Scanning AWS with OUTSCAN

Overview

As more and more people are moving their IT Infrastructure to cloud based services, there often seems to be an assumption that the cloud provider will secure the service. And to an extent, this is true. However, while they take extremely good measure to secure the service itself, it is almost never their responsibility to secure the hosts running within the service.

Amazon, for example, states that they will secure their cloud based service, AWS, however, it is the responsibility of the user to secure all services within that.

This means, that essentially users of AWS should treat (and secure) their IT assets as if they were running on their own infrastructure, including regular vulnerability scanning.

Because of the shared infrastructure model used by almost all cloud service providers, both server and network infrastructure may be utilized by others, and running any form of testing may have an impact on the availability and response.

If anyone wishes to conduct a vulnerability scan against anything hosted within AWS they are normally required to request permission from Amazon prior to the testing taking place. (http://aws.amazon.com/security/penetration-testing/) This can be a laborious task if there is a large number of hosts being scanned.

Outpost24 now offers pre-approved Vulnerability Scanning against AWS by utilizing the AWS API, enabling scanning of both instances and Elastic Load Balancers (ELB's)

This integrates with AWS and ensures that the scanning requirements and limitations set by Amazon have been met. These limitations are:

- Amazon Targets (Instance IDs) cannot be manually added. The only way to add an AWS instance is by using a Discovery scan, which uses the API to query the account and to gather instance IDs
- Only instance sizes of m1.medium and above can be scanned. If, during the Discovery phase, any instances are discovered that are smaller than an m1.medium, or ELBs that contain members smaller than an m1.medium, they will be reported back as 'Too small to scan'.
- Only instances with an Elastic IP (EIP), or Load Balancers will be scanned.
- Only safe tests can be run, and only using OUTSCAN.

While it may seem like there are a lot of limitations, these are exactly the same checks that Amazon would do if permission was explicitly required.

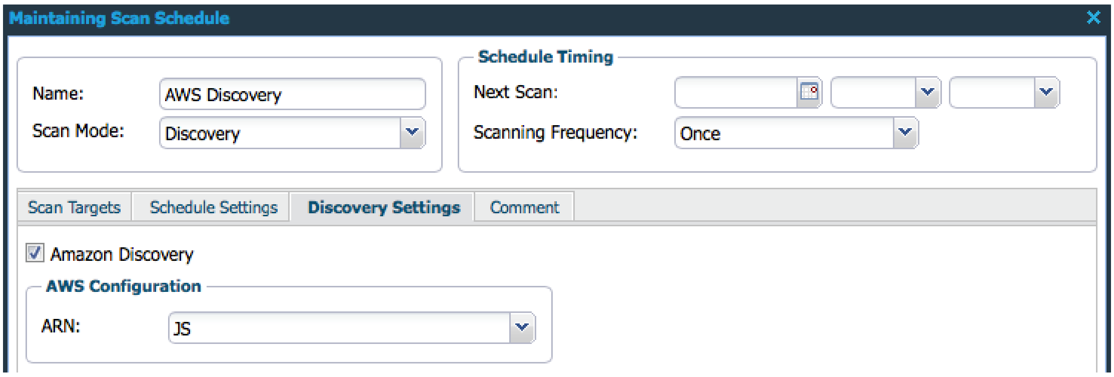
Configuration

There are 2 easy steps to the setup, firstly setup the AWS side of things, and then OUTSCAN. There are also a couple of pieces of information required to setup the AWS IAM access. Firstly, the Outpost24 account information, and then a permission policy to allow OUTSCAN to query the correct AWS API elements.

Full step by step details for configuring both AWS and OUTSCAN are given at the end, and it should be noted that this is only required to take place once. When the setup is complete, the only requirement is to setup a Discovery scan using the correct ARN.

It is worth noting that many companies use multiple AWS accounts and then make use of Amazons unified billing. In this instance, multiple ARNs can be added to OUTSCAN, with a name to describe them.

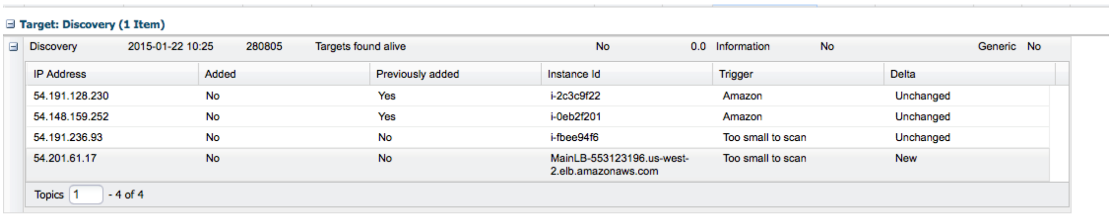
Once the ARNs have been added to OUTSCAN, as described in the step-by-step guide, the next step is to setup a Discovery scan on OUTSCAN, this is done similar to configuring a standard Discovery, except a new tick box will become available for 'Amazon Discovery'. Tick this box, and then select the name for the ARN to be used.



