Configure SCA to Ingest Multiple AWS Accounts through a Single AWS S3 Bucket

Contents

Introduction

This document describes how you configure an Amazon Web Services (AWS) Simple Storage Service (S3) to accept logs from a second AWS Account.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Secure Cloud Analytics
- AWS Identity Acess Management (IAM)
- AWS S3

Components Used

The information in this document is based on:

- AWS Account A (referred to as ACCOUNT_A_ID - This account host/owns the S3 buckets that already exist)
- AWS Account B (referred to as ACCOUNT_B_ID - This is a new (to Secure Cloud Analytics)

account that sends data to ACCOUNT_A_ID's S3_BUCKET_NAME)
- Secure Cloud Analytics (this must already be integrated with ACCOUNT_A_ID)

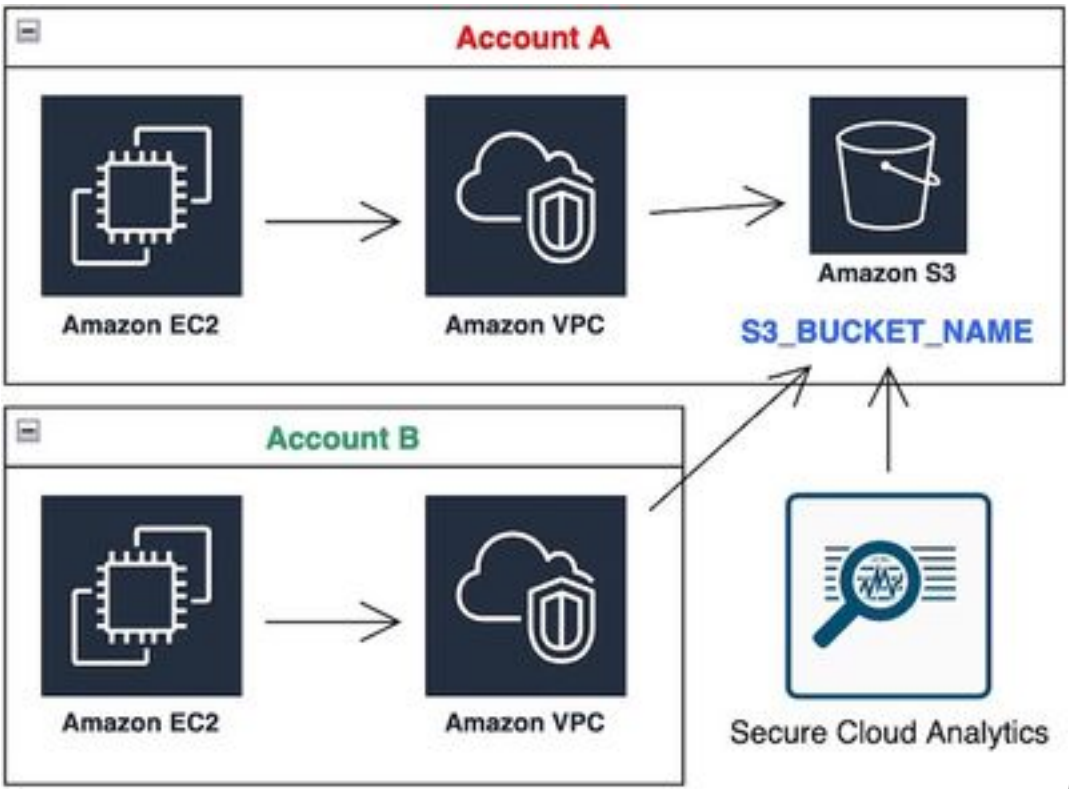
The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

Configure

There are five steps to have SCA ingest 2+ accounts from 1 S3 bucket:

1. Update **ACCOUNT_A_ID's S3_BUCKET_NAME** policy to grant **ACCOUNT_B_ID** account write permissions.
2. Configure the **ACCOUNT_B_ID** account to send VPC Flow Logs to **ACCOUNT_A_ID's S3_BUCKET_NAME**.
3. Create IAM Policy in **ACCOUNT_B_ID's** AWS IAM dashboard.
4. Create IAM Role in **ACCOUNT_B_ID's** AWS IAM dashboard.
5. Configure Secure Cloud Analytics Credentials for **ACCOUNT_B_ID**.

