

ATTACK DEFENSE

by PentesterAcademy

ATTACK DEFENSE LABS COURSES
PENTESTER ACADEMY TOOL BOX PENTESTING
JOINT WORLD-CLASS TRAINERS TRAINING HACKER
TOOL BOX PATV HACKER
HACKER PENTESTING
PATV RED TEAM LABS ATTACK DEFENSE LABS
TRAINING COURSES ACCESS POINT PENTESTER
TEAM LABS PENTESTER ACADEMY ATTACK DEFENSE LABS
GACCESS POINT TOOL BOX WORLD-CLASS TRAINERS
WORLD-CLASS TRAINERS
ATTACK DEFENSE LABS TRAINING COURSES PATV ACCESS
PENTESTER ACADEMY TOOL BOX PENTESTING
ATTACK DEFENSE LABS TRAINING COURSES PENTESTER ACADEMY
COURSES PENTESTER ACADEMY TOOL BOX PENTESTING
TOOL BOX HACKER PENTESTING
PATV RED TEAM LABS ATTACK DEFENSE LABS
COURSES PENTESTER ACADEMY
PENTESTER ACADEMY ATTACK DEFENSE LABS
WORLD-CLASS TRAINERS
RED TEAM TRAINING COURSES
PENTESTER ACADEMY TOOL BOX
PENTESTING

Name	IAM Enumeration
URL	https://attackdefense.com/challengedetails?cid=2245
Type	AWS Cloud Security : IAM

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.

Solution:

Console Based Enumeration

Step 1: Click on the lab link button to get access to the AWS lab credentials.

Access Credentials to your AWS lab Account

Login URL	https://276384657722.signin.aws.amazon.com/console
Region	Asia Pacific (Singapore) ap-southeast-1
Username	geDJMDdiVATcfSOUoldk
Password	dG3v6mFrNFY2Tcir
Access Key ID	AKIAUAWOPGE5DA2GTYSQ
Secret Access Key	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

Step 2: Sign-in into the AWS console.

Sign in as IAM user

Account ID (12 digits) or account alias

276384657722

IAM user name

geDJMDdiVATcfSOUoldk

Password

Sign in

[Sign in using root user email](#)

[Forgot password?](#)



Step 3: Search for the IAM dashboard and navigate to it.

A screenshot of the AWS search interface. The search bar at the top contains the text "iam". Below the search bar, there are two main sections: "Services" and "Features". The "Services" section shows one result: "IAM" with the subtext "Manage access to AWS resources". The "Features" section shows one result: "Groups" with the subtext "IAM feature". There is also a link "See all 10 results ▶" in the "Features" section.

IAM dashboard

Sign-in URL for IAM users in this account

<https://276384657722.signin.aws.amazon.com/console>  | [Customize](#)

IAM resources

Users: 24

Roles: 48

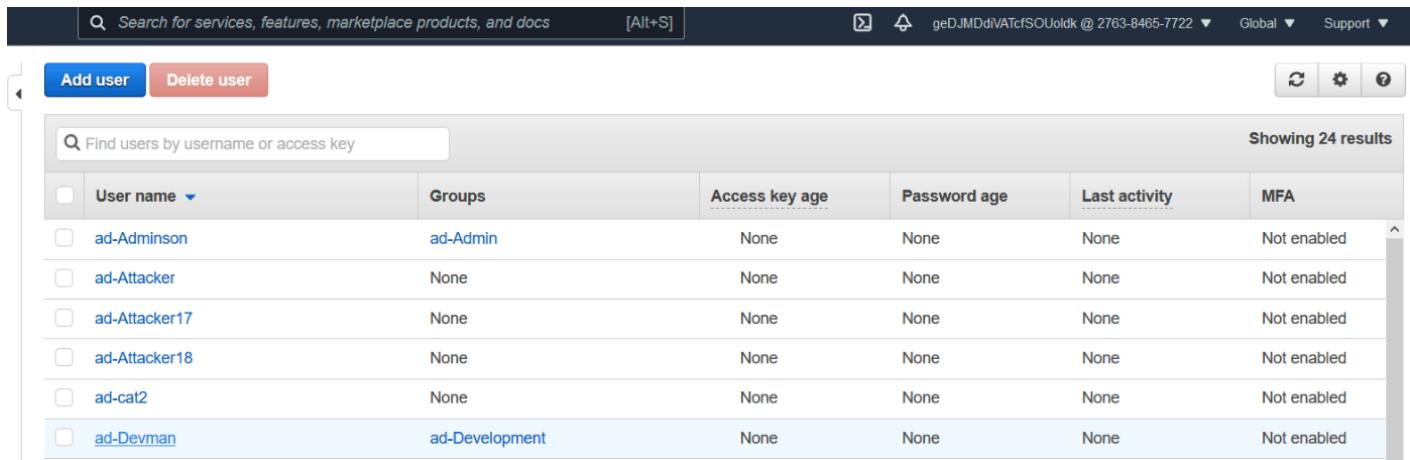
Groups: 6

Identity providers: 0

[Customer managed policies: 33](#)

Security alerts

Step 4: Click on users.



