trust to temporarily access resources in the current account.

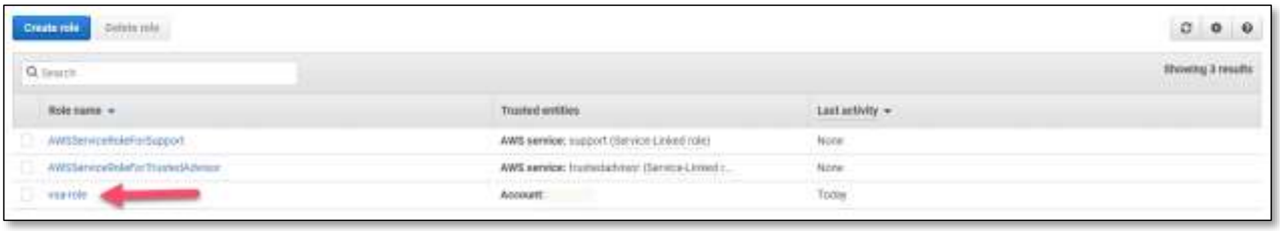
```
aws iam create-role --role-name VSA_Role2 --assume-role-policy-document
file://amazon_sts_assumerole_tenant.json
{
    "Role": {
        "Path": "/",
        "RoleName": "vsa-role",
        "RoleId": "AROA6RPGBN5M6TFOHP6M3",
        "Arn": "arn:aws:iam::xxxxxxxxxxxx:role/vsa-role",
        "CreateDate": "2020-06-24T02:15:45+00:00",
        "AssumeRolePolicyDocument": {
            "Version": "2012-10-17",
            "Statement": {
                "Effect": "Allow",
                "Action": "sts:AssumeRole",
                "Principal": {
                    "Service": "ec2.amazonaws.com",
                    "AWS": "arn:aws:iam::XXXXXXXXXXXX:root"
                }
            }
        }
    }
}
```

Note: xxxxxxxxxxxx will contain the unique Amazon Account ID of your **Tenant account**.
Note2: XXXXXXXXXXXX will contain the unique Amazon Account ID of your **Admin account**.

Now attach your newly created policy to allow the ***trusted entity*** to call STS:AssumeRole

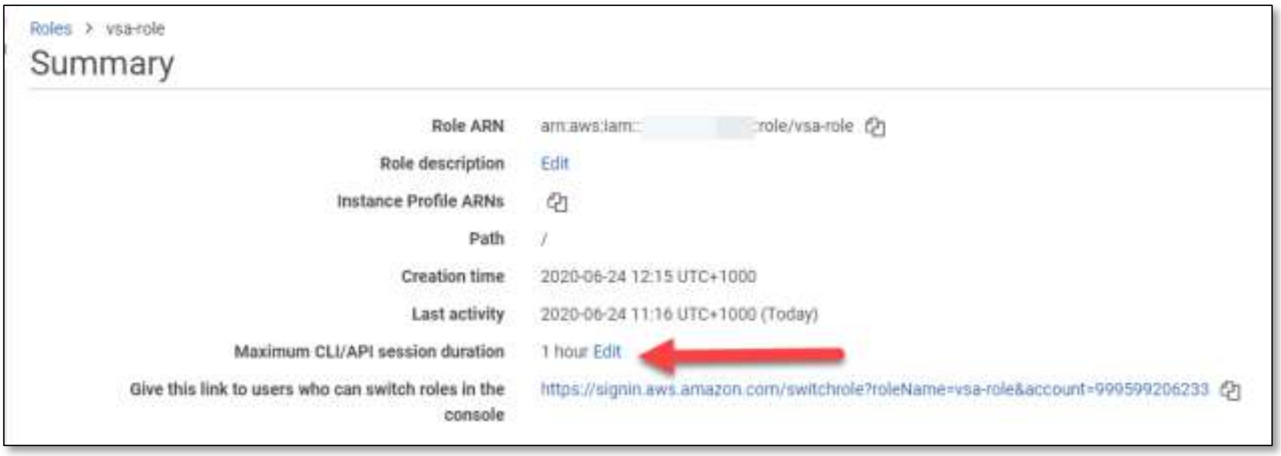
```
aws iam attach-role-policy --role-name vsa-role --policy-arn
"arn:aws:iam::999599206233:policy/sts-only"

aws iam attach-role-policy --role-name vsa-role --policy-arn
"arn:aws:iam::999599206233:policy/amazon_restricted_role"
```