Network Diagram



Data Flow Diagram

Configurations

1. Update ACCOUNT_A_ID's S3_BUCKET_NAME Policy to Grant ACCOUNT_B_ID Account Write Permissions

ACCOUNT_A_ID's S3_BUCKET_NAME bucket policy configuration is provided here. This configuration allows a secondary (or any number of accounts you desire) account to write (SID-AWSLogDeliveryWrite) to the S3 bucket, and to check ACLs (SID - AWSLogDeliveryAclCheck) for

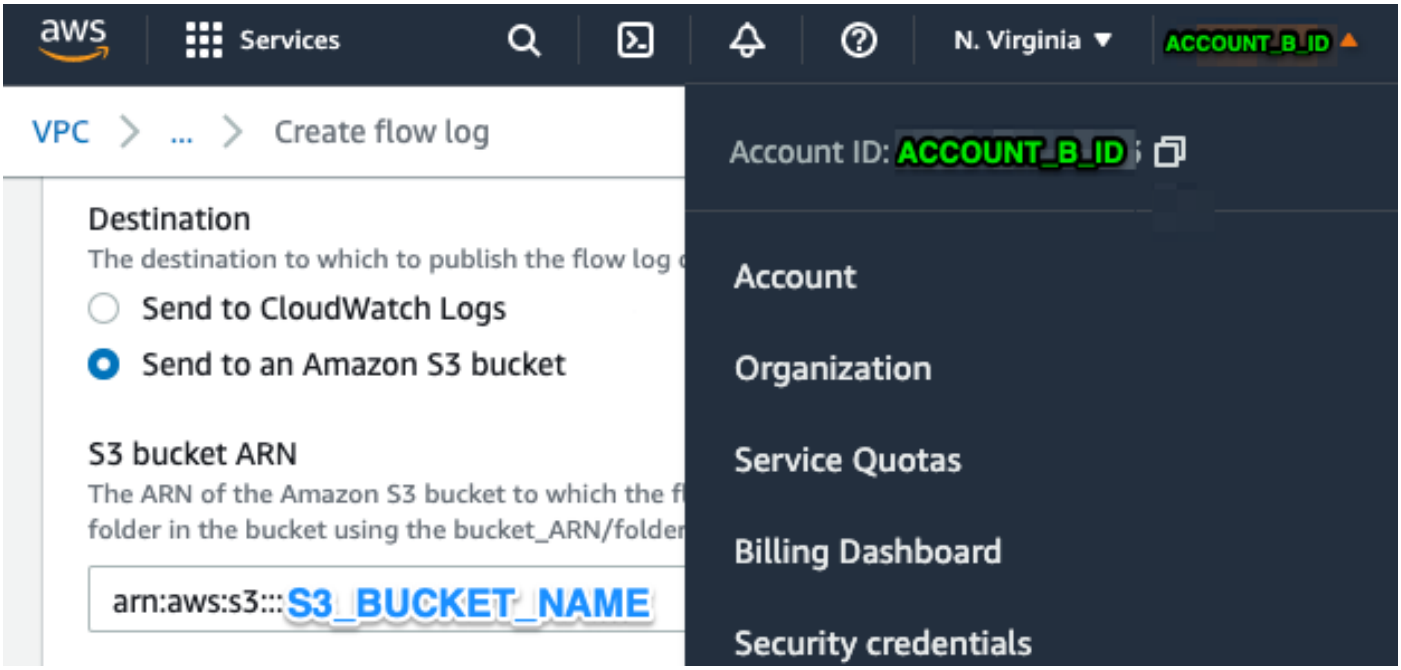
the bucket.