The screenshot shows the AWS IAM Users page. At the top, there is a search bar and navigation links for Global and Support. Below the header, there are two buttons: "Add user" (blue) and "Delete user" (red). A search bar with the placeholder "Find users by username or access key" is present. To the right, it says "Showing 24 results". The main area is a table with the following columns: User name, Groups, Access key age, Password age, Last activity, and MFA. The table lists six users, each with a checkbox next to their name. The users are: ad-Adminson, ad-Attacker, ad-Attacker17, ad-Attacker18, ad-cat2, and ad-Devman. All users listed have "None" in the Access key age, Password age, and Last activity columns, and "Not enabled" in the MFA column.

<input type="checkbox"/>	User name	Groups	Access key age	Password age	Last activity	MFA
<input type="checkbox"/>	ad-Adminson	ad-Admin	None	None	None	Not enabled
<input type="checkbox"/>	ad-Attacker	None	None	None	None	Not enabled
<input type="checkbox"/>	ad-Attacker17	None	None	None	None	Not enabled
<input type="checkbox"/>	ad-Attacker18	None	None	None	None	Not enabled
<input type="checkbox"/>	ad-cat2	None	None	None	None	Not enabled
<input type="checkbox"/>	ad-Devman	ad-Development	None	None	None	Not enabled

Step 5: To enumerate the user click on the username.

User ARN arn:aws:iam::276384657722:user/ad-Adminson

Path /

Creation time 2021-01-20 11:31 UTC+0530

Permissions Groups (1) Tags Security credentials Access Advisor

▼ Permissions policies (1 policy applied)

Add permissions Add inline policy

Policy name	Policy type
Attached from group	
AdministratorAccess	AWS managed policy from group ad-Admin

▶ Permissions boundary (not set)

Check user permission and policies.

Step 6: Check groups for the user.

Summary

User ARN arn:aws:iam::276384657722:user/ad-Adminson

Path /

Creation time 2021-01-20 11:31 UTC+0530

Permissions Groups (1) Tags Security credentials Access Advisor

Add user to groups

Group name	Attached permissions
ad-Admin	AdministratorAccess

Step 7: Check the user's security credentials.

Summary

User ARN: arn:aws:iam::276384657722:user/ad-Adminson [Edit](#)

Path: /

Creation time: 2021-01-20 11:31 UTC+0530

Permissions Groups (1) Tags Security credentials **Access Advisor**

Sign-in credentials

Summary	• User does not have console management access
Console password	Disabled Manage
Assigned MFA device	Not assigned Manage
Signing certificates	None Edit

Step 8: Similarly check for other user's permissions.

Summary

User ARN: arn:aws:iam::276384657722:user/ad-User [Edit](#)

Path: /

Creation time: 2021-01-20 11:31 UTC+0530

Permissions Groups Tags Security credentials **Access Advisor**

▼ Permissions policies (2 policies applied)

[Add permissions](#)

Policy name ▾	Policy type ▾
Attached directly	
▶ AmazonS3ReadOnlyAccess	AWS managed policy
▶ ad-UserDeneid	Inline policy

Step 9: Check groups for the user.

Summary

User ARN arn:aws:iam::276384657722:user/ad-User [Edit](#)

Path /

Creation time 2021-01-20 11:31 UTC+0530

Permissions Groups Tags Security credentials Access Advisor

Add user to groups

Group name	Attached permissions
	No results

Step 10: Check the user's security credentials.

Summary

User ARN arn:aws:iam::276384657722:user/ad-User [Edit](#)

Path /

Creation time 2021-01-20 11:31 UTC+0530

Permissions Groups Tags Security credentials Access Advisor

Sign-in credentials

Summary	
Console sign-in link: https://276384657722.signin.aws.amazon.com/console	Edit
MFA is required when signing in.	Learn more

Console password Enabled (never signed in) | [Manage](#)

Assigned MFA device arn:aws:iam::276384657722:mfa/ad-User (Virtual) | [Manage](#)

Signing certificates [UFIGCN7PBNLWIIMKL44Y2V2WHF55BAAWF](#) [Edit](#)
Active
2021-01-20 11:59 UTC+0530

Step 11: Check the user's access keys and ssh keys.