Your **Tenant Role** will now be visible in the Amazon Management Console



Role name	Trusted entities	Last activity
AWSServiceRoleForSupport	**AWS service:** support (Service-Linked role)	None
AWSServiceRoleForTrustedAdvisor	**AWS service:** trustedadvisor (Service-Linked r...	None
vsa-role	**Account:** [redacted]	Today

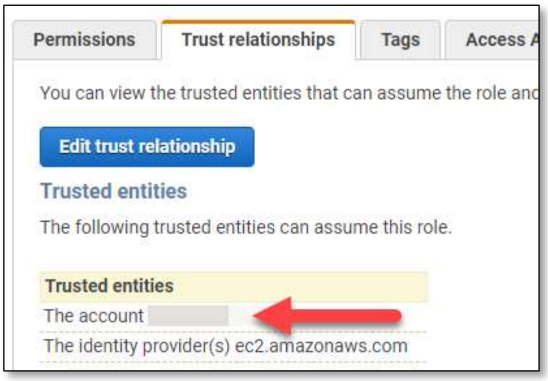
By default any credentials issued from this account will expire after **1 hour**.


Roles > vsa-role

Summary

Role ARN	arn:aws:iam::[redacted]:role/vsa-role
Role description	Edit
Instance Profile ARNs	
Path	/
Creation time	2020-06-24 12:15 UTC+1000
Last activity	2020-06-24 11:16 UTC+1000 (Today)
Maximum CLI/API session duration	1 hour Edit
Give this link to users who can switch roles in the console	https://signin.aws.amazon.com/switchrole?roleName=vsa-role&account=999599206233

You can observe the **trust relationship** from your Tenant Admin to your **Admin account** by checking that the Trusted Entities lists your **Admin Account Id**



You can view the trusted entities that can assume the role and

Trusted entities

The following trusted entities can assume this role.

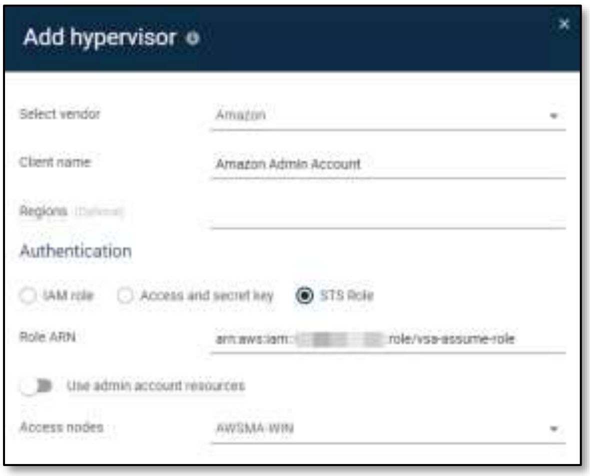
Trusted entities
The account [redacted]
The identity provider(s) ec2.amazonaws.com

Configure Commvault Hypervisor and VM Group

You can now configure Commvault for both the **Admin** account (***trusted*** entity) and the Tenant account(s) (trusting entities).

Adding the Admin Hypervisor

1. Login to Command Center at https://YOUR_COMMSERVE/adminconsole as **admin**
2. Navigate to Protect > **Virtualization**
3. Click **Add Hypervisor** button (right)
4. Complete the following details and click **Save** button (see screenshot below)
    a. Vendor = **Amazon**
    b. Client name = free form name for hypervisor
    c. Regions (leave blank)
    d. Authentication = **STS Role** (you can use **IAM Role** or **Access and secret key**)
    e. Role ARN = **Enter the ARN of the trusted entity** (the **Admin** account)
    f. Use admin account resources – **Must be disabled**
    g. Access nodes = **Select the access node you attached the vsa-assume-role to**



Add hypervisor

Select vendor: Amazon

Client name: Amazon Admin Account

Regions (Optional):

