Outpost24

Vulnerability Management made easy

Scanning AWS With OUTSCAN

Version: 2.4

2019-02-26

Table of Contents

About This Document

The purpose of this document is to provide step by step details to setup scanning of Amazon Web Services (AWS) and the correct Identity and Access Management (IAM) access, and how to add the resulting Amazon Resource Name (ARN) to OUTSCAN™. This document has been elaborated under the assumption the reader has access to the OUTSCAN/HIAB account and portal interface.

For support information, visit https://www.outpost24.com/support.

1 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. 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 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 used 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/penetrationtesting/). 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 using 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 the same checks that Amazon would do if permission was explicitly required.

2 Configuration

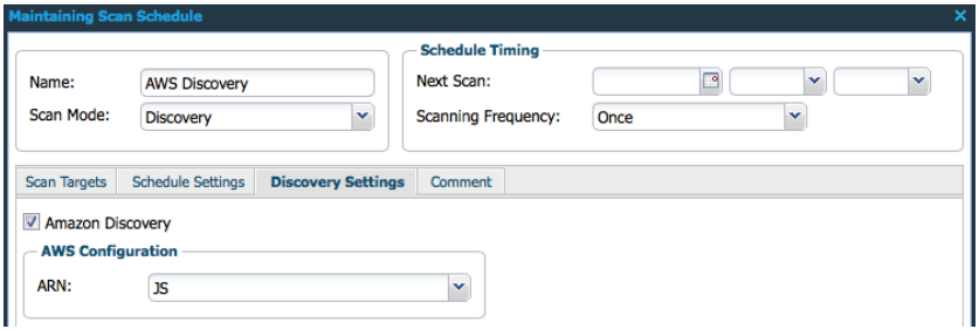
There are two easy steps to the setup. First setup the AWS side of things, and then OUTSCAN™. There are also a couple of pieces of information required to setup the AWS IAM access.

Firstly, the Outpost24® account information, and then a permission policy to allow OUTSCAN™ to query the correct AWS API elements.

A 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 section *3.1 Step-by-step setup*, the next step is to setup a Discovery scan on OUTSCAN™, this is done similar to configuring a standard Discovery, except a new check box becomes available for *Amazon Discovery*. Select this box, and then select the name for the ARN to be used.



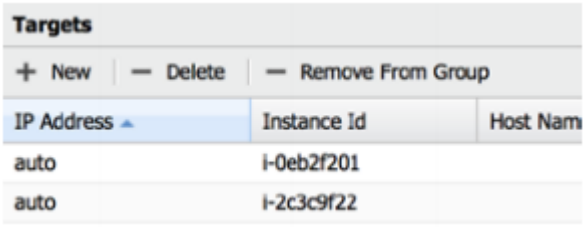
During the discovery process, several things happen. Initially, OUTSCAN™ use the API to query the account and list all the associated instances and ELBs. Once OUTSCAN™ knows which instances are available, it queries the instance for their size. Anything less than an **m1.medium** is marked as *Too small to scan*. Similarly, OUTSCAN™ 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 is marked as *Too small to scan*.

***Note**: Even if an ELB has six large instances and one small instance, this is enough to stop it being eligible for scanning.*

***Note**: Both ELB v1, v2 and CloudFront are supported.*

Once the Discovery process is completed, open reporting tools to see what was found during the process.

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) is added by their InstanceID.



Out of the three instances, and a discovered ELB, only two are eligible for scanning by Amazon.

From this point on, AWS assets are treated 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.

