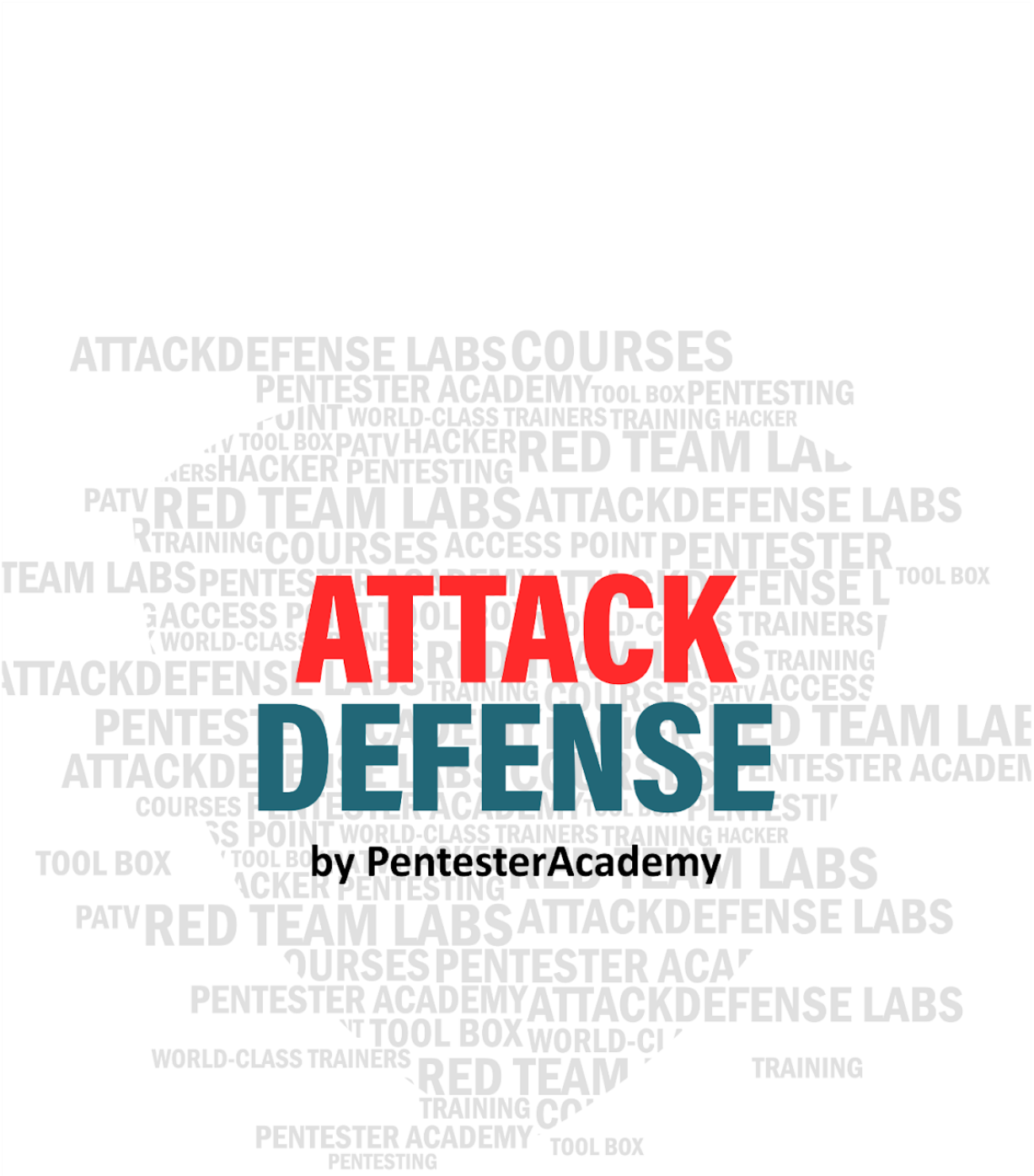

ATTACK
DEFENSE
by PentesterAcademy

Name	S3 Basics
URL	https://attackdefense.com/challengedetails?cid=1222
Type	Cloud Services : Amazon S3

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Objective: Interact with the exposed S3 Bucket endpoint and perform the following tasks:

1. Identify the S3 Endpoint.
2. List all the buckets owned by the current user.
3. List all the files stored on the S3 Bucket "hello-world".
4. Retrieve content stored in the text file "flag" on the S3 Bucket "hello-world".
5. Upload a file to S3 bucket "hello-world" with the content "Hello World".
6. Is Bucket listing allowed on the S3 Bucket "welcome"?
7. Modify the Bucket policy on S3 Bucket "welcome" to list the objects present in it.
8. Delete the file "welcome" from the S3 Bucket "hello-world"

1) Identifying S3 Endpoint

Solution:

Step 1: Find the IP address of the S3 Endpoint.