Access keys

Use access keys to make secure REST or HTTP Query protocol requests to AWS service APIs. For your protection, you should never share your secret keys with anyone. As a best practice, we recommend frequent key rotation. [Learn more](#)

[Create access key](#)

Access key ID	Created	Last used	Status	
AKIAUAWOPGE5PDVCCFLW	2021-01-20 11:31 UTC+0530	N/A	Active	Make inactive X

SSH keys for AWS CodeCommit

Use SSH public keys to authenticate access to AWS CodeCommit repositories. [Learn more](#)

[Upload SSH public key](#)

SSH key ID	Uploaded	Status	
APKAUAWOPGE5M47NZEIT	2021-01-20 11:31 UTC+0530	Active	Make inactive X

Access key ID
AKIAUAWOPGE5PDVCCFLW

SSH keys for AWS CodeCommit
Use SSH public keys to authenticate access to AWS CodeCommit repositories.
[Upload SSH public key](#)

SSH key ID
APKAUAWOPGE5M47NZEIT

HTTPS Git credentials
Generate a user name and password
[more](#)
[Generate credentials](#)

No credentials have been generated.

Credentials for Amazon S3

Show SSH Key

-----BEGIN PUBLIC KEY-----

```
MIICljANBgkqhkiG9w0BAQEFAAOCAg8AMIIICgKCAGEAx/wOS/NpqoCDYrX8J2GMX5fpy9+K+6bAkqlrX7EnIEhcnmwlvKhKYLsmGjSELfcpsGUwMcj5ad+GjCBSY+/it12DA2mj5sKN74rQlG+zpC0NPn7lyMmAsibQS7cjJx88REko7t9XagXGv2JEPzYwWEnrVBh5E2ojdatTNDJc5nr9F8FXxHwKnqUIulrl1ezG+EyLuyDz8bgWeTyJFN6VC3PckFyYnid9p1n6Lma2TwNvs1g+iYudoMfHkC3inEZHJOaGh7ZZ5IMMK7N8SfnvdC0Usal8PkZRLIo6vPBtaydic9MDK9tP6Hqr+Ba+XiR62egx5bGGuq4p0a67BNOQP1t6dd+aOY9udJGwh/p0aZRztIL+Ht65etkyZ4bR8fBUzp90Yd96guto4hzPbqFObKdRVAervvxsum1WwzW7SAZQ2xUBVwArSPCK53eRSEiKenydtjTguwF5FHe6QKZqjpdSXNdoMmtmL1NiL0RxooN0/NQWaReU2XhUD3vyx154DsxEUEhs1GjKSkCEyL8Vq6wkqeEcjYpV5fNsuxqsgClUHNqv7WiJTJXbQiIyNQSY22vmPAuKfVATIZtjMhvKef0x529oiVApb2hLjW2NUiF/jBH2HDiuWns5joCvsWmTNywoobibfTaFmq/w2RXo6dV+63VxNWgmtbKoEg/uV8CAwEAA==
```

-----END PUBLIC KEY-----

[Cancel](#)

Step 12: Click on groups on the left panel to enumerate groups.

The screenshot shows the AWS Identity and Access Management (IAM) Groups page. On the left, there's a navigation sidebar with options like Dashboard, Access management (Groups, Users, Roles, Policies, Identity providers, Account settings), and Group Actions. The main area displays a table of groups:

	Group Name	Users
<input type="checkbox"/>	ad-Admin	1
<input type="checkbox"/>	ad-Development	2
<input type="checkbox"/>	ad-Production	1
<input type="checkbox"/>	ad-Tester	2
<input type="checkbox"/>	ad-Users	0

Step 13: Click on the group name to open group details.

The screenshot shows the detailed view for the 'ad-Admin' group. It includes fields for Group ARN (arn:aws:iam::276384657722:group/ad-Admin), Users (in this group) (1), Path (/), and Creation Time (2021-01-20 11:31 UTC+0530). Below this, there are tabs for Users, Permissions, and Access Advisor. The 'Users' tab is selected, showing a table with one user entry:

User	Actions
ad-Adminson	Remove User from Group

Check for the users that are part of the group.

Step 14: Check for the group attached policies.