Name: AWS Discovery
Scan Mode: Discovery
Next Scan:
Scanning Frequency: Once
☑ Amazon Discovery
ARN: JS

During the discovery process, and number of things happen. Initially, OUTSCAN will use the API to query the account and list all of the associated instances and ELBs. Once OUTSCAN knows which instances are available, it queries the instance for their size. Anything less than an m1.medium will be marked as 'Too small to scan'. Similarly, OUTSCAN will also query the ELB's for the InstanceID's behind them. If any of those instances are less than an m1.medium, the load balancer cannot be scanned, and the ELB will be marked as 'Too small to scan'. It should be noted that even if an ELB has 6 large instances and 1 small instance, this is enough to stop it being eligible for scanning.

Once the Discovery has run, open reporting tools to see what was found during the discovery process.



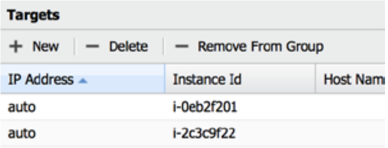
Target: Discovery (1 Item)

Discovery	2015-01-22 10:25	280805	Targets found alive	No	0.0	Information	No	Generic	No

IP Address	Added	Previously added	Instance Id	Trigger	Delta
54.191.128.230	No	Yes	i-2c3c9f22	Amazon	Unchanged
54.148.159.252	No	Yes	i-0eb2f201	Amazon	Unchanged
54.191.236.93	No	No	i-fbee94f6	Too small to scan	Unchanged
54.201.61.17	No	No	MainLB-553123196.us-west-2.elb.amazonaws.com	Too small to scan	New

Topics 1 - 4 of 4

Anything discovered and added to the targets during the scan (and if the ‘Add Targets to group’ option was selected during the setup of the Discovery) will be added by their InstanceID.



IP Address ▲	Instance Id	Host Nam
auto	i-0eb2f201	
auto	i-2c3c9f22	

Out of the 3 instances, and 1 ELB that have been discovered on my account, only 2 are eligible for scanning by Amazon.

From this point on, AWS assets will be treated exactly the same as a normal OUTSCAN target. However, when it comes to scan time, OUTSCAN will again use the API, query the account to ensure the InstanceIDs are still associated and of an acceptable size, and query the current IP address to ensure that the correct host is scanned.

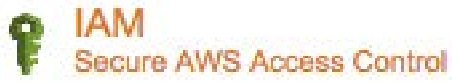
Summary

OUTSCAN now offers an effective way to run pre-authorized scans against AWS, effectively treating them as a standard target, but with all of the necessary checks required by Amazon prior to providing any form of scanning. Below are the step by step details to setup both the AWS side of things and setup the correct IAM access, and how to add the resulting ARN to OUTSCAN.

Step-by-step setup

a) Login to your AWS Management Console

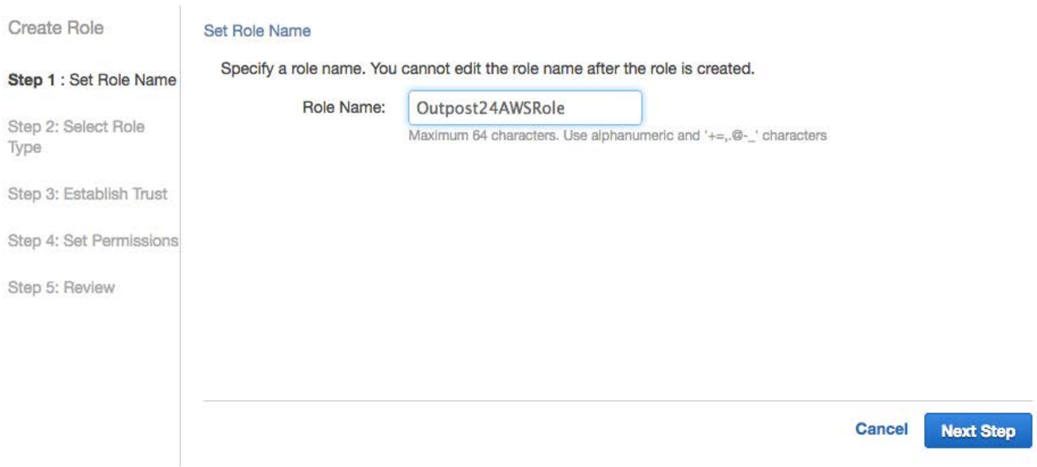
b) Click the IAM – Secure AWS Access Control Icon



c) Click ‘Roles’ in the left hand menu

d) Click ‘Create New Role’

e) Give the role a name such as ‘Outpost24AWSRole’



Create Role

Step 1 : Set Role Name

Step 2: Select Role Type

Step 3: Establish Trust

Step 4: Set Permissions

Step 5: Review

Set Role Name

Specify a role name. You cannot edit the role name after the role is created.