Command: ip addr

```
root@attackdefense:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
520: eth0@if521: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:01:01:09 brd ff:ff:ff:ff:ff:ff link-netnsid 0
```

```
    inet 10.1.1.9/24 brd 10.1.1.255 scope global eth0
       valid_lft forever preferred_lft forever
523: eth1@if524: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:c0:45:61:02 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.69.97.2/24 brd 192.69.97.255 scope global eth1
       valid_lft forever preferred_lft forever
root@attackdefense:~#
```

The attacker machine has an IP address 192.69.97.2. The machine running the S3 Endpoint will be located at the IP address 192.69.97.3

Step 2: Run default nmap scan and identify the open port on the target machine.

Command: nmap 192.69.97.3

```
root@attackdefense:~# nmap 192.69.97.3

Starting Nmap 7.60 ( https://nmap.org ) at 2019-09-11 10:36 IST
Nmap scan report for klyrl3hkahrhjgyceehlmwrwv.temp-network_a-69-97 (192.69.97.3)
Host is up (0.000015s latency).
Not shown: 999 closed ports
PORT     STATE SERVICE
9000/tcp open  cslistener
MAC Address: 02:42:C0:45:61:03 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.62 seconds
root@attackdefense:~#
```

Step 3: Send a curl request and check the received response.

Command: curl 192.69.97.3:9000

```
root@attackdefense:~# curl 192.69.97.3:9000
<?xml version="1.0" encoding="UTF-8"?>
<Error><Code>AccessDenied</Code><Message>Access Denied.</Message><Resource>/</Resource><RequestId>15C34A11CEAF254F</RequestId><HostId
>49db7dfe-bdf5-4b2b-bbba-7e3fb2d7e1a3</HostId></Error>root@attackdefense:~#
root@attackdefense:~#
```

XML response was received.

Step 4: Use xmllint to beautify the output returned by curl

Command: curl 192.69.97.3:9000 -s | xmllint --format -

```
root@attackdefense:~# curl 192.69.97.3:9000 -s  | xmllint --format -
<?xml version="1.0" encoding="UTF-8"?>
<Error>
  <Code>AccessDenied</Code>
  <Message>Access Denied.</Message>
  <Resource>/</Resource>
  <RequestId>15C34A632D074E15</RequestId>
  <HostId>49db7dfe-bdf5-4b2b-bbba-7e3fb2d7e1a3</HostId>
</Error>
root@attackdefense:~#
```

From the Error output received, it can be concluded that an S3 Endpoint is possibly running on port 9000.

2) List all the buckets owned by the current user.

Solution:

Step 1: Configure AWS Client

Command: aws configure
Enter Access Key ID: IVUSXS59YLBY2BTK19IG
Enter Secret Access Key: ou3LmvuYhwBuSICcnk7dJPh+FiljfqLgE+ZhrLEB

```
root@attackdefense:~# aws configure
AWS Access Key ID [None]: IVUSXS59YLBY2BTK19IG
AWS Secret Access Key [None]: ou3LmvuYhwBuSICcnk7dJPh+FiljfqLgE+ZhrLEB
Default region name [None]:
Default output format [None]:
root@attackdefense:~#
```

Step 2: Use aws client to list the buckets owned by the current user.

Command: aws --endpoint http://192.69.97.3:9000 s3api list-buckets

```
root@attackdefense:~# aws --endpoint http://192.69.97.3:9000 s3api list-buckets
{
    "Buckets": [
        {
            "Name": "hello-world",
            "CreationDate": "2019-09-10T19:57:41.000Z"
        },
        {
            "Name": "private",
            "CreationDate": "2019-09-10T15:12:53.000Z"
        },
        {
            "Name": "public",
            "CreationDate": "2019-09-10T19:50:38.000Z"
        },
        {
            "Name": "welcome",
            "CreationDate": "2019-09-10T19:52:52.000Z"
        }
    ],
    "Owner": {
        "DisplayName": "",
        "ID": "02d6176db174dc93cb1b899f7c6078f08654445fe8cf1b6ce98d8855f66bdbf4"
    }
}
root@attackdefense:~#
```

The current user owns 4 buckets: hello-world, private, public and welcome.