▼ Summary

Group ARN: arn:aws:iam::276384657722:group/ad-Admin [Edit](#)

Users (in this group): 1

Path: /

Creation Time: 2021-01-20 11:31 UTC+0530

Users

Permissions

Access Advisor

Managed Policies

The following managed policies are attached to this group. You can attach up to 10 managed policies.

[Attach Policy](#)

Policy Name	Actions
AdministratorAccess	Show Policy Detach Policy Simulate Policy

Inline Policies

Step 15: Similarly check for the other groups.

▼ Summary

Group ARN: arn:aws:iam::276384657722:group/ad-Development [Edit](#)

Users (in this group): 2

Path: /

Creation Time: 2021-01-20 11:31 UTC+0530

Users

Permissions

Access Advisor

This view shows all users in this group: **2 Users**

User	Actions
ad-Devwoman	Remove User from Group
ad-Devman	Remove User from Group

▼ Summary

Group ARN: arn:aws:iam::276384657722:group/ad-Development 

Users (in this group): 2

Path: /

Creation Time: 2021-01-20 11:31 UTC+0530

Users

Permissions

Access Advisor

Managed Policies

The following managed policies are attached to this group. You can attach up to 10 managed policies.

Attach Policy

Policy Name	Actions
 AmazonS3FullAccess	Show Policy Detach Policy Simulate Policy

Step 16: Click on policies in the left pane to enumerate policies.

Identity and Access Management (IAM)

Dashboard

▼ Access management

Groups

Users

Roles

Policies

Identity providers

Account settings

Create policy

Policy actions ▾

Filter policies ▾

 Search

	Policy name ▾	Type	Used as
<input type="radio"/>	 AccessAnalyzerServiceRoleP...	AWS managed	Permissions policy (1)
<input type="radio"/>	 ad-customer-managed-policy	Customer managed	None
<input type="radio"/>	 AdministratorAccess	Job function	Permissions policy (3)
<input type="radio"/>	 AdministratorAccess-Amplify	AWS managed	None
<input type="radio"/>	 AdministratorAccess-AWSElas...	AWS managed	None
<input type="radio"/>	 AlexaForBusinessDeviceSetup	AWS managed	None

Steps 17: Check for customer-managed and AWS managed policies.

Filter policies					Search	Show
	Policy name	Type	Used as	Description		
○	AccessAnalyzerServiceRoleP...	AWS managed	Permissions policy (1)	Allow Access Analyzer to analyze resource metadata		
●	ad-customer-managed-policy	Customer managed	None			
○	AdministratorAccess	Job function	Permissions policy (3)	Provides full access to AWS services and resources.		
○	AdministratorAccess-Amplify	AWS managed	None	Grants account administrative permissions while explicitly all		
○	AdministratorAccess-AWSElas...	AWS managed	None	Grants account administrative permissions. Explicitly allows		

Step 18: Click on the policy name to enumerate the policy.

Policies > ad-customer-managed-policy

Summary

Policy ARN arn:aws:iam::276384657722:policy/ad-customer-managed-policy [Edit](#)

Description

Permissions Policy usage Policy versions Access Advisor

Policy summary { } JSON Edit policy [?](#)

Filter

Service	Access level	Resource	Request condition
Allow (1 of 267 services) Show remaining 266	S3	Limited: Read	Multiple
			None

Step 19: Check policy document, policy usages and policy permissions.

Permissions	Policy usage	Policy versions	Access Advisor
Back S3			
Policy summary { } JSON Edit policy			
<input type="text"/> Filter			
Action (4 of 107) Show remaining 103	Resource	Request condition	
Read (4 of 43 actions)			
GetBucketPolicy	BucketName string like file-uploader-saved-files	None	
GetObject	BucketName string like file-uploader-saved-files, ObjectPath string like All	None	
GetObjectAcl	BucketName string like file-uploader-saved-files, ObjectPath string like All	None	
GetObjectVersion	BucketName string like file-uploader-saved-files, ObjectPath string like All	None	

Permissions Policy usage Policy versions Access Advisor

▼ Permissions

Attach Detach

Filter: [Filter](#) Search

<input type="checkbox"/>	Name	Type
		No results

▶ Permissions boundaries

The screenshot shows the 'Policy versions' tab selected in the AWS IAM interface. It displays a table with one version entry:

	Version	Creation time
<input type="checkbox"/>	Version 1 (Default)	2021-01-20 13:21 UTC+0530