Authentication

○ IAM role ○ Access and secret key ● STS Role

Role ARN: arn:aws:iam::[redacted]:role/vsa-assume-role

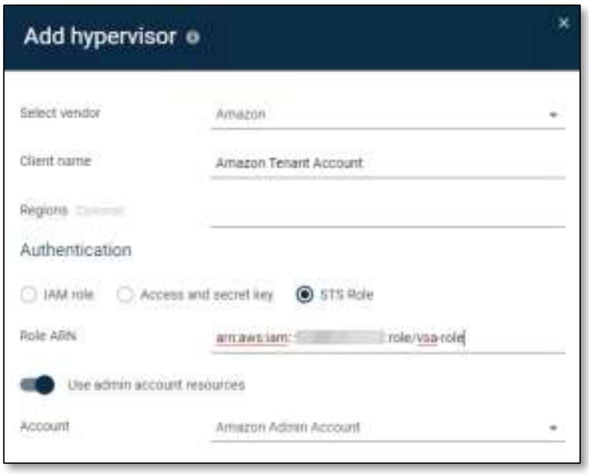
Use admin account resources

Access nodes: AWSMA-WIN

5. Click **Skip for now** on the Add VM Group page.

Adding the Tenant Hypervisor

1. Click **Add Hypervisor** button (right)
2. Complete the following details and click **Save** button (see screenshot below)
   a. Vendor = **Amazon**
   b. Client name = free form name for hypervisor
   c. Regions (leave blank)
   d. Authentication = **STS Role**
   e. Role ARN = **Enter the ARN of the trusting entity** (the **Tenant** account)
   f. Use admin account resources – **Must be enabled**
   g. Access nodes = **Select the access node you attached the vsa-assume-role to**



Add hypervisor

Select vendor: Amazon

Client name: Amazon Tenant Account

Regions Optional

Authentication

○ IAM role ○ Access and secret key ● STS Role

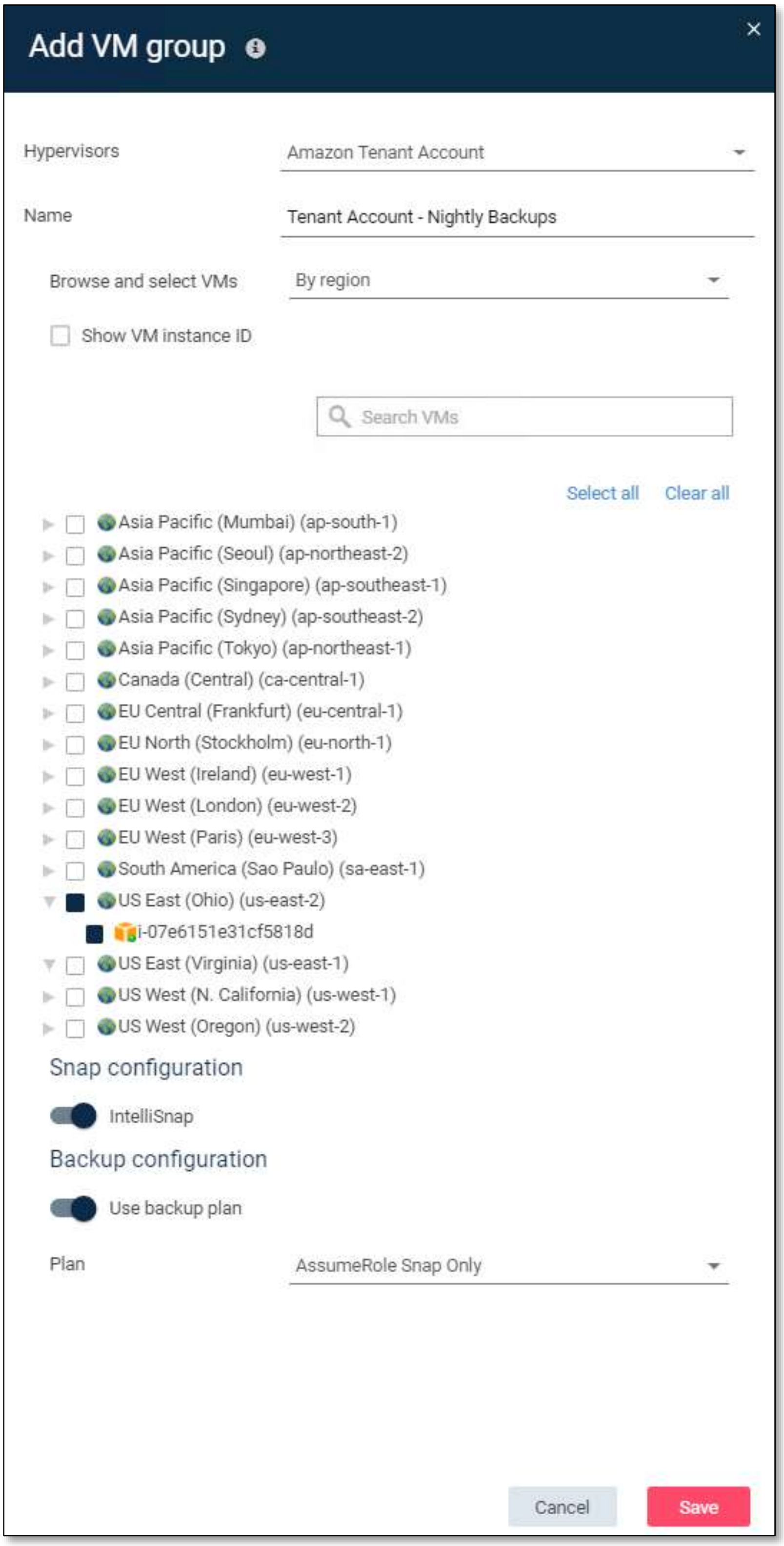
Role ARN: arn:aws:iam:[redacted]role/vsa-role

