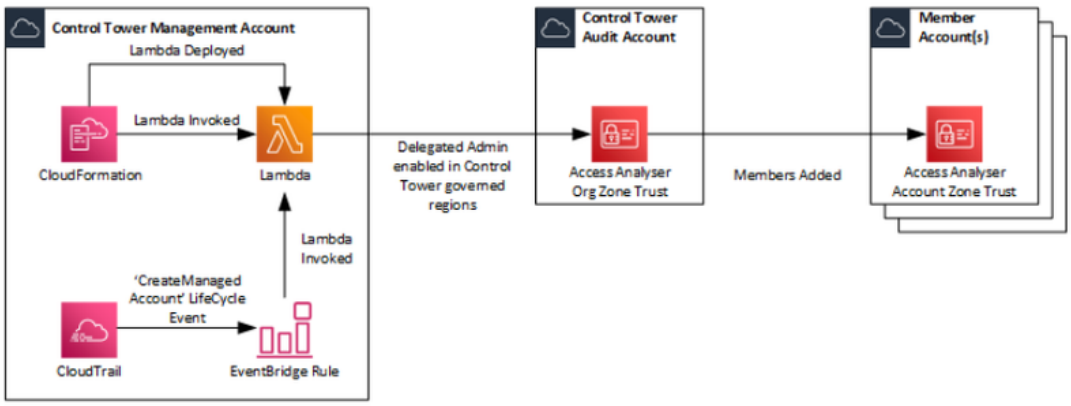
Synctree Flosports | AccessAnalyser

AWS Identity and Access Management Access Analyser helps identify potential resource-access risks by enabling you to identify any policies that grant access to an external principal. It does this by using logic-based reasoning to analyze resource-based policies in your AWS environment. An external principal can be another AWS account, a root user, an IAM user or role, a federated user, an AWS service, or an anonymous user. You can also use IAM Access Analyser to preview and validate public and cross-account access to your resources before deploying permissions changes. This guide describes the AWS Identity and Access Management Access Analyser operations you can call programmatically. For general information about the IAM Access Analyser.

The rationale behind this is for several reasons:

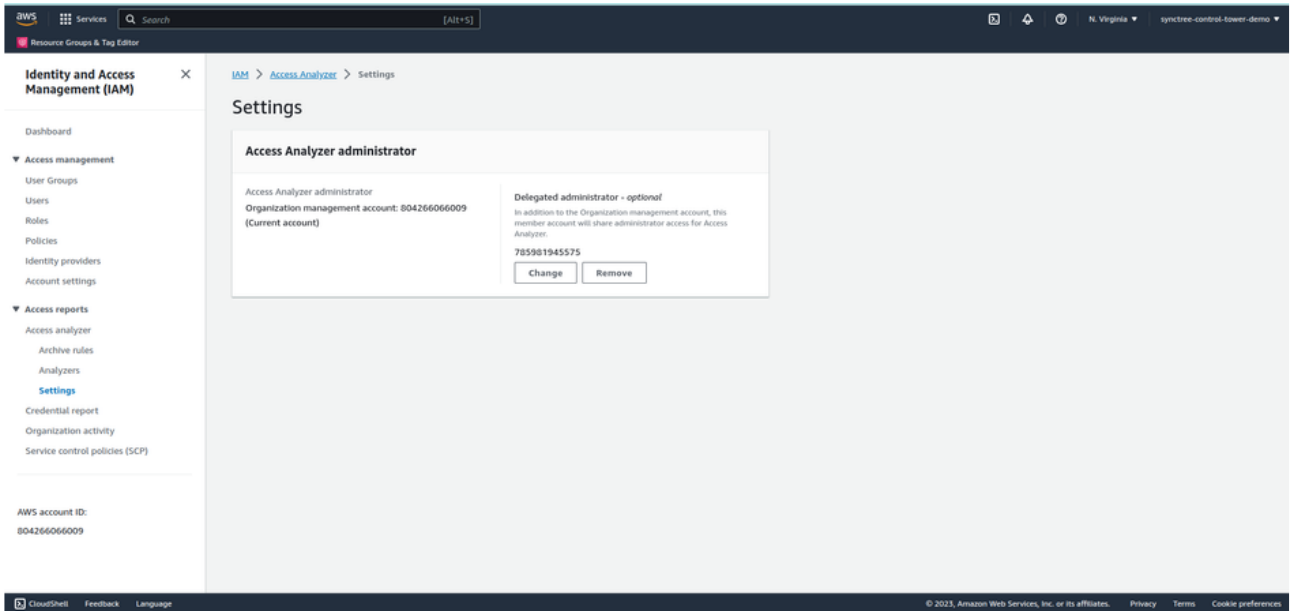
- The Organizational Zone of Trust provides visibility to a Single AWS Account (through Delegated Administration) and the ability to have visibility of everything going on within the Organisation e.g. all IAM Roles, and S3 Buckets. However this is just from what I've personally noticed that it doesn't seem to have visibility of SQS Policies, KMS Key Policies, Lambda Functions, Lambda Layer Version, or Secrets Manager Secrets.
- The Account Zone of Trust provides visibility into everything within the AWS Account including all the items that the Organization Zone of Trust seemed to be missing.
- S3 Access Analyser (within the S3 Service Console) is only available when there is an Account Zone of Trust configured.