3) List all the files stored on the S3 Bucket "hello-world".

Solution:

Command: aws --endpoint http://192.69.97.3:9000 s3 ls s3://hello-world

```
root@attackdefense:~# aws --endpoint http://192.69.97.3:9000 s3 ls s3://hello-world
2019-09-11 00:29:55         33 flag
2019-09-11 01:27:41     145374 wallpaper.jpg
2019-09-11 00:30:59         35 welcome
root@attackdefense:~#
```

There are 3 files in the hello-world bucket: flag, wallpaper.jpg and welcome

4) Retrieve content stored in the text file "flag" on the S3 Bucket "hello-world".

Solution:

Step 1: Retrieve the flag file from the s3 bucket.

Commands:
aws --endpoint http://192.69.97.3:9000 s3 cp s3://hello-world/flag ./
cat flag

```
root@attackdefense:~# aws --endpoint http://192.69.97.3:9000 s3 cp s3://hello-world/flag ./
download: s3://hello-world/flag to ./flag
root@attackdefense:~#
root@attackdefense:~#
root@attackdefense:~# cat flag
f840fbb90dbf8490bf9e9b597899c969
root@attackdefense:~#
```

Flag: f840fbb90dbf8490bf9e9b597899c969

5) Upload a file to S3 bucket "hello-world" with the content "Hello World".

Solution:

Step 1: Create a file with 'hello world' string.

Command: echo 'Hello World' > hello

```
root@attackdefense:~#
root@attackdefense:~#
root@attackdefense:~# echo  'Hello World' > hello
root@attackdefense:~#
root@attackdefense:~#
```

Step 2: Upload the file to S3 bucket.

Command: aws --endpoint http://192.69.97.3:9000 s3 cp hello s3://hello-world/

```
root@attackdefense:~# aws --endpoint http://192.69.97.3:9000 s3 cp hello s3://hello-world/
upload: ./hello to s3://hello-world/hello
root@attackdefense:~#
```

The file was uploaded successfully.

6) Is Object listing allowed for anonymous users on the S3 Bucket "welcome"?

Solution:

Step 1: Use curl command to check whether bucket 'welcome' is accessible without authentication.

Command: curl http://192.69.97.3:9000/welcome/ -s | xmllint --format -

```
root@attackdefense:~# curl http://192.69.97.3:9000/welcome/ -s  | xmllint --format -
<?xml version="1.0" encoding="UTF-8"?>
<Error>
  <Code>AccessDenied</Code>
  <Message>Access Denied.</Message>
  <BucketName>welcome</BucketName>
  <Resource>/welcome/</Resource>
  <RequestId>15C34D05A48809A5</RequestId>
  <HostId>49db7dfe-bdf5-4b2b-bbba-7e3fb2d7e1a3</HostId>
</Error>
root@attackdefense:~#
```

The objects are not listed.

Step 2: Checking bucket policy of welcome bucket.

Command: aws --endpoint http://192.69.97.3:9000 s3api get-bucket-policy --bucket welcome

```
root@attackdefense:~# aws --endpoint http://192.69.97.3:9000 s3api get-bucket-policy --bucket welcome
{
    "Policy": "{\"Version\":\"2012-10-17\",\"Statement\":[{\"Sid\":\"AddPerm\",\"Effect\":\"Allow\",\"Principal\":{\"AWS\":[\"*\"]},\
"Action\":[\"s3:GetObject\"],\"Resource\":[\"arn:aws:s3:::welcome/*\"]}]}"
}
root@attackdefense:~#
```

ListBucket Action is not specified hence, the objects will not be listed for anonymous users.