3 Summary

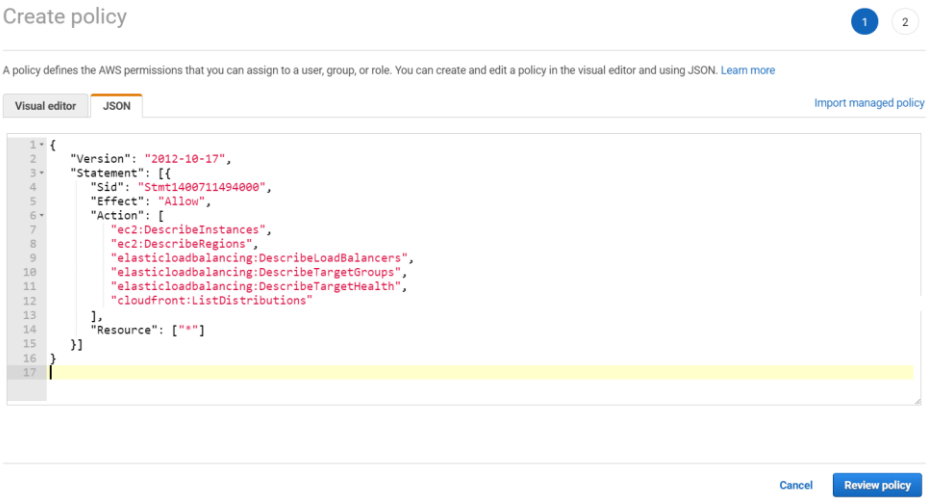
OUTSCAN™ now offers an effective way to run pre-authorized scans against AWS, effectively treating them as a standard target, but with all the necessary checks required by Amazon prior to providing any form of scanning. Below are the step by step details to setup both the AWS side of things and setup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

3.1 Step-by-step setup

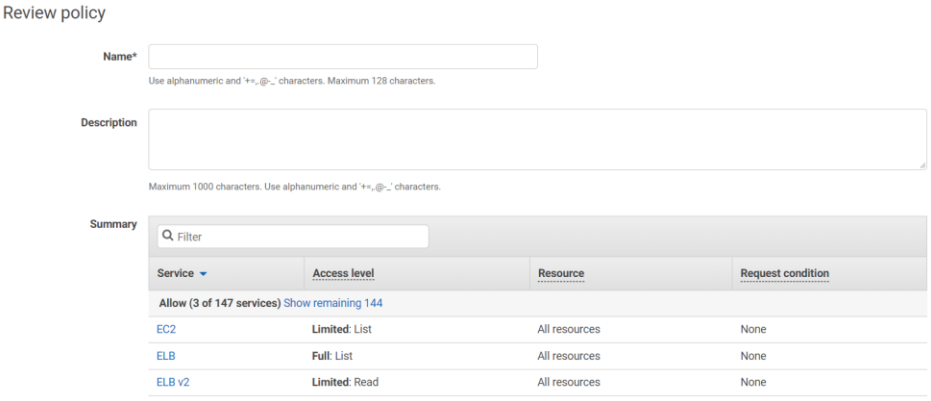
1. Login to your AWS Management Console.
2. Click **Services** at the top and then **IAM**.
3. Click **Roles** in the left-hand menu.
4. Click on **Create policy** then click on the JSON tab and copy the following statement in the window.