Use admin account resources: ON

Account: Amazon Admin Account

3. You will be redirected back to the **Hypervisors** page, click the **VM Groups** tab (top)
4. Click **Add VM group** button (right)
5. Select your **Amazon Tenant Account** Hypervisor from the drop-down.
6. Enter the following and click **Save** button
   a. Enter a description **Name** for the VM group
   b. Select the **Instances / Regions / AZs** you want to protect
   c. Ensure **IntelliSnap** toggle is **ON**
   d. Ensure **Use Backup Plan** toggle is **ON**
   e. Select an appropriate Plan for the **RPO** you want to create backups for.

Note: Remember that AssumeRole backups do not perform a backup copy. Commvault recommends creating a new Plan that has **Backup Copy** disabled.



Add VM group

Hypervisors: Amazon Tenant Account

Name: Tenant Account - Nightly Backups

Browse and select VMs: By region

□ Show VM instance ID

- □ Asia Pacific (Mumbai) (ap-south-1)
- □ Asia Pacific (Seoul) (ap-northeast-2)
- □ Asia Pacific (Singapore) (ap-southeast-1)
- □ Asia Pacific (Sydney) (ap-southeast-2)
- □ Asia Pacific (Tokyo) (ap-northeast-1)
- □ Canada (Central) (ca-central-1)
- □ EU Central (Frankfurt) (eu-central-1)
- □ EU North (Stockholm) (eu-north-1)
- □ EU West (Ireland) (eu-west-1)
- □ EU West (London) (eu-west-2)
- □ EU West (Paris) (eu-west-3)
- □ South America (Sao Paulo) (sa-east-1)
- ■ US East (Ohio) (us-east-2)
  - ■ i-07e6151e31cf5818d
- □ US East (Virginia) (us-east-1)
- □ US West (N. California) (us-west-1)
- □ US West (Oregon) (us-west-2)