- Change **ACCOUNT_A_ID** and **ACCOUNT_B_ID** to their respective numerical values without dashes.
- Change **S3_BUCKET_NAME** to the respective bucket name.
- Ignore the formatting here, AWS can edit it as needed.

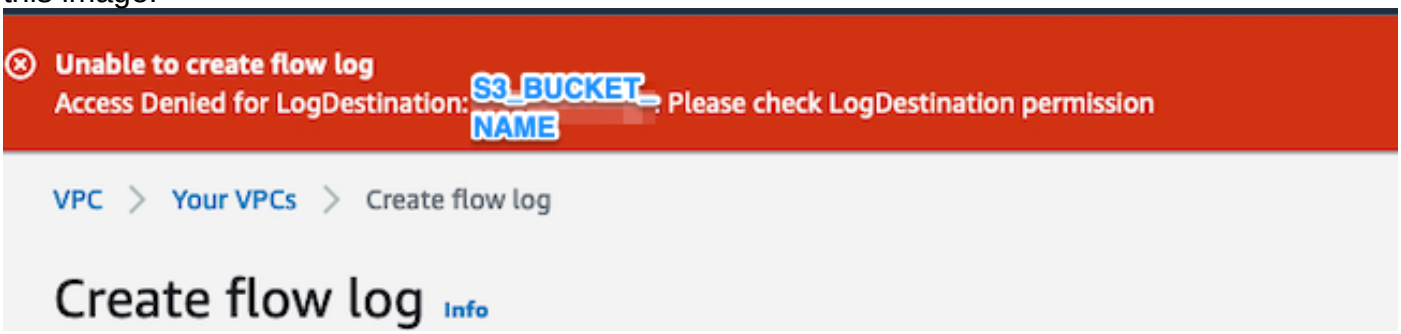
```
{
"Version": "2012-10-17",
"Statement": [
{
"Sid": "AWSLogDeliveryWrite",
"Effect": "Allow",
"Principal": {"Service": "delivery.logs.amazonaws.com"},
"Action": "s3:PutObject",
"Resource": ["arn:aws:s3:::S3_BUCKET_NAME","arn:aws:s3:::S3_BUCKET_NAME/*"],
"Condition": {
"StringEquals": {"aws:SourceAccount": ["ACCOUNT_A_ID","ACCOUNT_B_ID"]},
"ArnLike": {"aws:SourceArn": ["arn:aws:logs:*:ACCOUNT_A_ID:*","arn:aws:logs:*:ACCOUNT_B_ID:*"]}
}
},
{
"Sid": "AWSLogDeliveryAclCheck",
"Effect": "Allow",
"Principal": {
"Service": "delivery.logs.amazonaws.com"
},
"Action": "s3:GetBucketAcl",
"Resource": "arn:aws:s3:::S3_BUCKET_NAME",
"Condition": {
"StringEquals": {"aws:SourceAccount": ["ACCOUNT_A_ID","ACCOUNT_B_ID"]},
"ArnLike": {"aws:SourceArn": ["arn:aws:logs:*:ACCOUNT_A_ID:*","arn:aws:logs:*:ACCOUNT_B_ID:*"]}
}
}
]
}
```

2. Configure the ACCOUNT_B_ID Account to Send VPC Flow Logs to ACCOUNT_A_ID's S3_BUCKET_NAME

Create a VPC Flow Log **ACCOUNT_B_ID** that has **ACCOUNT_A_ID'sS3_BUCKET_NAME** bucket ARN into the destination as shown in this image:



If the permissions on the S3 bucket are not configured properly then you see an error similar to this image:



3. Create IAM Policy in ACCOUNT_B_ID's AWS IAM Dashboard

The IAM Policy configuration that is attached to the swc_role on **ACCOUNT_B_ID** is:

```
swc_single_policy
{
"Version": "2012-10-17",
"Statement": [
{
"Action": [
"autoscaling:Describe*",
"cloudtrail:LookupEvents",
"cloudwatch:Get*",
"cloudwatch:List*",
"ec2:Describe*",
"ecs:List*",
"ecs:Describe*",
"elasticache:Describe*",
"elasticache:List*",
"elasticloadbalancing:Describe*",
"guardduty:Get*",
"guardduty:List*",
"iam:Get*",
"iam:List*",
"inspector:*",
```

```
"rds:Describe*",
"rds:List*",
"redshift:Describe*",
"workspaces:Describe*",
"route53:List*"
],
"Effect": "Allow",
"Resource": "*"
},
{
"Action": [
"logs:Describe*",
"logs:GetLogEvents",
"logs:FilterLogEvents",
"logs:PutSubscriptionFilter",
"logs:DeleteSubscriptionFilter"
],
"Effect": "Allow",
"Resource": "*"
},
{
"Sid": "CloudCompliance",
"Action": [
"access-analyzer:ListAnalyzers",
"cloudtrail:DescribeTrails",
"cloudtrail:GetEventSelectors",
"cloudtrail:GetTrailStatus",
"cloudtrail:ListTags",
"cloudwatch:DescribeAlarmsForMetric",
"config:Get*",
"config:Describe*",
"ec2:GetEbsEncryptionByDefault",
"iam:GenerateCredentialReport",
"iam:Get*",
"iam:List*",
"kms:GetKeyRotationStatus",
"kms:ListKeys",
"logs:DescribeMetricFilters",
"logs:Describe*",
"logs:GetLogEvents",
"logs:FilterLogEvents",
"organizations:ListPolicies",
"s3:GetAccelerateConfiguration",
"s3:GetAccessPoint",
"s3:GetAccessPointPolicy",
"s3:GetAccessPointPolicyStatus",
"s3:GetAccountPublicAccessBlock",
"s3:GetAnalyticsConfiguration",
"s3:GetBucket*",
"s3:GetEncryptionConfiguration",
"s3:GetInventoryConfiguration",
"s3:GetLifecycleConfiguration",
"s3:GetMetricsConfiguration",
"s3:GetObjectAcl",
"s3:GetObjectVersionAcl",
"s3:GetReplicationConfiguration",
"s3:ListAccessPoints",
"s3:ListAllMyBuckets",
"securityhub:Get*",
"sns:ListSubscriptionsByTopic"
],
"Effect": "Allow",
"Resource": "*"
},
```

```
{
"Action": [
"s3:ListBucket",
"s3:GetBucketLocation",
"s3:GetObject"
],
"Effect": "Allow",
"Resource": [
"arn:aws:s3:::S3_BUCKET_NAME/*",
"arn:aws:s3:::S3_BUCKET_NAME"
]
}
]
}
```

4. Create IAM Role in ACCOUNT_B_ID's AWS IAM Dashboard

1. Select **Roles.**

2. Select **Create role.**

3. Select the Another AWS account role type.

4. Enter 757972810156 in the Account ID field.

5. Select the Require external ID option.

6. Enter your Secure Cloud Analytics web portal name as the **External ID.**
7. Click **Next: Permissions .**

8. Select the **swc_single_policy** policy that you just created.

9. Click **Next: Tagging**.

10. Click **Next: Review**.

11. Enter swc_role as the Role name.

12. Enter a **Description**, such as a Role to allow cross-account access.

13. Click **Create role** .

14. Copy the role ARN and paste it into a plaintext editor.

5. Configure Secure Cloud Analytics Credentials for ACCOUNT_B_ID

1. Log in to Secure Cloud Analytics and select **Settings > Integrations > AWS > Credentials**.

2. Click **Add New Credentials**.

3. For the **Name**, suggested naming schema would be **Account_B_ID_creds** (for example; 012345678901_creds) for each account, you wish to ingest.