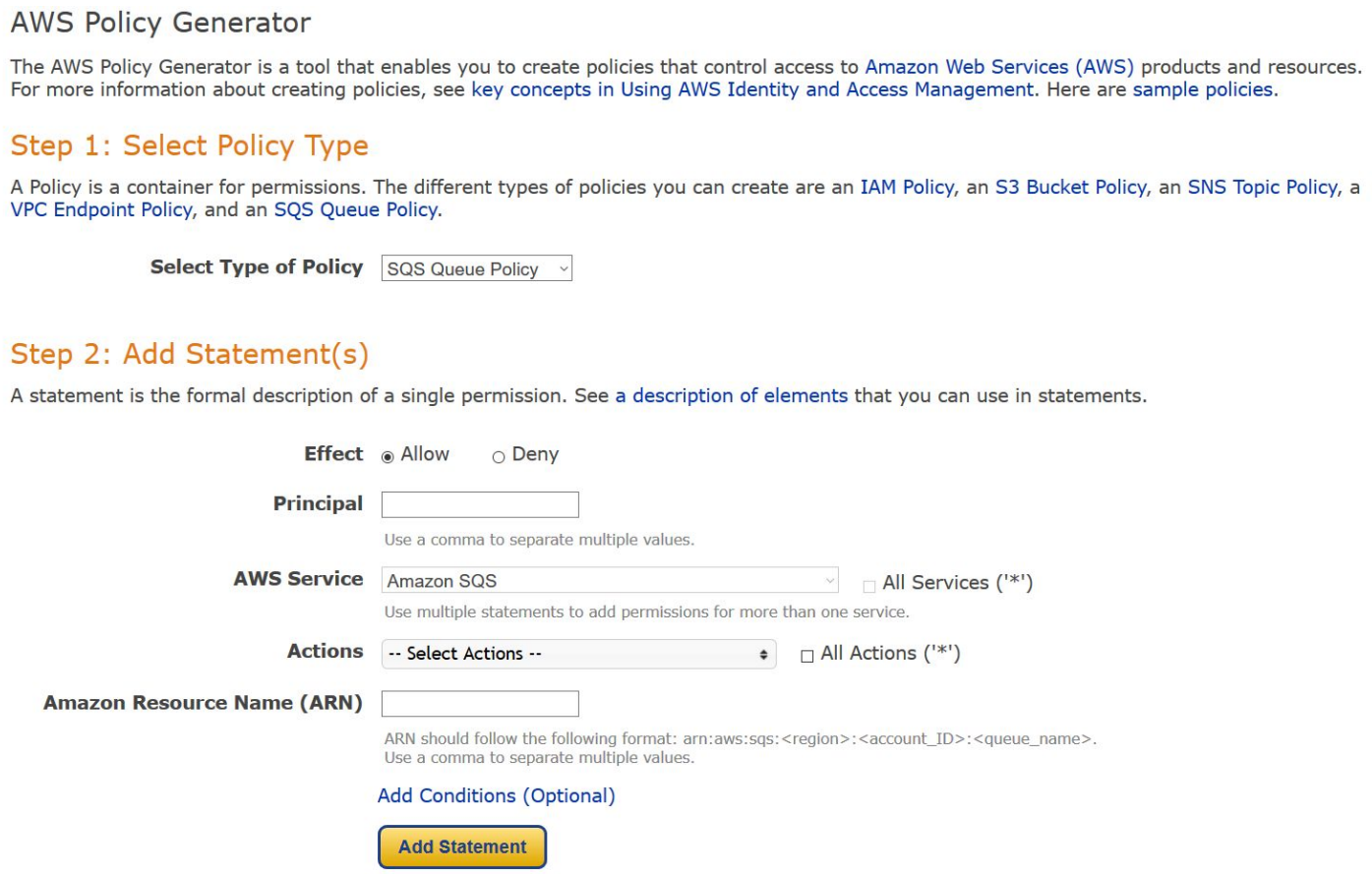
7) Modify the Bucket policy on S3 Bucket "welcome" to list the objects present in it.

Solution:

Step 1: Create a bucket policy.