Snap configuration

● IntelliSnap

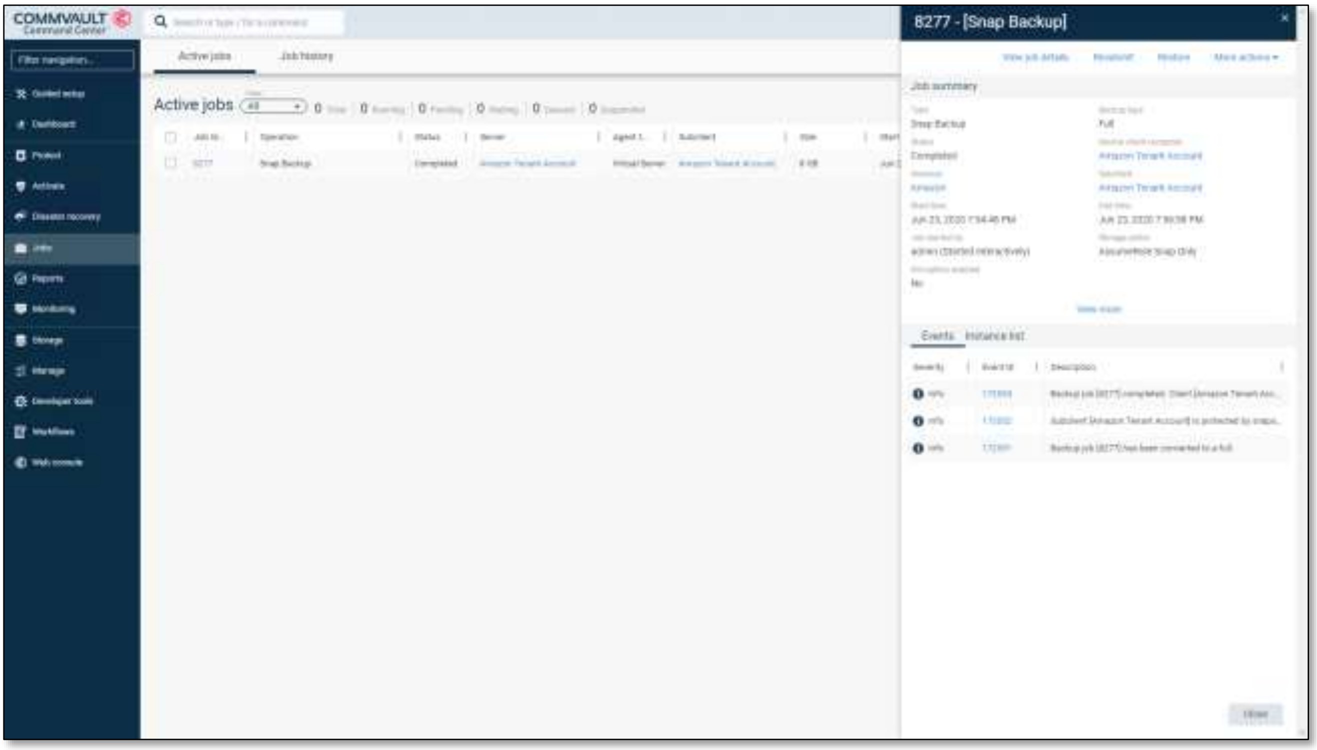
Backup configuration

● Use backup plan

Plan: AssumeRole Snap Only

7. Click **Back up** (right) to perform your first AssumeRole enabled backup, leave defaults, click **OK**

You may observe snapshot creation in the AWS Management Console



8277 - [Snap Backup]

Job summary

Snap Backup

Full

Completed

Amazon Tenant Account

Amazon

Amazon Tenant Account

Jun 23, 2020 7:54:46 PM

Jun 23, 2020 7:59:58 PM

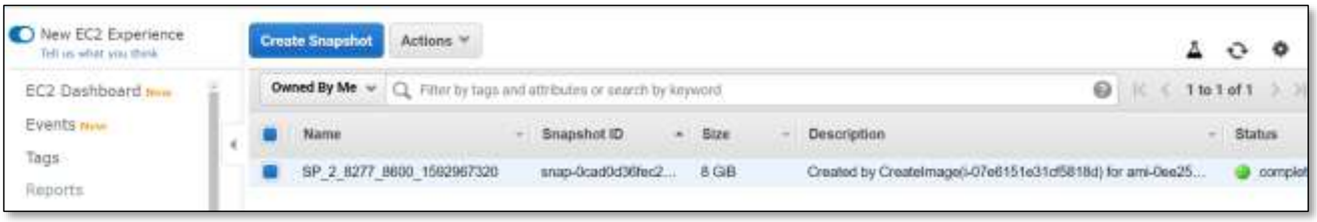
admin (Started interactively)

No

Events | Instance list

Severity	Event ID	Description
Info	[illegible]	Backup job [8277] completed. Client [Amazon Tenant Acc...
Info	[illegible]	Subclient [Amazon Tenant Account] is protected by snaps...
Info	[illegible]	Backup job [8277] has been converted to a full.