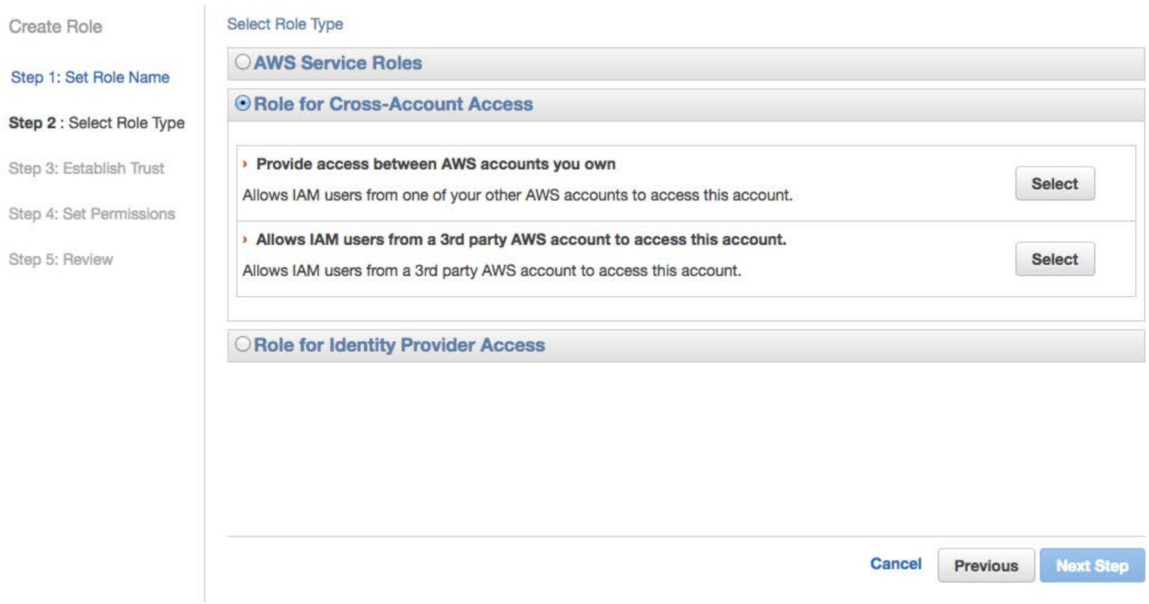
Role Name: Outpost24AWSRole

Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters

f) Click ‘Next Step’.

g) Select ‘Role for Cross-Account Access’

h) Select the ‘Allows IAM users from a 3rd party AWS account to access this account’ option



Create Role

Step 1: Set Role Name

Step 2 : Select Role Type

Step 3: Establish Trust

Step 4: Set Permissions

Step 5: Review

Select Role Type

○ AWS Service Roles

● Role for Cross-Account Access

› **Provide access between AWS accounts you own**

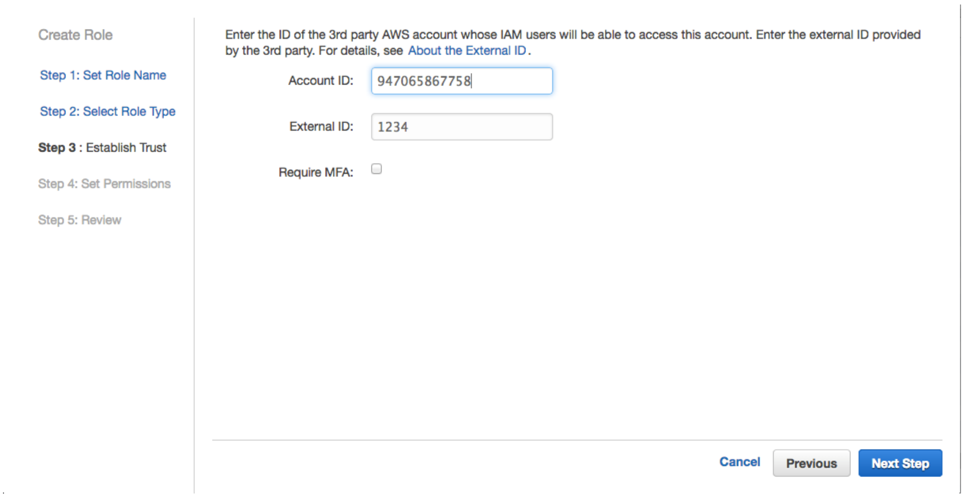
Allows IAM users from one of your other AWS accounts to access this account.