For a better understanding see the below Architecture Diagram:



IAM Access Analyser delegation

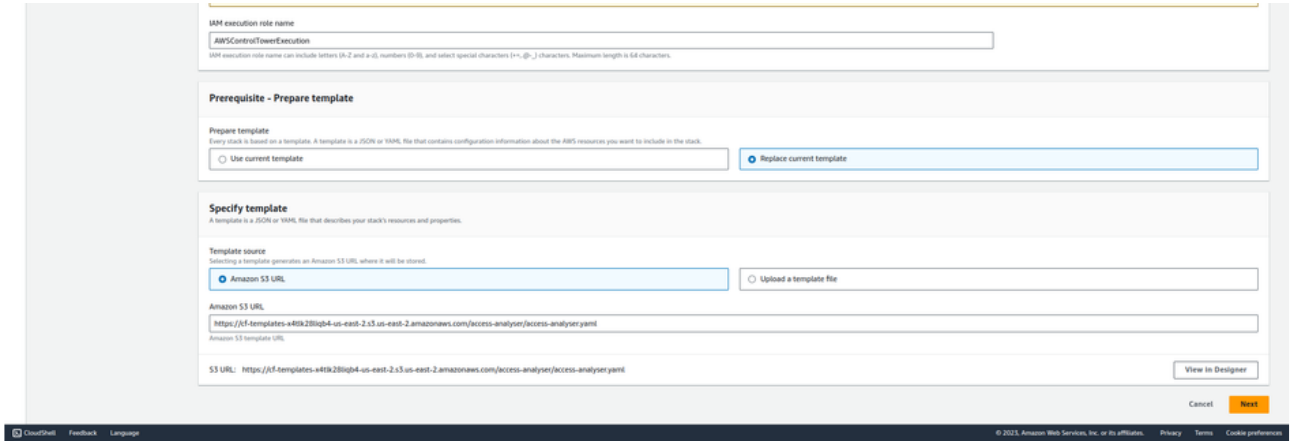
Before creating the stackset add the 12-digit account ID of your audit account for a delegated administrator for IAM Access Analyser. Only the master account can add, remove, or change a delegated administrator for IAM Access Analyser. From your AWS Control Tower master account, navigate to the **IAM console >** select Access Analyser > Settings. From here, you can add a delegated administrator. Add the 12-digit account ID of your audit account collected earlier, and save changes.