Job ID	Operation	Status	Server	Agent t...	Subclient	Size
8277	Snap Backup	Completed	Amazon Tenant Account	Virtual Server	Amazon Tenant Account	8 GB



Name	Snapshot ID	Size	Description	Status
SP_2_8277_8600_1592967320	snap-0cad0d36fec2...	8 GiB	Created by CreateImage(i-07e6151e31d5818d) for ami-0ee25...	complet

Troubleshooting

All actions that are performed during a STS:AssumeRole backup are logged to the **Access Node** performing the backup.

Please consult the following log files during troubleshooting:

- **`cvd.log`**
- **`vsdiscovery.log`**
- **`vsbkp.log`**
- **`vsrst.log`**

These log files may be found in the **Log Files** directory on your Access Node:

- Windows (default): `C:\Program Files\Commvault\ContentStore\Log Files`
- UNIX/Linux: `/var/log/commvault/Log_Files`

Example log output

During a successfully STS:AssumeRole backup or recovery operation you can observe the use of temporary credential request and usage via the following log entries:

```
960f6c   05/09 11:24:04 16972 CAmazonInfo::Connect() - Connecting to Url=[default],
username[arn:aws:iam::xxxxxxxxxxxx:role/VSA_Role]
```

Indicates that Commvault is attempting to connect to Url with the specified **Amazon Resource Name (ARN) >**, the ARN will refer to the **IAM Role** that is configured within the **Tenant Account** (i.e. An IAM role called **VSA_Role**)

```
6924  f48   05/08 20:07:24 16960 CAmazonInfo::Connect() - Connecting to Url=[default],
user name[arn:aws:iam:: xxxxxxxxxxxx :role/VSA_Role]6924  f48   05/08 20:07:24 16960
```

The 12-digit identifier will identify the specific Amazon account that is being utilized for the action.

```
6924f48   05/08 20:07:24 16960 AmazonConnection::ValidateCredentials() -
Assume role [arn:aws:iam:: xxxxxxxxxxxx  :role/VSA_Role]
```

Indicates that a validation of the supplied ARN role for backup purposes is being validated. Commvault will validate that credentials may be requested, received and utilized to perform backup and recovery actions.

Amazon Backup Permissions JSON

The following JSON may be used to define the backup and recovery IAM role for admin & tenant accounts **source >**