› **Allows IAM users from a 3rd party AWS account to access this account.**

Allows IAM users from a 3rd party AWS account to access this account.

○ Role for Identity Provider Access

i) Copy the Account ID (947065867758) and the unique external ID which has been allocated by OUTSCAN and paste it to the Role setup, then click ‘Next Step’



Create Role

Step 1: Set Role Name

Step 2: Select Role Type

Step 3 : Establish Trust

Step 4: Set Permissions

Step 5: Review

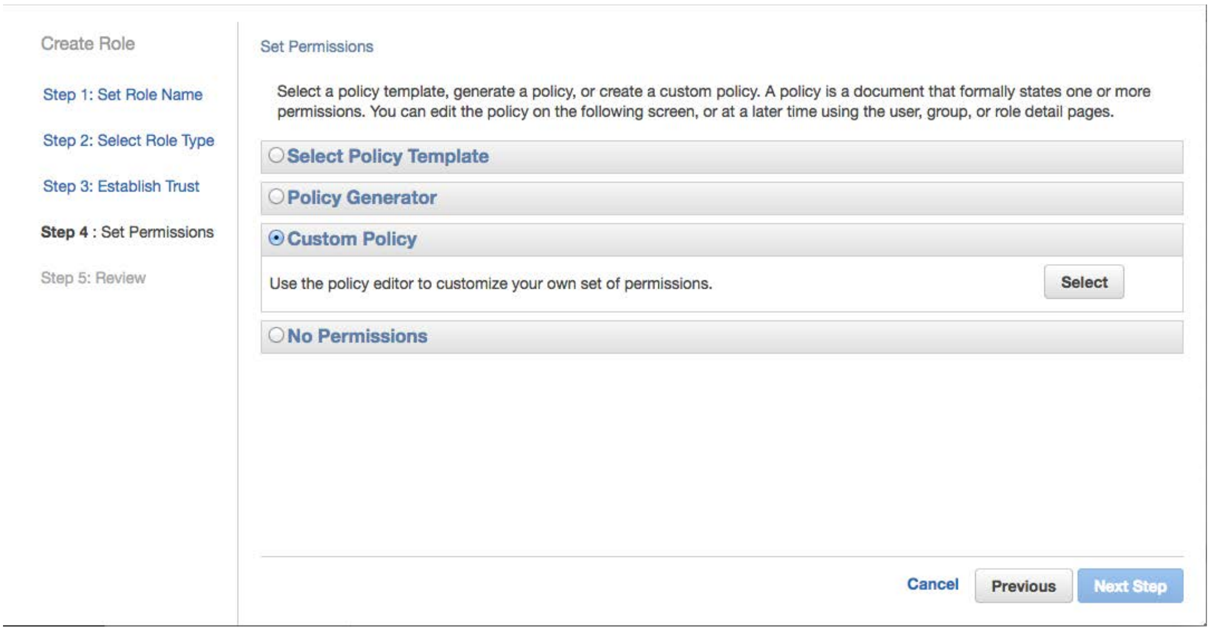
Enter the ID of the 3rd party AWS account whose IAM users will be able to access this account. Enter the external ID provided by the 3rd party. For details, see About the External ID.

Account ID: 947065867758

External ID: 1234

Require MFA: □

j) Select ‘Custom Policy’ and click select



Create Role

Step 1: Set Role Name

Step 2: Select Role Type

Step 3: Establish Trust

Step 4 : Set Permissions

Step 5: Review

Set Permissions

Select a policy template, generate a policy, or create a custom policy. A policy is a document that formally states one or more permissions. You can edit the policy on the following screen, or at a later time using the user, group, or role detail pages.