The bucket policy can be created with AWS Policy Generator

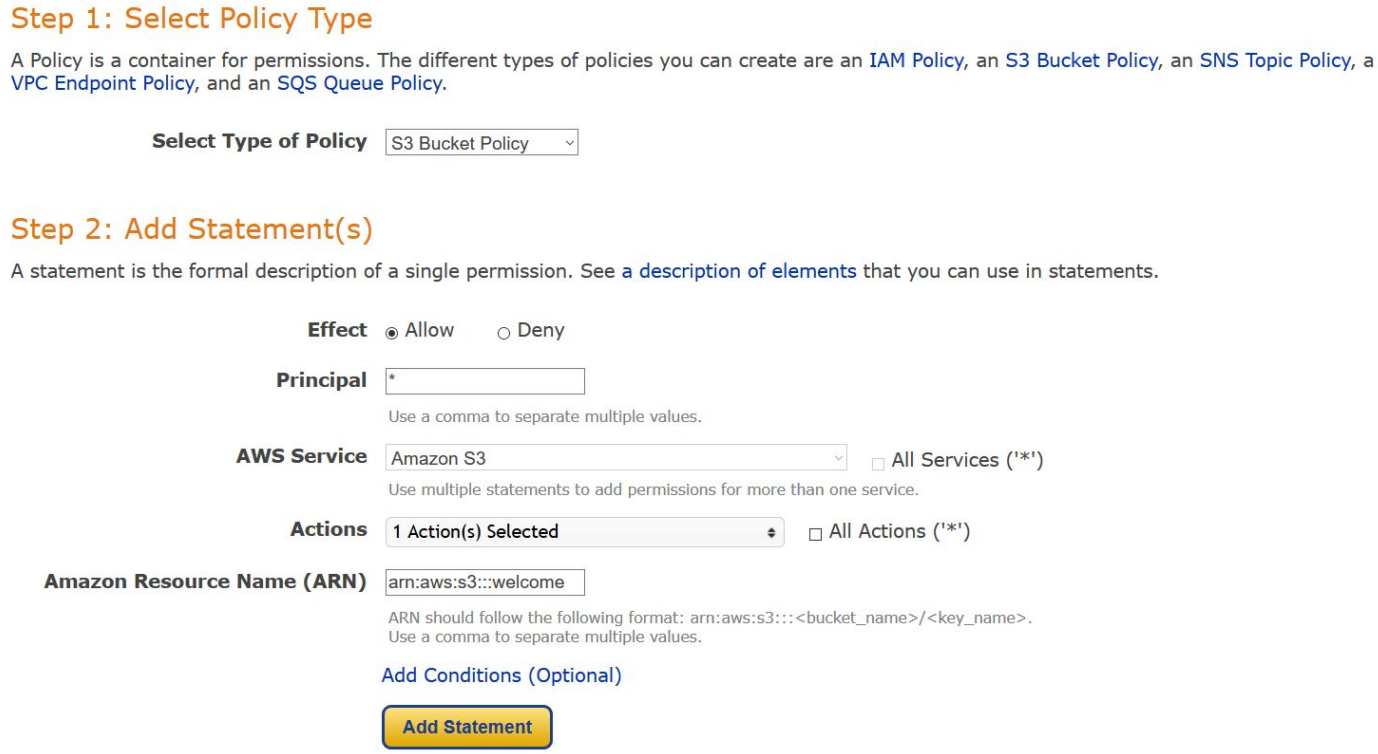
AWS Policy Generator: https://awspolicygen.s3.amazonaws.com/policygen.html



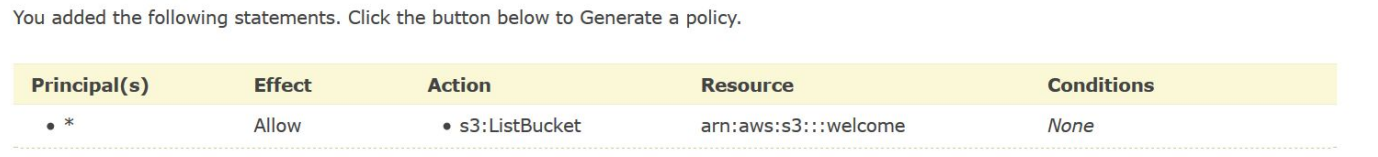
In AWS Policy Generator select the “S3 Bucket Policy” in “Select Policy Type” section and following options in “Add Statements” section

Effect: “Allow”

Principal: *
AWS Service: Amazon S3
Actions: ListBucket
Amazon Resource Name: arn:aws:s3:::welcome

