

Invoking Old Security Group

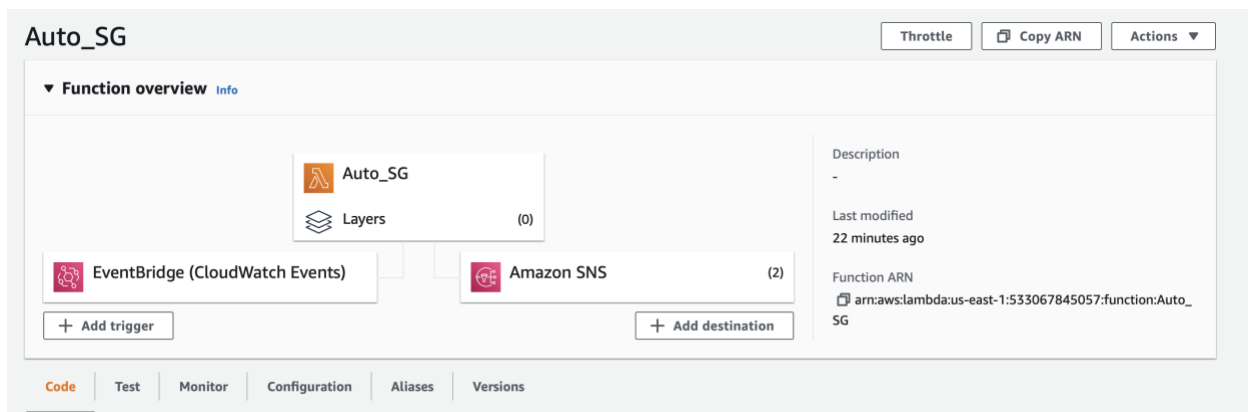
If one of your team members (inadvertently | wickedly) alters your VPC security group to enable SSH access to the world, you want the change to be instantly restored and you to be notified that the security group change was revoked.

- Someone adds a new ingress rule to your security group
- A CloudWatch event that continually monitors changes to your security groups detects the new ingress rule and invokes Lambda function
- Lambda function determines whether you are monitoring this security group
 - Reverts the new security group ingress rule.
 - Sends you an SNS Notification email to let you know what the change was, who made it, and that the change was reverted.

Requirements:

- AWS CloudTrail must be enabled in the AWS Region where the solution is deployed
- VPC with custom Security Group that we intend to monitor.
 - Note down the security group id, we will need it later to update the lambda function
- IAM Role - i.e Lambda Service Role - with EC2FullAccess permissions and SNSFull Access.
 - You may use an Inline policy with more restrictive permissions

Lambda Function:




Cloud Watch Event Configuration:

We are going to use Cloudwatch Events that will be triggered by CloudTrail API

1. Choose Create a new Rule
2. Fill the Rule Name & Rule Description
3. For Rule Type - Choose Event pattern
 1. Below that, Choose EC2 Service
 2. In the next field, Choose AWS API call via CloudTrail
4. Check the Operation box,
 1. In below field, Type/Choose both AuthorizeSecurityGroupIngress & AuthorizeSecurityGroupEgress and RevokeSecurityGroupIngress and RevokeSecurityGroupEgress.
5. Enable Trigger by Checking the box
6. Click on Add and Save the Lambda Function

☐ Trigger

**EventBridge (CloudWatch Events): SG_AUTO_VPC**
arn:aws:events:us-east-1:533067845057:rule/SG_AUTO_VPC

☐ Details

Event bus: **default**

Event pattern:

```
{
  "detail-type": [
    "AWS API Call via CloudTrail"
  ],
  "source": [
    "aws.ec2"
  ],
  "detail": {
    "eventSource": [
      "ec2.amazonaws.com"
    ]
  }
}
```

Configure Destination in Lambda:

Destinations

Info

Remove

Edit

Add destination

Q

Find destinations

	Source	Stream	Condition	Destination
<input type="radio"/>	Asynchronous invocation	-	On success	arn:aws:sns:us-east-1:533067845057:SNS_ROT
<input type="radio"/>	Asynchronous invocation	-	On failure	arn:aws:sns:us-east-1:533067845057:SNS_ROT

Once deploy Lambda , if any changes done in VPC security group it will revoke automatically and sends notification through SNS:

RoT <no-reply@sns.amazonaws.com> 19:38 (7 minutes ago) ☆ ↶ ⋮
to me ▾

```
{
  "version": "1.0",
  "timestamp": "2022-01-14T01:38:13.706Z",
  "requestContext": {
    "requestId": "52aa891f-070a-4891-8b50-fcdb8d9eac9c",
    "functionArn": "arn:aws:lambda:us-east-1:533067845057:function:Auto_SG:$LATEST",
    "condition": "Success",
    "approximateInvokeCount": 1
  },
  "requestPayload": {
    "version": "0",
    "id": "bb64f625-08cf-8dbe-aa2b-eda7c85a7d48",
    "detail-type": "AWS API Call via CloudTrail",
    "source": "aws.ec2",
    "account": "533067845057",
    "time": "2022-01-14T01:38:07Z",
    "region": "us-east-1",
    "resources": [
    ],
    "detail": {
      "eventVersion": "1.08",
      "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAXYHKRZXA4L42YVDDB:Auto_SG",
        "arn": "arn:aws:sts:533067845057:assumed-role/Auto_SG/Auto_SG",
        "accountId": "533067845057",
        "sessionName": "Auto_SG"
      },
      "webIdFederationData": {
      },
      "attributes": {
        "creationDate": "2022-01-14T01:20:37Z",
        "mfaAuthenticated": "false"
      }
    },
    "eventName": "AuthorizeSecurityGroupIngress",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "98.164.19.149",
    "userAgent": "console.ec2.amazonaws.com",
    "requestParameters": {
      "groupId": "sg-0110814afd8691bb0",
      "ipPermissions": {
        "items": [
          {
            "ipProtocol": "tcp",
            "fromPort": 80,
            "toPort": 80,
            "groups": [
            ],
            "ipRanges": {
              "items": [
                {
                  "cidrIp": "0.0.0.0/8"
                }
              ]
            },
            "ipv6Ranges": [
            ],
            "prefixListIds": [
            ]
          }
        ],
        "responseElements": {
          "requestId": "cbc8df4f-f8ed-4c1b-b787-1db733d192d9",
          "return": true,
          "securityGroupRuleSet": {
            "items": [
              {
                "groupOwnerId": "533067845057",
                "groupId": "sg-0110814afd8691bb0",
                "securityGroupRuleId": "sgr-0d66dc146d2b1cc5c",
                "isEgress": false,
                "ipProtocol": "tcp",
                "fromPort": 80,
                "toPort": 80,
                "cidrIpv4": "0.0.0.0/8"
              }
            ]
          }
        },
        "requestId": "cbc8df4f-f8ed-4c1b-b787-1db733d192d9",
        "eventId": "4abe9ffe-a75a-4eda-9cb5-ee358449b9f4",
        "readOnly": false,
        "eventType": "AwsApiCall",
        "managementEvent": true,
        "recipientAccountId": "533067845057",
        "eventCategory": "Management"
      },
      "responseContext": {
        "statusCode": 200,
        "executedVersion": "$LATEST",
        "responsePayload": null
      }
    }
  }
}
```

RoT <no-reply@sns.amazonaws.com> 19:38 (7 minutes ago) ☆ ↶ ⋮
to me ▾

```
{
  "version": "1.0",
  "timestamp": "2022-01-14T01:38:15.905Z",
  "requestContext": {
    "requestId": "da7f85dd-32a9-4da3-be66-57b0c10d843a",
    "functionArn": "arn:aws:lambda:us-east-1:533067845057:function:Auto_SG:$LATEST",
    "condition": "Success",
    "approximateInvokeCount": 1
  },
  "requestPayload": {
    "version": "0",
    "id": "27e318dd-adfb-1f1b-86b9-441b6d511b7d",
    "detail-type": "AWS API Call via CloudTrail",
    "source": "aws.ec2",
    "account": "533067845057",
    "time": "2022-01-14T01:38:13Z",
    "region": "us-east-1",
    "resources": [
    ],
    "detail": {
      "eventVersion": "1.08",
      "userIdentity": {
        "type": "AssumedRole",
        "principalId": "AROAXYHKRZXA4L42YVDDB:Auto_SG",
        "arn": "arn:aws:sts:533067845057:assumed-role/Auto_SG/Auto_SG",
        "accountId": "533067845057",
        "sessionName": "Auto_SG"
      },
      "webIdFederationData": {
      },
      "attributes": {
        "creationDate": "2022-01-14T01:38:10Z",
        "mfaAuthenticated": "false"
      }
    },
    "eventName": "RevokeSecurityGroupIngress",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "3.86.86.19",
    "userAgent": "Boto3/1.18.55 Python/3.9.8 Linux/4.14.252-207.481.amzn2.x86_64 exec-env/AWS_Lambda_python3.9 Botocore/1.21.55",
    "requestParameters": {
      "groupId": "sg-0110814afd8691bb0",
      "ipPermissions": {
        "items": [
          {
            "ipProtocol": "tcp",
            "fromPort": 80,
            "toPort": 80,
            "groups": [
            ],
            "ipRanges": {
              "items": [
                {
                  "cidrIp": "0.0.0.0/8"
                }
              ]
            },
            "ipv6Ranges": [
            ],
            "prefixListIds": [
            ]
          }
        ],
        "responseElements": {
          "requestId": "ed02eef6-a2b6-47b4-9d46-b1f0a8a3b1fd",
          "return": true,
          "requestId": "ed02eef6-a2b6-47b4-9d46-b1f0a8a3b1fd",
          "eventId": "0c1799a6-54fb-41a2-adcb-3475a1c6f90e",
          "readOnly": false,
          "eventType": "AwsApiCall",
          "managementEvent": true,
          "recipientAccountId": "533067845057",
          "eventCategory": "Management"
        },
        "responseContext": {
          "statusCode": 200,
          "executedVersion": "$LATEST",
          "responsePayload": null
        }
      }
    }
  }
}
```