The JSON policy document for Version 1 (Default) is shown in a modal window:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "s3:GetObjectAcl",
        "s3:GetObject",
        "s3:GetBucketPolicy",
        "s3:GetObjectVersion"
      ]
    }
  ]
}
```

Step 20: Similarly enumerate other AWS managed policies.

Summary

The screenshot shows the 'Policy summary' tab selected for the 'AdministratorAccess' policy. It displays the following details:

- Policy ARN:** arn:aws:iam::aws:policy/AdministratorAccess
- Description:** Provides full access to AWS services and resources.

The 'Policy versions' tab is also visible at the top.

The 'JSON' button is highlighted in the summary section.

The main table shows the following permissions:

Service	Access level	Resource	Request condition
Allow (267 of 267 services)			
Access Analyzer	Full access	All resources	None
Account	Full access	All resources	None

Step 21: Check policy document, policy usages and policy permissions.

The screenshot shows the AWS IAM console with the "Policy usage" tab selected. At the top, there are four tabs: "Permissions", "Policy usage" (selected), "Policy versions", and "Access Advisor". Below the tabs, a section titled "Permissions (3)" is shown with the instruction "Attach this policy to an IAM entity to apply its permissions to the entity. [Learn more](#)". There are "Attach" and "Detach" buttons. A search bar and a filter dropdown are also present. A table lists three entities: "ad-Admin" (Group) and "CajDeploy" (User). The "Name" column has checkboxes, and the "Type" column has dropdowns.

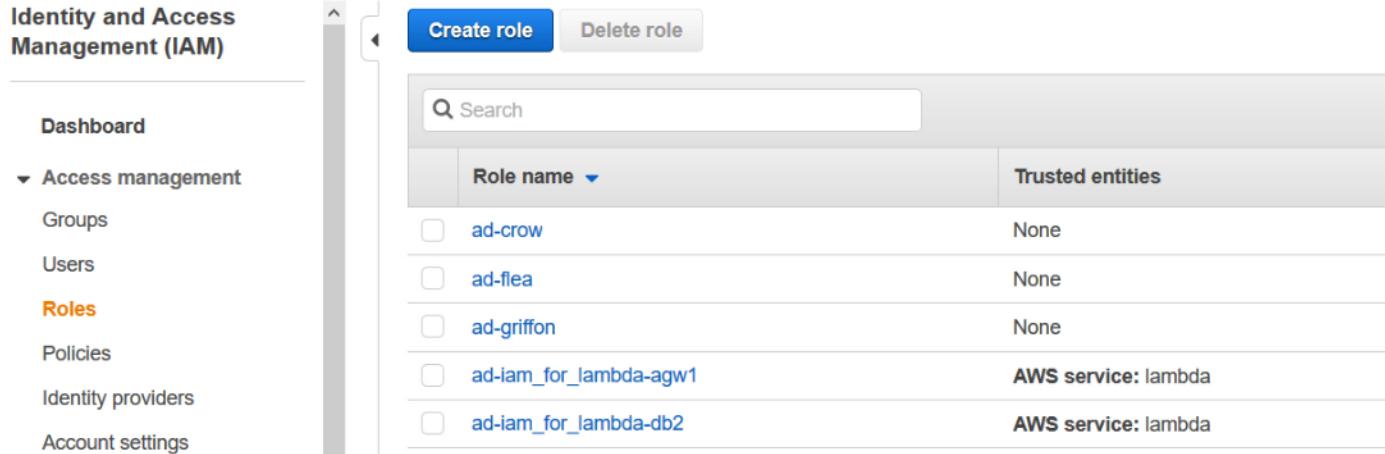
	Name	Type
<input type="checkbox"/>	ad-Admin	Group
<input type="checkbox"/>	CajDeploy	User

Summary

The screenshot shows the "Policy summary" tab selected under the "Policy usage" tab. It displays the "Policy ARN" (arn:aws:iam::aws:policy/AdministratorAccess) and a "Description" (Provides full access to AWS services and resources). Below this, there are four tabs: "Permissions", "Policy usage" (selected), "Policy versions", and "Access Advisor". The "Policy summary" tab includes a "JSON" button and a "Filter" search bar. A table below shows access levels for various services and accounts.

Service	Access level	Resource	Request condition
Allow (267 of 267 services)			
Access Analyzer	Full access	All resources	None
Account	Full access	All resources	None

Step 22: Click on roles on the left panels to enumerate roles.



A screenshot of the AWS Identity and Access Management (IAM) service. On the left, a sidebar shows navigation options: Dashboard, Access management (Groups, Users, Roles, Policies, Identity providers), and Account settings. The 'Roles' option is highlighted in orange. The main content area has a title 'Create role' and a 'Delete role' button. Below is a search bar and a table titled 'Role name'. The table lists five roles: 'ad-crow', 'ad-flea', 'ad-griffon', 'ad-iam_for_lambda-agw1', and 'ad-iam_for_lambda-db2'. The 'ad-iam_for_lambda-*' roles are associated with the 'AWS service: lambda'.

Role name	Trusted entities
ad-crow	None
ad-flea	None
ad-griffon	None
ad-iam_for_lambda-agw1	AWS service: lambda
ad-iam_for_lambda-db2	AWS service: lambda

Step 23: Click on role name to enumerate roles for the AWS account.

Roles > ad-crow

Summary

Role ARN	arn:aws:iam::276384657722:role/ad-crow	
Role description	Edit	
Instance Profile ARNs		
Path	/	
Creation time	2021-01-20 11:45 UTC+0530	
Last activity	Not accessed in the tracking period	
Maximum session duration	1 hour	Edit

[Permissions](#)

[Trust relationships](#)

[Tags](#)

[Access Advisor](#)

[Revoke sessions](#)

▼ Permissions policies

 [Get started with permissions](#)

Step 24: Check for the role trust policy.

Permissions Trust relationships Tags Access Advisor Revoke sessions