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

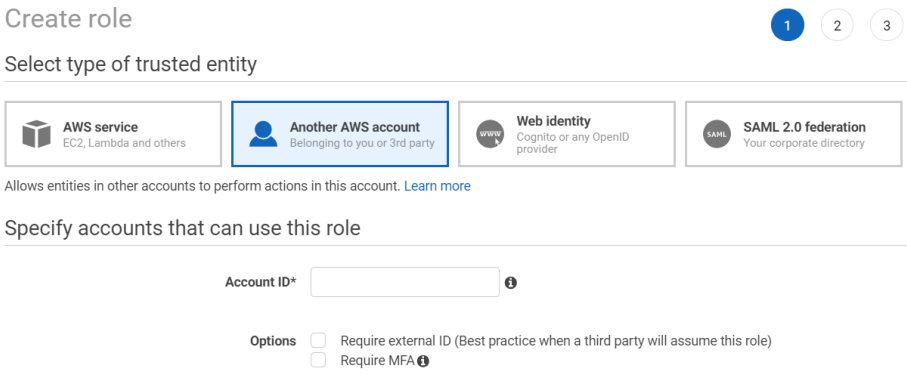
5. Click **Review policy.**



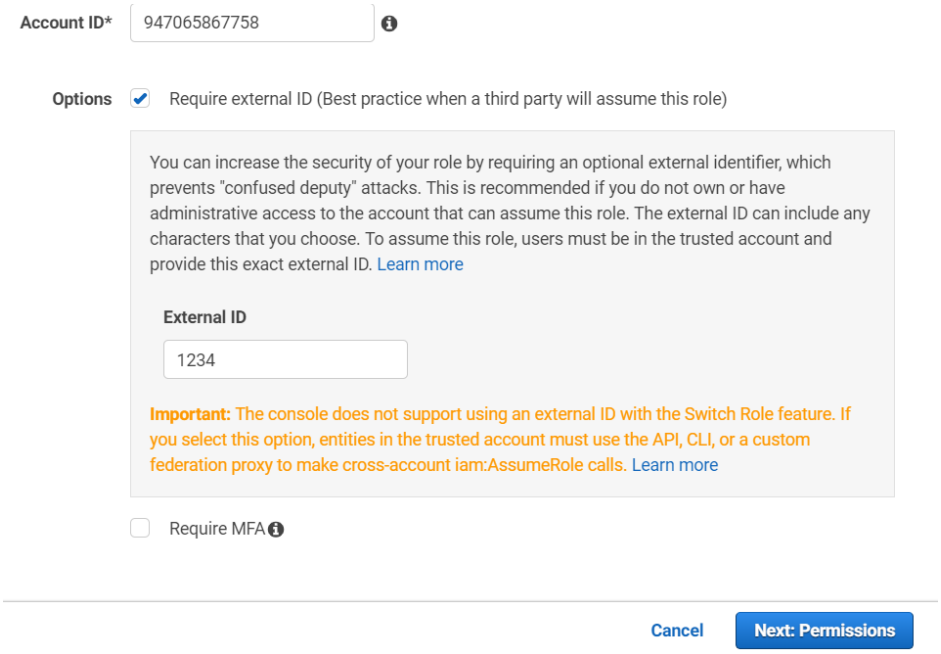
6. Give the role a name such as *Outpost24AWSRole.*



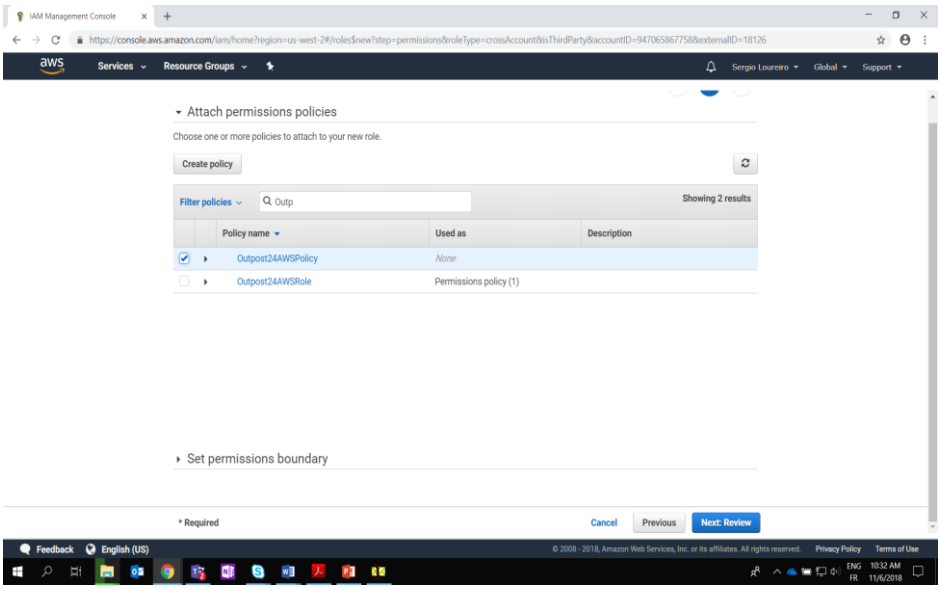
7. Click **Create Policy.**
8. Click **Create Role**.
9. Click on **Another AWS account**.