○ **Select Policy Template**

○ **Policy Generator**

● **Custom Policy**

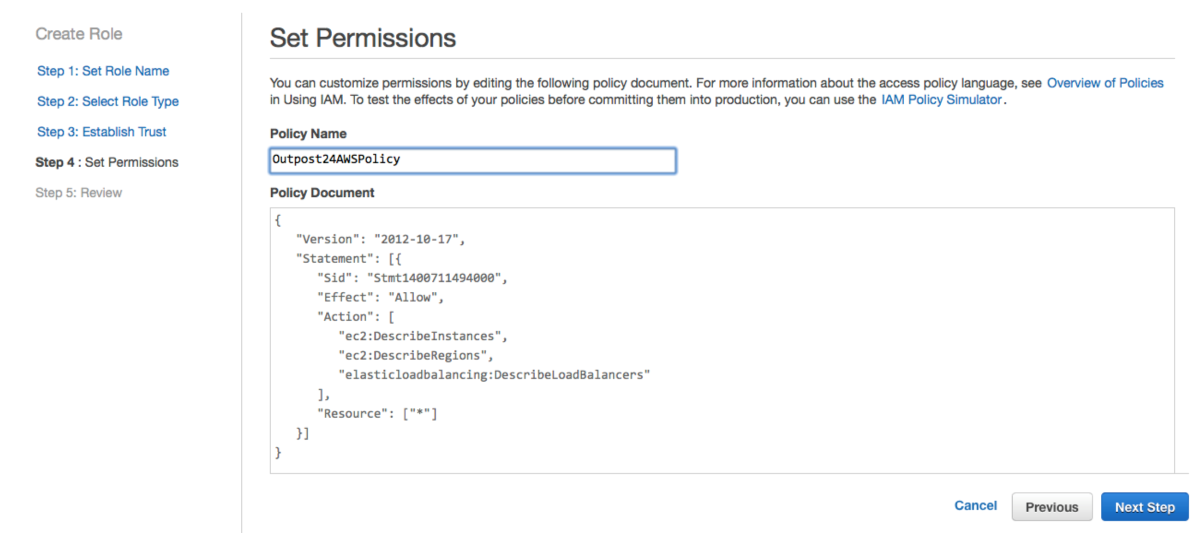
Use the policy editor to customize your own set of permissions.

○ **No Permissions**

k) Give the Policy a name, and then paste the following into the Policy Document:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "Stmt1400711494000",
    "Effect": "Allow",
    "Action": [
      "ec2:DescribeInstances",
      "ec2:DescribeRegions",
      "elasticloadbalancing:DescribeLoadBalancers"
    ],
    "Resource": ["*"]
  }]
}
```

Click ‘Next Step’



Set Permissions

You can customize permissions by editing the following policy document. For more information about the access policy language, see Overview of Policies in Using IAM. To test the effects of your policies before committing them into production, you can use the IAM Policy Simulator.

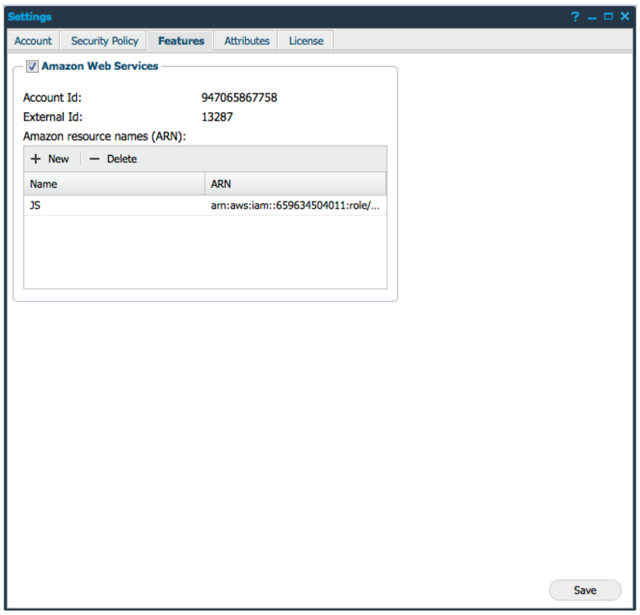
Policy Name: Outpost24AWSPolicy

Policy Document

```
{
   "Version": "2012-10-17",
   "Statement": [{
      "Sid": "Stmt1400711494000",
      "Effect": "Allow",
      "Action": [
         "ec2:DescribeInstances",
         "ec2:DescribeRegions",
         "elasticloadbalancing:DescribeLoadBalancers"
      ],
      "Resource": ["*"]
   }]
}
```

l) Review the settings and copy the ‘Role ARN’ and click ‘Create Role’

m) Login to your OUTSCAN account. Under Menu, Settings, Features, ensure Amazon AWS is selected and Click ‘New’. Under ‘Name’ enter a name to identify the ARN, and paste the ARN. Click ‘Save’



☑ Amazon Web Services

Account Id: 947065867758

External Id: 13287

Amazon resource names (ARN):

Name	ARN
JS	arn:aws:iam::659634504011:role/...