You can view the trusted entities that can assume the role and the access conditions for the role. [Show policy document](#)

[Edit trust relationship](#)

Trusted entities

The following trusted entities can assume this role.

Trusted entities

None

Conditions

The following conditions define |

There are no conditions associa

Step 25: Check role's trusted entities.

Role name	Trusted entities	Last activity
<input type="checkbox"/> ad-crow	None	None
<input type="checkbox"/> ad-flea	None	None
<input type="checkbox"/> ad-griffon	None	None
<input checked="" type="checkbox"/> ad-iam_for_lambda-agw1	AWS service: lambda	None
<input type="checkbox"/> ad-iam_for_lambda-db2	AWS service: lambda	None
<input type="checkbox"/> ad-lbex	None	None
<input type="checkbox"/> ad-LoggingRole	Account: *	8 days

Summary

Role ARN	arn:aws:iam::276384657722:role/ad-iam_for_lambda-db2 Edit
Role description	Edit
Instance Profile ARNs	Edit
Path	/
Creation time	2021-01-27 08:13 UTC+0530
Last activity	Not accessed in the tracking period
Maximum session duration	1 hour Edit

[Permissions](#) [Trust relationships](#) [Tags](#) [Access Advisor](#) [Revoke sessions](#)

▼ Permissions policies (1 policy applied)

[Attach policies](#)

Policy name	Policy type
▶ terraform-20210127024317343500000001	Inline policy

Step 26: Check for the role's inline policies.

Policy name	Policy type		
▶ terraform-20210127024317343500000001	Inline policy X		
Policy summary { } JSON Edit policy Simulate policy			
<input type="text"/> Filter			
Service	Access level	Resource	Request condition
Allow (1 of 267 services) Show remaining 266			
CloudWatch Logs	Limited: Write	All resources	None

Step 27: Check for the role's trusted entities.



Permissions Trust relationships Tags Access Advisor Revoke sessions

You can view the trusted entities that can assume the role and the access conditions for the role. [Show policy document](#)

Edit trust relationship

Trusted entities

The following trusted entities can assume this role.

Trusted entities

[The identity provider\(s\) lambda.amazonaws.com](#)

Conditions

The following conditions define how and when true.

There are no conditions associated with this role.

Step 28: Similarly enumerate other roles.

Give this link to users who can switch roles in the console <https://signin.aws.amazon.com/switchrole?roleName=ad-LoggingRole&account=276384657722> 

Permissions Trust relationships Tags Access Advisor Revoke sessions

▼ Permissions policies (1 policy applied)

Attach policies

Policy name	Policy type
ad-ReadS3Bucket	Inline policy

[Policy summary](#) [{ } JSON](#) [Edit policy](#)

Filter

Service	Access level	Resource	Request condition
Allow (1 of 267 services) Show remaining 266			
S3	Limited: List, Read	Multiple	None

Permissions Trust relationships Tags Access Advisor Revoke sessions



Overly Permissive policy

Broad access: Principals that include a wildcard (*, ?) can be overly permissive.

You can view the trusted entities that can assume the role and the access conditions for the role. [Show policy document](#)

[Edit trust relationship](#)

Trusted entities

The following trusted entities can assume this role.

Trusted entities

*

Conditions

The following conditions define how and when trus

There are no conditions associated with this role.

Note: Roles with * as trusted entities can be assumed by any AWS resource which can be inside or outside an organisation.

CLI Based Enumeration

Step 1: Click on the lab link button to get access to the AWS lab credentials.

Access Credentials to your AWS lab Account

Login URL	https://276384657722.signin.aws.amazon.com/console
Region	Asia Pacific (Singapore) ap-southeast-1
Username	qiVMHVacHfwkveFcsASM
Password	Um7uYTIKeMjU35Jc
Access Key ID	AKIAUAWOPGE5KB43QAEB
Secret Access Key	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

Step 2: Configure AWS CLI to use the provided credentials.

Command: aws configure