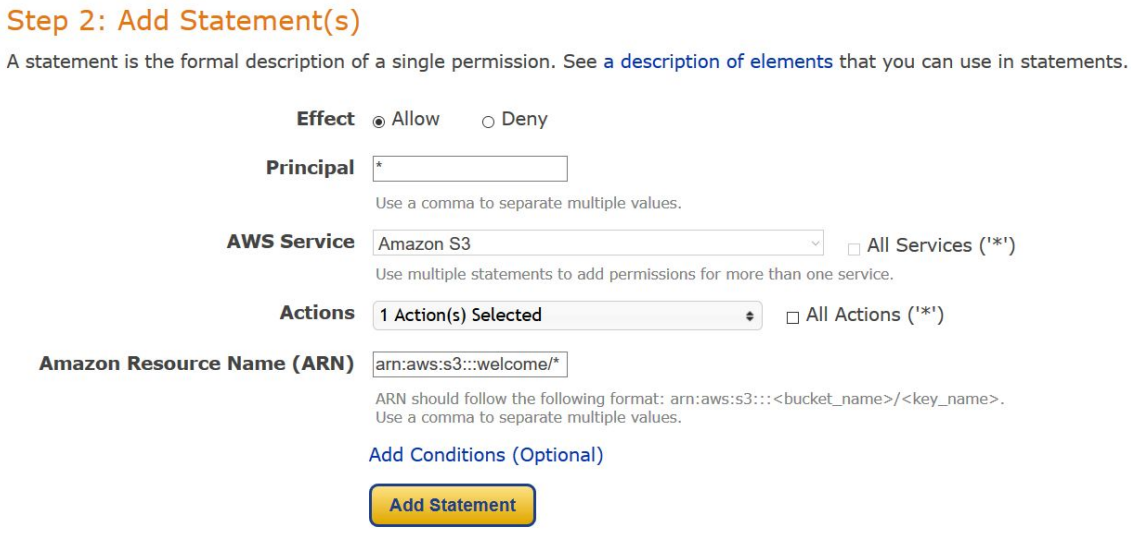


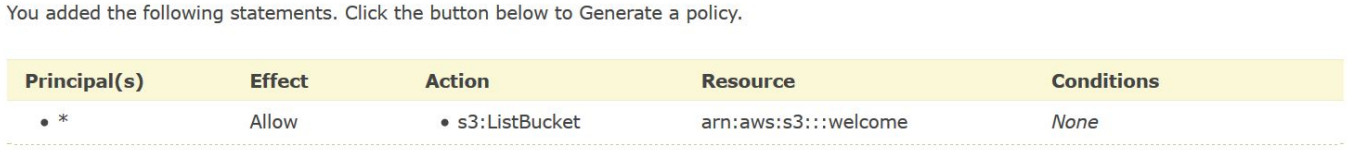
Click on “Add Statement” Button:



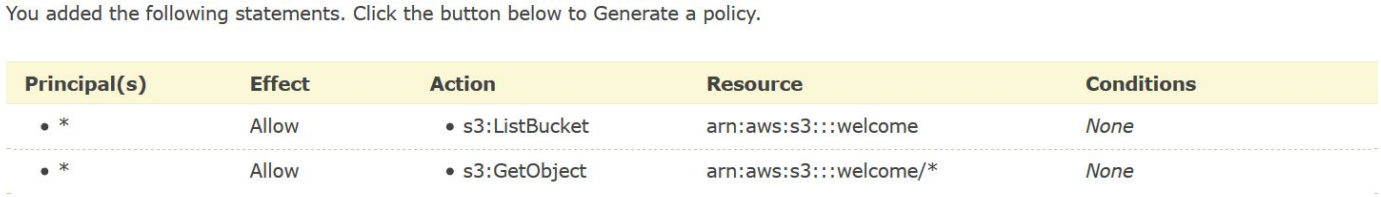
Add one more statement:

Effect: “Allow”
Principal: *
AWS Service: Amazon S3
Actions: GetObject
Amazon Resource Name: arn:aws:s3:::welcome/*





Click on “Add Statement” Button:



The statements will be listed on the webpage. Click on Generate Policy button to get the policy in JSON format

Step 3: Generate Policy

A *policy* is a document (written in the Access Policy Language) that acts as a container for one or more statements.