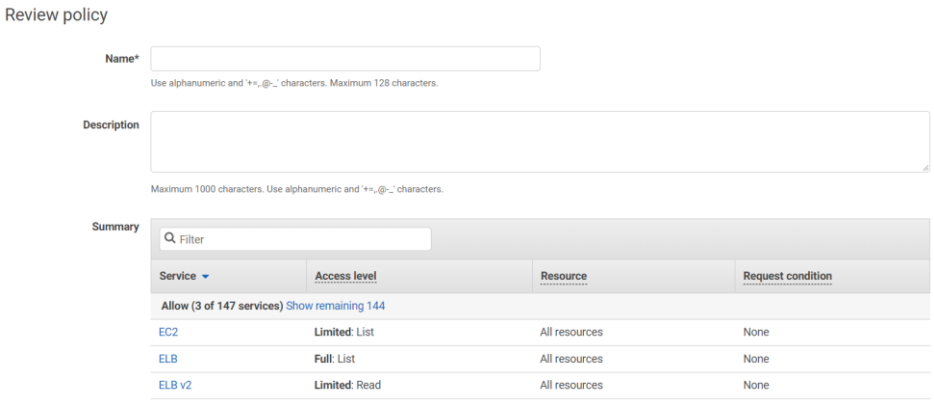
10. Copy the **Account ID** (947065867758) and the unique **External ID** which has been allocated by OUTSCAN and paste it.



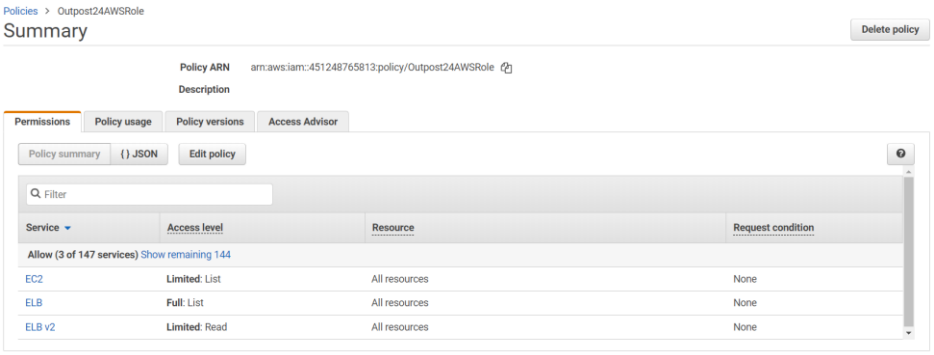
11. Click **Next: Permissions**.
12. Search for your policy, for example *Outpost24AWSPolicy* and tick the box.



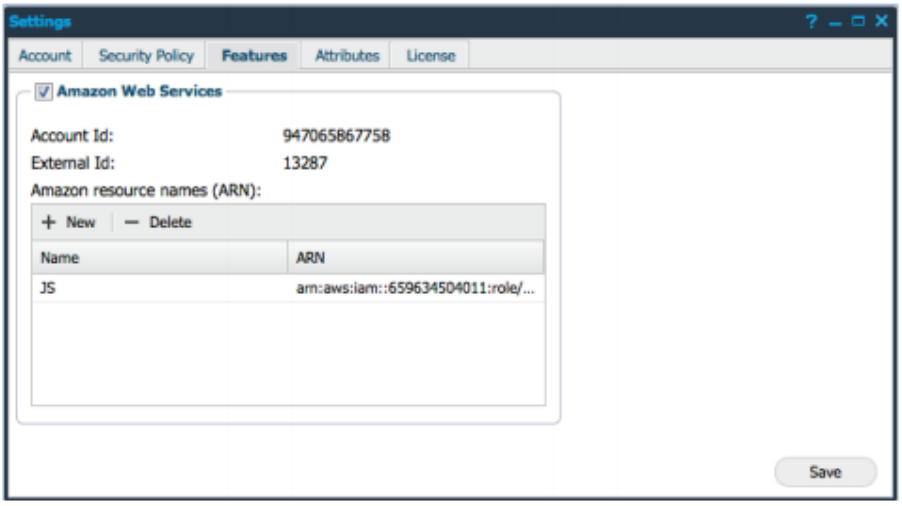
13. Click **Next:Review**.
14. Give the role a name such as *Outpost24AWSRole.*



15. Click **Create role**.
16. Click on Outpost24AWSRole and review the role.



17. Login to your OUTSCAN account.
18. Under **Menu** → **Settings** → **Features**, ensure Amazon AWS is selected and click **New**.



19. Under *Name* enter a name to identify the ARN and paste the ARN.
20. Click **Save**.