```
⚡ root@Kali ➔ aws configure
AWS Access Key ID [*****K2JI]: AKIAUAWOPGE5KB43QAEB
AWS Secret Access Key [*****Ing]: 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
Default region name [us-east-1]: ap-southeast-1
Default output format [None]:
⚡ root@Kali ➔
```

Step 3: Get a list of the users on the AWS account.

Command: aws iam list-users

	Access Key ID	Secret Access Key
{ "Path": "/", "UserName": "LabAdmin", "UserId": "AIDAUAWOPGE5NDH7SQWF4", "Arn": "arn:aws:iam::276384657722:user/LabAdmin", "CreateDate": "2020-09-10T00:40:21Z", "PasswordLastUsed": "2021-02-09T06:18:32Z" }, { "Path": "/", "UserName": "qiVMHVacHfwkveFcsASM", "UserId": "AIDAUAWOPGE5N33FFBIDY", "Arn": "arn:aws:iam::276384657722:user/qiVMHVacHfwkveFcsASM", "CreateDate": "2021-02-13T02:58:49Z" }]	AKIAUAWOPGE5KB43QAEB	w1OqQUFmc4uQaeJ4iHresM

```
⚡ root@Kali ➔
```

Briefly check user name, user id and ARNs.

Step 4: Check groups for users.

Command: aws iam list-groups-for-user --user-name ad-adminson

```
root@Kali:~$ aws iam list-groups-for-user --user-name ad-adminson
{
    "Groups": [
        {
            "Path": "/",
            "GroupName": "ad-Admin",
            "GroupId": "AGPAUAWOPGE5CRFDP2XCR",
            "Arn": "arn:aws:iam::276384657722:group/ad-Admin",
            "CreateDate": "2021-01-20T06:01:45Z"
        }
    ]
}
```

Step 5: Check policies attached to the user.

Command: aws iam list-attached-user-policies --user-name ad-user

```
root@Kali:~$ aws iam list-attached-user-policies --user-name ad-user
{
    "AttachedPolicies": [
        {
            "PolicyName": "AmazonS3ReadOnlyAccess",
            "PolicyArn": "arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess"
        }
    ]
}
```

Step 6: Check for any signing certificates for the user.

Command: aws iam list-signing-certificates --user-name ad-user

```
root@Kali:~$ aws iam list-signing-certificates --user-name ad-user
{
    "Certificates": [
        {
            "UserName": "ad-User",
            "CertificateId": "UFICN7PBNLWIIMKL44Y2V2WHF55BAWF",
            "CertificateBody": "-----BEGIN CERTIFICATE-----\nMIIDETCCAfkCFDWs8Ss6Cjinu2B/2zNYM/yAKePKMA0GCSq5dCBXaWRnaXRzIFB0eSBMdGQwHhcNMjEwMTIwMDYyOTI3WhcNMjIwMTIwMDYy\nnOTI3WjBFMQswCQYDVQQGEwJBVTETMBEGA1UECAwKU2tnAQ8AMIIIBCgKCAQEAA4CkwJtCjmLNzrGReKKK+xSbEmXIJDH6a5JyRAegXKh5Xke/\nnagupZJwIgraKZ9tXeEiASVlyL/1bOTRRVHAcK2xXO\\n8Yg0Kz/3f8wRYighGbyMIPPA4/OP+XiweBWpzhQ1YTBlLzSbWfsu3w6C97+V99D\\nztGNexMgbNJ5w6eBUWgjcmPblzvCK6c+Qy0nmJBCwUA\\nA4IBAQAAye2trrt8T0PIcawSooIr+DD7/9ZP3yiCcylBR+tHl4u50YcZaae0e0\\nSjSRe2uUiMGxd8kaaoe1t8uCdRfHTP0RnELyJSnwTz0\\n7SXZK2/inWE9HpwTb6ujozZ8A6hDh6VgvcjZp1lam3gJn1csqb3NvcSMZsMwt4dv\\nyD0BJQKUALKQ/8/txppEwQCB0tsgr1x6--\\n",
            "Status": "Active",
            "UploadDate": "2021-01-20T06:29:43Z"
        }
    ]
}
```

Step 7: Check for any public ssh keys for the user.

Command: aws iam list-ssh-public-keys --user-name ad-user