4. Paste the role ARN from the previous step and paste it into the **Role ARN** field.

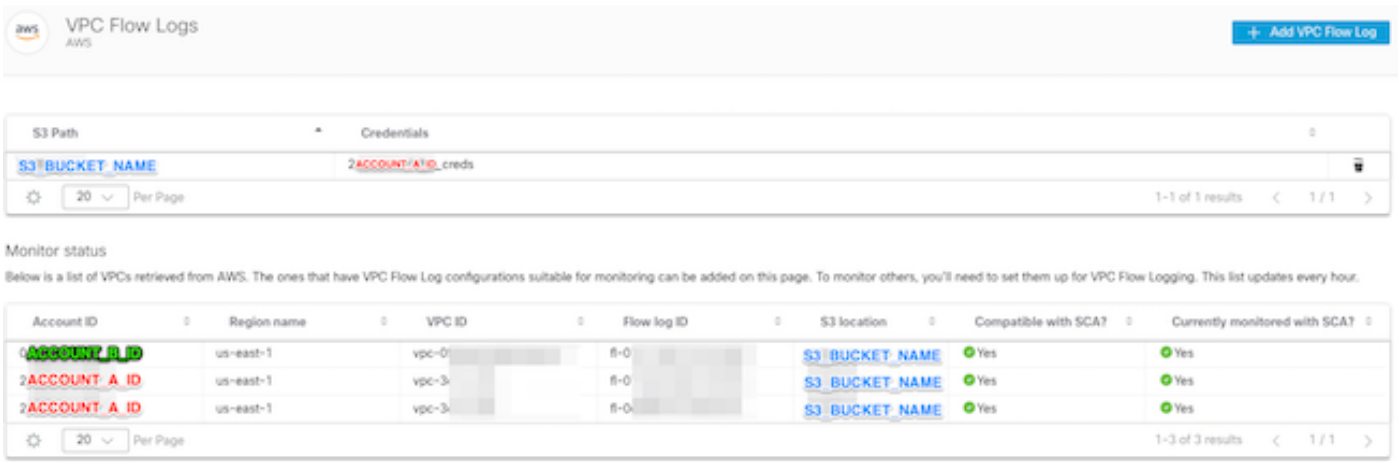
5. Click **Create**.

No further configuration steps are required.

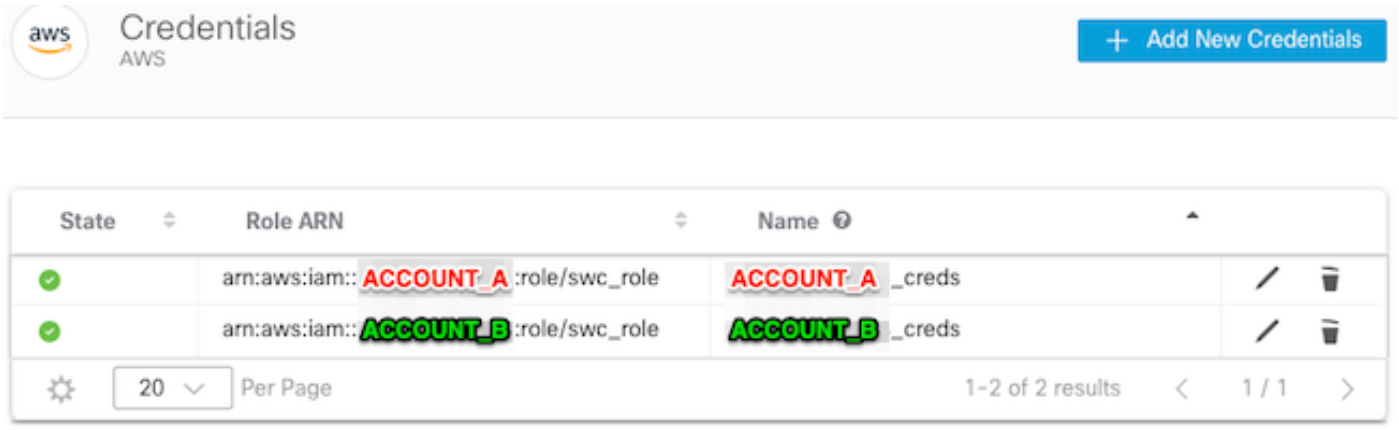
Verify

Use this section in order to confirm that your configuration works properly.