Generate Policy Start Over

Policy JSON Document:

Policy JSON Document

Click below to edit. To save the policy, copy the text below to a text editor.
Changes made below will **not be reflected in the policy generator tool.**

```
{
  "Id": "Policy1568185116930",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1568184932403",
      "Action": [
        "s3:ListBucket"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::welcome",
      "Principal": "*"
    },
    {
      "Sid": "Stmt1568185007451",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::welcome/*",
      "Principal": "*"
```

Close

JSON Policy:

```
{
  "Id": "Policy1568185116930",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1568184932403",
      "Action": [
        "s3:ListBucket"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::welcome",
      "Principal": "*"
    },
    {
      "Sid": "Stmt1568185007451",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::welcome/*",
      "Principal": "*"
    }
  ]
}
```

Save the json as policy.json

Command: cat policy.json

```
root@attackdefense:~# cat policy.json
{
  "Id": "Policy1568185116930",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1568184932403",
      "Action": [
        "s3:ListBucket"
```

```
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::welcome",
      "Principal": "*"
    },
    {
      "Sid": "Stmt1568185007451",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::welcome/*",
      "Principal": "*"
    }
  ]
}
root@attackdefense:~#
```

Step 2: Update the Bucket Policy.

Command: aws --endpoint http://192.69.97.3:9000 s3api put-bucket-policy --policy file:///root/policy.json --bucket welcome

```
root@attackdefense:~#
root@attackdefense:~# aws --endpoint http://192.69.97.3:9000 s3api put-bucket-policy --policy file:///root/policy.json --bucket welco
me
root@attackdefense:~#
```

Since no error was thrown, the bucket policy was updated successfully.

Step 3: Use curl and check whether objects can be listed on the object.

Command: curl http://192.69.97.3:9000/welcome/ -s | xmllint --format -

```
root@attackdefense:~# curl http://192.69.97.3:9000/welcome/ -s  | xmllint --format -
<?xml version="1.0" encoding="UTF-8"?>
<ListBucketResult xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Name>welcome</Name>
  <Prefix/>
  <Marker/>
  <MaxKeys>1000</MaxKeys>
  <Delimiter/>
  <IsTruncated>false</IsTruncated>
```

```
  <Contents>
    <Key>Documents/doc1.txt</Key>
    <LastModified>2019-09-10T19:21:10.000Z</LastModified>
    <ETag>"498e71ee1e6be39d454f7c4e45cad37b"</ETag>
    <Size>24</Size>
    <Owner>
      <ID>02d6176db174dc93cb1b899f7c6078f08654445fe8cf1b6ce98d8855f66bdbf4</ID>
      <DisplayName/>
    </Owner>
    <StorageClass>STANDARD</StorageClass>
  </Contents>
  <Contents>
    <Key>Documents/doc2.txt</Key>
    <LastModified>2019-09-10T19:21:10.000Z</LastModified>
    <ETag>"25127ad5216235937301970ce4b5049d"</ETag>
    <Size>24</Size>
    <Owner>
      <ID>02d6176db174dc93cb1b899f7c6078f08654445fe8cf1b6ce98d8855f66bdbf4</ID>
      <DisplayName/>
    </Owner>
```

The bucket policy was updated and all the objects in a bucket can be listed/viewed by an anonymous user.

8) Delete the file "welcome" from the S3 Bucket "hello-world"

Solution:

Step 1: Delete the file

Command: aws --endpoint http://192.69.97.3:9000 s3 rm s3://hello-world/welcome

```
root@attackdefense:~#
root@attackdefense:~# aws --endpoint http://192.69.97.3:9000 s3 rm s3://hello-world/welcome
delete: s3://hello-world/welcome
root@attackdefense:~#
root@attackdefense:~#
```

Step 2: List the files in hello-world bucket and check whether the file was deleted.

Command: aws --endpoint http://192.69.97.3:9000 s3 ls s3://hello-world/

```
root@attackdefense:~#
root@attackdefense:~# aws --endpoint http://192.69.97.3:9000 s3 ls s3://hello-world/
2019-09-11 00:29:55         33 flag
2019-09-11 11:17:49         12 hello
2019-09-11 01:27:41     145374 wallpaper.jpg
root@attackdefense:~#
```

References:

1. AWS CLI Reference S3 (https://docs.aws.amazon.com/cli/latest/reference/s3/index.html)
2. AWS CLI Reference S3API (https://docs.aws.amazon.com/cli/latest/reference/s3api/index.html)
3. AWS Policy Generator (https://awspolicygen.s3.amazonaws.com/policygen.html)