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "ec2:DeleteVolume",
                "ec2:DeleteSnapshot"
            ],
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ec2:ResourceTag/_GX_BACKUP_": "*"
                }
            }
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": "ec2:DeleteTags",
            "Resource": "*",
            "Condition": {
                "ForAnyValue:StringEquals": {
                    "aws:TagKeys": [
                        "CV_Retain_Snap",
                        "CV_Integrity_Snap",
                        "_GX_BACKUP_",
                        "_GX_AMI_",
                        "Name",
                        "Description"
                    ]
                }
            }
        },
        {
            "Sid": "VisualEditor2",
            "Effect": "Allow",
            "Action": "ec2:DetachVolume",
            "Resource": "arn:aws:ec2:*:*:volume/*",
            "Condition": {
                "StringLike": {
                    "ec2:ResourceTag/_GX_BACKUP_": "*"
                }
            }
        },
        {
            "Sid": "VisualEditor3",
            "Effect": "Allow",
            "Action": "ec2:TerminateInstances",
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ec2:ResourceTag/CV_Integrity_Snap": "*"
                }
            }
        },
        {
            "Sid": "VisualEditor4",
            "Effect": "Allow",
            "Action": "ec2:TerminateInstances",
            "Resource": "*",
            "Condition": {
                "StringLike": {
                    "ec2:ResourceTag/_GX_BACKUP_": "*"
                }
            }
        },
        {
            "Sid": "VisualEditor5",
            "Effect": "Allow",
            "Action": "ec2:DetachVolume",
            "Resource": "arn:aws:ec2:*:*:instance/*"
        },
        {
            "Sid": "VisualEditor6",
            "Effect": "Allow",
            "Action": [
                "ssm:CancelCommand",
                "ssm:SendCommand",
                "ec2:CopySnapshot",
                "ec2:AuthorizeSecurityGroupIngress",
                "kms:Decrypt",
                "ec2:ModifyVolumeAttribute",
                "ec2:DescribeInstances",
                "ssm:ListCommands",
                "ec2:CreateKeyPair",
                "ec2:DescribeVolumesModifications",
                "ec2:CreateImage",
                "kms:DescribeKey*",
                "s3:CreateBucket",
                "ec2:DescribeSnapshots",
                "kms:ReEncrypt*",
                "kms:GenerateDataKey*",
                "ec2:ModifySnapshotAttribute",
                "ec2:ModifyImageAttribute",
                "ec2:StartInstances",
                "kms:Encrypt",
                "ec2:DescribeVolumes",
                "ec2:DescribeAccountAttributes",
                "s3:PutBucketAcl",
                "s3:PutObjectTagging",
                "ssm:ListDocuments",
                "iam:GetAccountAuthorizationDetails",
                "ec2:ImportImage",
                "ec2:DescribeKeyPairs",
                "s3:DeleteObject",
                "iam:GetRole",
                "ec2:ModifyVolume",
                "ec2:CreateTags",
                "ec2:ModifyNetworkInterfaceAttribute",
                "ec2:DeleteNetworkInterface",
                "ec2:RunInstances",
                "ec2:StopInstances",
                "ec2:DescribeVolumeAttribute",
                "ec2:CreateVolume",
                "ec2:CreateNetworkInterface",
                "s3:PutObject",
                "s3:GetObject",
                "ec2:DisassociateIamInstanceProfile",
                "ec2:DescribeSubnets",
                "ec2:AttachVolume",
                "ec2:DeregisterImage",
                "ssm:DescribeDocument",
                "ec2:DescribeInstanceAttribute",
                "ec2:DescribeRegions",
                "kms:GenerateDataKeyWithoutPlaintext",
                "kms:ListResourceTags",
                "s3:ListBucket",
               "ec2:ModifyImageAttribute",
               "ebs:ListChangedBlocks",
                "ec2:GetConsoleOutput",
                "iam:PassRole",
               "kms:TagResource",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeAvailabilityZones",
                "ec2:DescribeNetworkInterfaceAttribute",
                "ec2:CreateSecurityGroup",
                "ec2:CreateSnapshot",
                "kms:CreateKey",
                "ec2:ModifyInstanceAttribute",
                "kms:CreateGrant",
                "s3:PutObjectAcl",
                "ec2:DescribeInstanceStatus",
                "ec2:AuthorizeSecurityGroupEgress",
                "ec2:DetachNetworkInterface",
                "ec2:DescribeIamInstanceProfileAssociations",
                "ec2:DescribeTags",
                "iam:ListRoles",
               "ebs:ListSnapshotBlocks",
                "ec2:DescribeImportImageTasks",
                "iam:ListInstanceProfiles",
                "s3:GetBucketAcl",
                "kms:ListGrants",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeImages",
                "kms:ListKeys",
                "s3:ListAllMyBuckets",
                "ssm:DescribeInstanceInformation",
                "ec2:DescribeVpcs",
                "kms:ListAliases",
                "kms:CreateAlias",
                "ec2:AttachNetworkInterface",
                "iam:GetUser",
                "s3:GetBucketLocation",
                "ec2:AssociateIamInstanceProfile"
            ],
            "Resource": "*"
        }
    ]
}
```

Additional Resources

Documentation

Amazon Cloud Architecture Guide (CAG)

The Public Cloud Architecture Guide for Amazon is published with each Commvault Feature Release and represents the current best practices from Commvault and Amazon.

Public Cloud Architecture Guide for Amazon Web Services >

Books Online – Cloud Storage

The **Cloud Storage >** section from the Commvault® Books Online documentation covers technical procedures and information on Supported Cloud Targets, Advanced procedures, Troubleshooting and FAQ sections for Commvault customers.

AWS IAM Permissions

All required Amazon user permissions can be accessed from Books Online here:

- **Amazon Web Services User Permissions for Backups and Restores >**
- **Amazon Web Services User Permissions for VM Conversion >**
- **Creating a Role with Restricted Access >**