Your VPC Flow Logs page in your Secure Cloud Analytics web page looks like this image after about an hour. URL to VPC Flow Logs page: **https://portal-name.obsrvbl.com/v2/#/settings/integrations/aws/vpc_logs**



Your AWS Credentials page looks like this:



Troubleshoot

This section provides information you can use in order to troubleshoot your configuration.

If you do not see the same results on your VPC Flow Log page, then you need to enable AWS S3's Server Access Logging.

Examples of S3 Server Access Logging (SCA sensor GET-ing data from S3):

acfb735656a2b1bbd16c05582b753d11a9306f3a8dc20a4b9edc8c999aef9dd2 S3_BUCKET_NAME [10/Apr/2022:22:55:12 +0000]

10.0.129.197 arn:aws:sts::ACCOUNT_A_ID:assumed-role/swc_role/b401ed3c-58d1-472d-ab20-4801d0a7 CSQPM6SB0YZNWE03 REST.GET.BUCKET - "GET /?list-type=2&delimiter=%2F&prefix=AWSLogs%2FACCOUNT_B_ID%2Fvpcflowlogs%2F&encoding-type=url HTTP/1.1" 200 - 421 - 13 13 "-" "Boto3/1.17.85 Python/3.6.9 Linux/5.4.0-1064-aws Botocore/1.20.85" - ghD4o28lk0G1X3A33qCtXIg4qDRfo4eN3uebyV+tdCBQ6tOHk5XvLHGwbd7/EKXdzX+6PQxLHys= SigV4 ECDHE-RSA-AES128-GCM-SHA256 AuthHeader S3_BUCKET_NAME.s3.amazonaws.com TLSv1.2 - acfb735656a2b1bbd16c05582b753d11a9306f3a8dc20a4b9edc8c999aef9dd2 S3_BUCKET_NAME [10/Apr/2022:22:55:12 +0000] 10.0.129.197 arn:aws:sts::ACCOUNT_A_ID:assumed-role/swc_role/b401ed3c-58d1-472d-ab20-4801d0a7 CSQTXPDG4G6MY2CR REST.GET.BUCKET - "GET /?list-type=2&delimiter=%2F&prefix=AWSLogs%2F&encoding-type=url HTTP/1.1" 200 - 445 - 33 33 "-" "Boto3/1.17.85 Python/3.6.9 Linux/5.4.0-1064-aws Botocore/1.20.85" - geCd2CjQUqwxYjVs0JUt+gyEuKw92p3iJt52qx0A+bOaWhjaiNI77OxGqmvFlJZpMT5GePh6i9Y= SigV4 ECDHE-RSA-AES128-GCM-SHA256 AuthHeader S3_BUCKET_NAME.s3.amazonaws.com TLSv1.2 - acfb735656a2b1bbd16c05582b753d11a9306f3a8dc20a4b9edc8c999aef9dd2 S3_BUCKET_NAME [10/Apr/2022:22:55:12 +0000] 10.0.129.197 arn:aws:sts::ACCOUNT_A_ID:assumed-role/swc_role/b401ed3c-58d1-472d-ab20-4801d0a7 CSQVVKEPV0XD9987 REST.GET.BUCKET - "GET /?list-type=2&delimiter=%2F&prefix=AWSLogs%2FACCOUNT_A_ID%2Fvpcflowlogs%2F&encoding-type=url HTTP/1.1" 200 - 421 - 11 11 "-" "Boto3/1.17.85 Python/3.6.9 Linux/5.4.0-1064-aws Botocore/1.20.85" - hHR2+J5engOwp/Bi7Twn5ShsDXNYnH5rcB8YByFJP5OnZb64S1Y7/d+c7BSbBb861TpuJ0Jtpes= SigV4 ECDHE-RSA-AES128-GCM-SHA256 AuthHeader S3_BUCKET_NAME.s3.amazonaws.com TLSv1.2 -

Log field reference: https://docs.aws.amazon.com/AmazonS3/latest/userguide/LogFormat.html