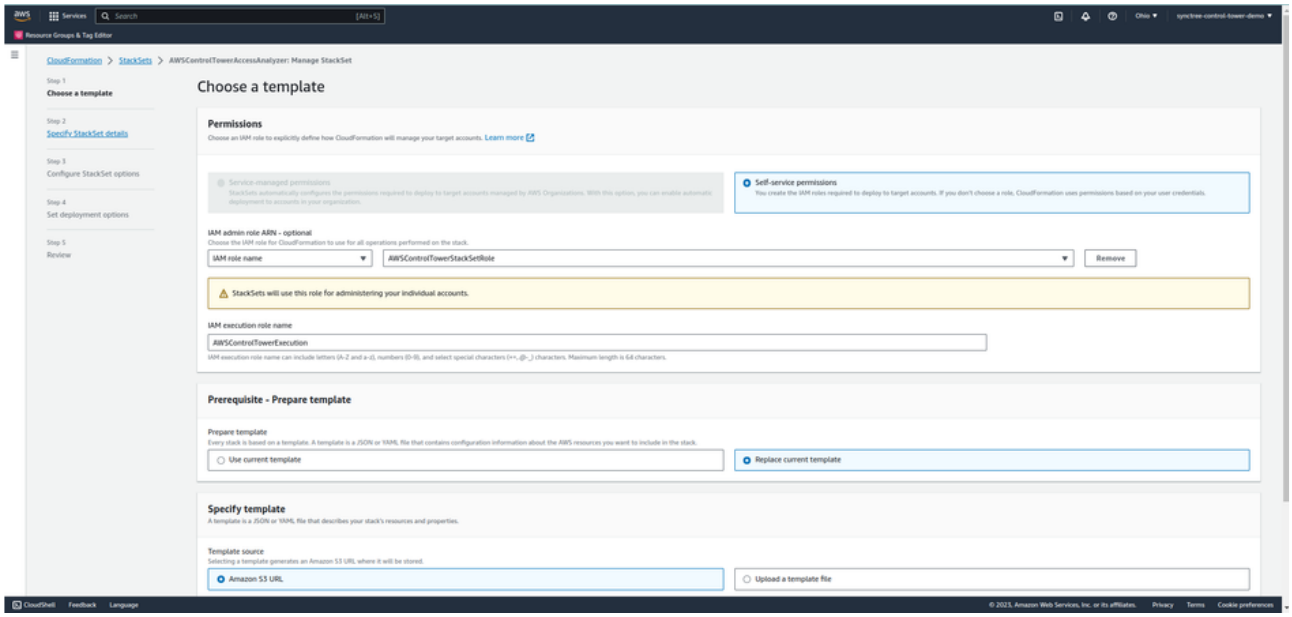
Start Creating CloudFormation StackSet :

Step1: Go to AWS Cloudformation>StackSets and Select Create StackSet

Step 2: Give the template file S3 URL and select Next.



Step 3: Give the IAM admin role (*AWSControlTowerStackSetRole)* and IAM execution role name (*AWSControlTowerExecution):*



Note:- If these roles are not already created, create them in the master account.

AWSControlTowerStackSetRole

Policy:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Action": [
                "sts:AssumeRole"
            ],
            "Resource": [
                "arn:aws:iam::*:role/AWSControlTowerExecution"
            ],
            "Effect": "Allow"
        }
    ]
}
```

Trust Relationship Policy:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "Service": "cloudformation.amazonaws.com"
            },
            "Action": "sts:AssumeRole"
        }
    ]
}
```

AWSControlTowerExecution

Policy:

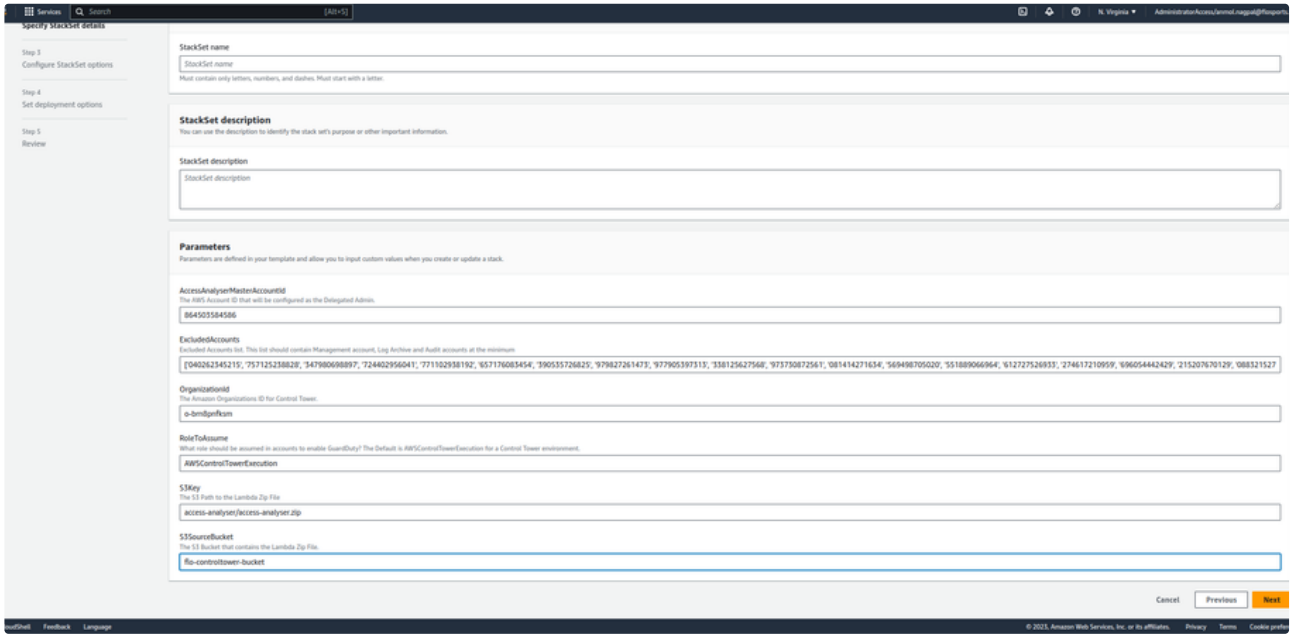
```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "*",
            "Resource": "*"
        }
    ]
}
```

Or give necessary permissions at our convenience.

Trust Relationship Policy:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "arn:aws:iam::<Master Account ID>:root"
            },
            "Action": "sts:AssumeRole",
            "Condition": {}
        }
    ]
}
```

Step 4: Add all the parameters required to create CloudFormation StackSet and select Next

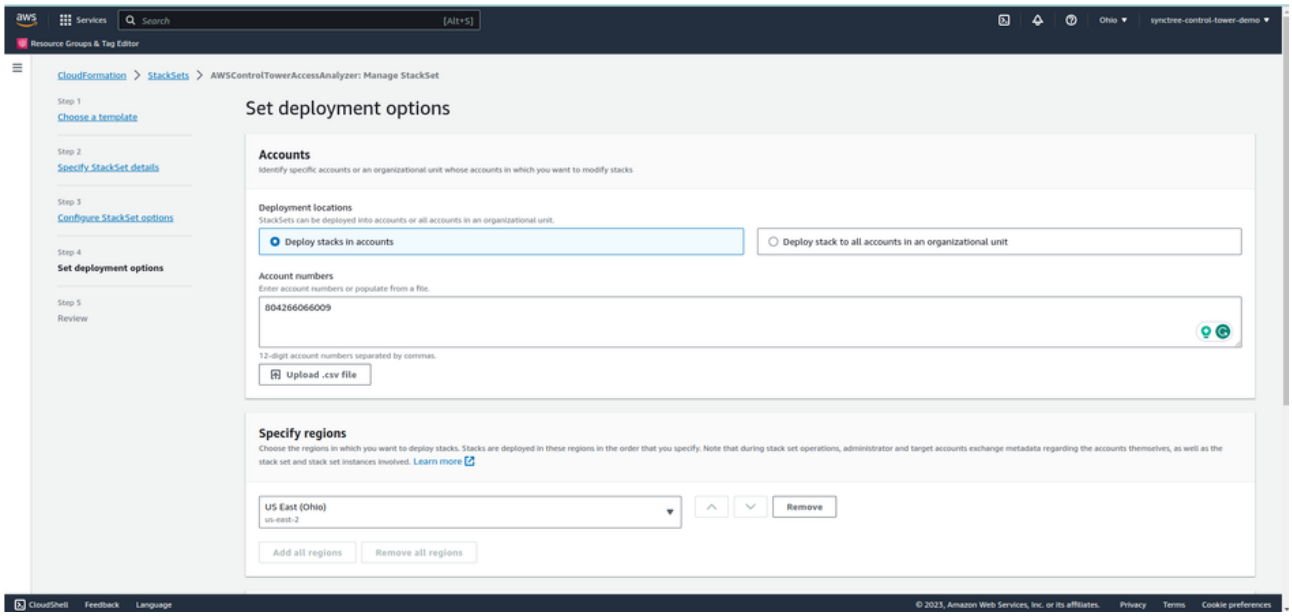


Note:- AccountAnalyserMasterAccountId should your be Audit Account ID.

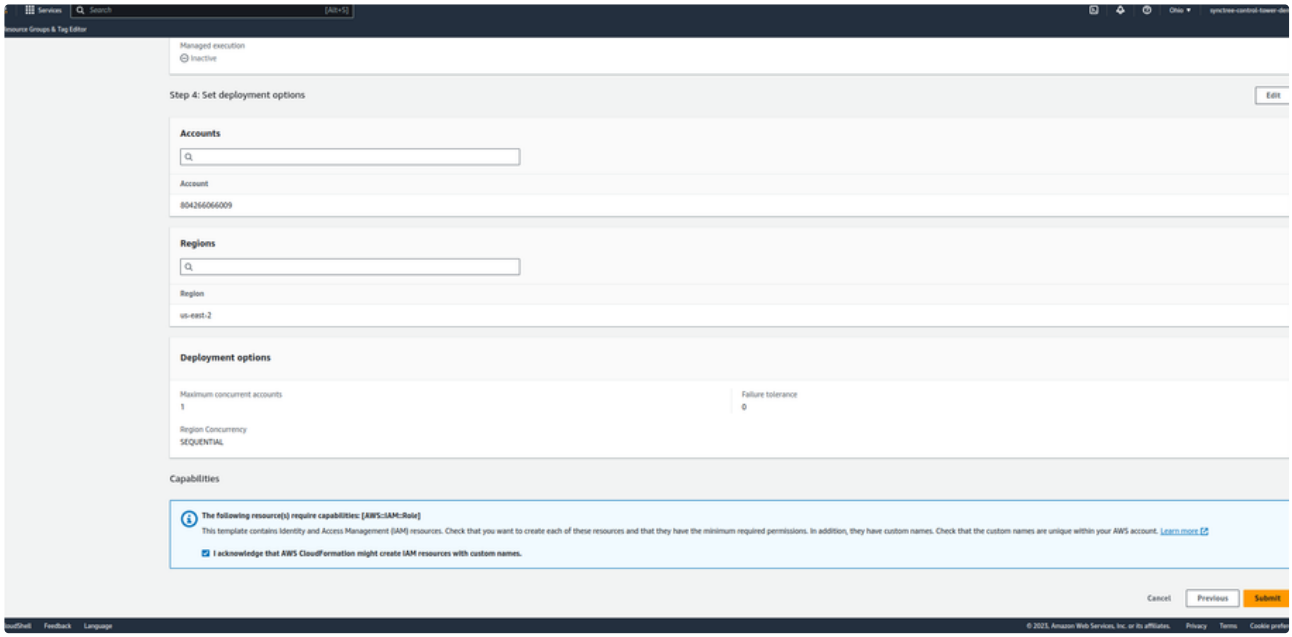
- Excluded Accounts- All accounts Id's that you want to exclude (e.g. ['111111111111', '222222222222',])
- OrganizationId- Organisation Id

- Role-to-Assume-AWS- `Control-Tower-Execution`
- S3-Source-Bucket- Add your Source Bucket name
- S3-Source-Key- Add path of source code of lambda file

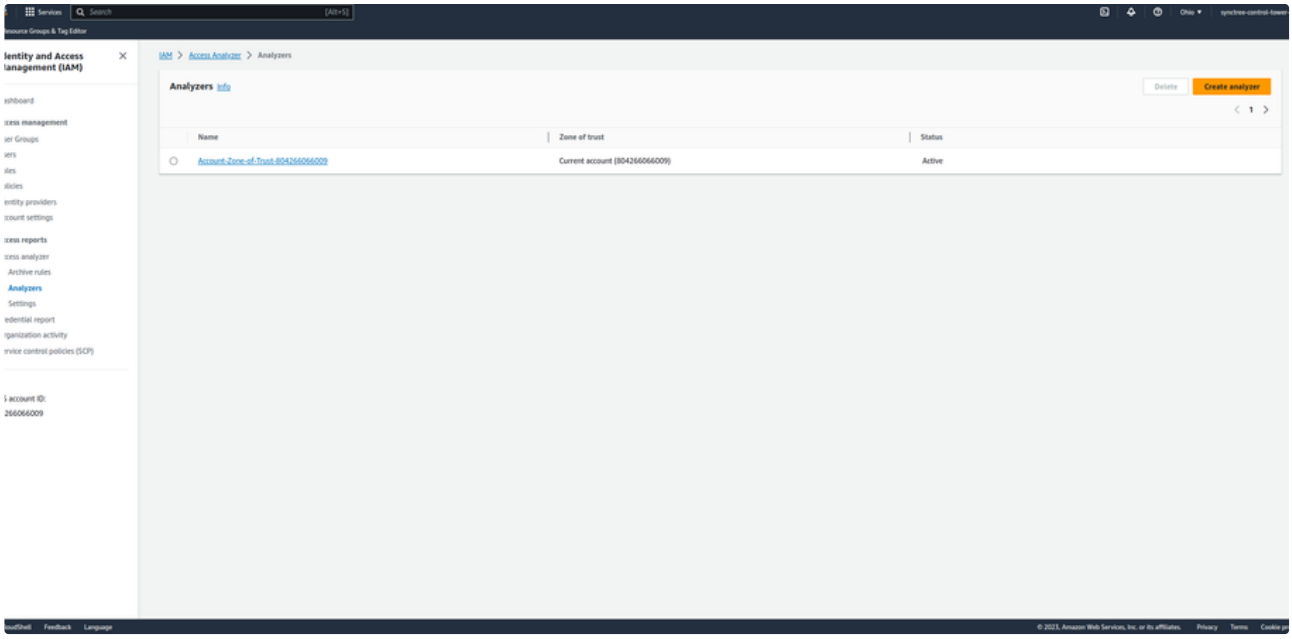
Step 5: Set the deployment option Deploy stack in account, Give master account Id, Specify region which is your control tower home region, and select Next



Step 6: Once reviewed parameters of stack, click on submit.



Step 7: Go to IAM > Access > Analysers to verify.



You can see that Access Analyser has been created and it is in Active status.

If you are getting the error "`CloudFormation did not receive a response from your Custom Resource. Please check your logs for requestId []. If you are using the Python cfn-response module,`"
Change the custom resource name in yaml template file:

```
  CustomResourceEnableAccessAnalyser:
    Type: Custom::EnableAccessAnalyser
```

Reference Link:

Enabling AWS IAM Access Analyzer on AWS Control Tower accounts | Amazon Web Services

https://github.com/AdamDivall/CfCT-AWS-Access-Analyser