```
⚡ root@Kali ~ ➔ aws iam list-ssh-public-keys --user-name ad-user
{
  "SSHPublicKeys": [
    {
      "UserName": "ad-User",
      "SSHPublicKeyId": "APKAUAWOPGE5M47NZEIT",
      "Status": "Active",
      "UploadDate": "2021-01-20T06:01:51Z"
    }
  ]
}
```

Step 8: Get ssh key details.

Command: aws iam get-ssh-public-key --user-name ad-user --encoding PEM
--ssh-public-key-id APKAUAWOPGE5M47NZEIT

```
⚡ root@Kali ~ ➔ aws iam get-ssh-public-key --user-name ad-user --encoding PEM --ssh-public-key-id APKAUAWOPGE5M47NZEIT
{
  "SSHPublicKey": {
    "UserName": "ad-user",
    "SSHPublicKeyId": "APKAUAWOPGE5M47NZEIT",
    "Fingerprint": "ba:1f:d9:94:62:7a:78:70:fc:ef:6b:72:9a:29:06:5b",
    "SSHPublicKeyBody": "-----BEGIN PUBLIC KEY-----\nMIICjANBgkqhkiG9w0BAQEAAQgBAMIIICgKCAGEAx/W0S/NpqoCDYrX8J2GM\nmjsKN74rQlG+zcP0NPn7lyMmAsibQS7cjJx88REko7t9XagXGv2JEPzYwW\\nEnrVh5E2o\njdatTNDJcK5nr9F8FXxHWKnqUIulrL1ezG+EyLuyDz8bGweTyJFN6V\n0Usal8PkZRLI06vPBtaydic9MDK9tP6HqR+Ba+XiR62egx5bGGuq4p0a67BNO\\nQP1t6dd+aOY9udJGwn/p0aZRZtIL+Ht65etkyZ4bR8fBUzp90Yd96guto4hzP\nqjpdSXNdoMmtmL1NiL0RxooN0/NQWaReU2XhUD3vyx154DsxEHsLgjKSckCeyL8V\\nq6wkqeEcjYpV5fNsuxqxcLuhNqV7WiJTJXbQiIyNQSY22vmPAuKfVATIZ\nX\\no6dV+63VxNWgmtbKoEg/uV8CAwEAAQ=\n-----END PUBLIC KEY-----\n",
    "Status": "Active",
    "UploadDate": "2021-01-20T06:01:51Z"
  }
}
```

Step 9: Check for MFA devices for users.

Command: aws iam list-virtual-mfa-devices

```
⚡ root@Kali ➜ aws iam list-virtual-mfa-devices
{
    "VirtualMFADevices": [
        {
            "SerialNumber": "arn:aws:iam::276384657722:mfa/ad-User",
            "User": {
                "Path": "/",
                "UserName": "ad-User",
                "UserId": "AIDAUAWOPGE5E7QHAWL43",
                "Arn": "arn:aws:iam::276384657722:user/ad-User",
                "CreateDate": "2021-01-20T06:01:45Z"
            },
            "EnableDate": "2021-01-20T06:47:34Z"
        }
    ]
}
```

Access Credentials

Step 10: Check for user login profile.

Command: aws iam get-login-profile --user-name ad-user

```
⚡ root@Kali ➜ aws iam get-login-profile --user-name ad-user
{
    "LoginProfile": {
        "UserName": "ad-user",
        "CreateDate": "2021-01-20T06:01:51Z",
        "PasswordResetRequired": false
    }
}
⚡ root@Kali ➜
```

Step 11: Enumerate groups for the AWS account.

Command: aws iam list-groups

```
⚡ root@Kali ➜ aws iam list-groups
{
    "Groups": [
        {
            "Path": "/",
            "GroupName": "ad-Admin",
            "GroupId": "AGPAUAWOPGE5CRFDP2XCR",
            "Arn": "arn:aws:iam::276384657722:group/ad-Admin",
            "CreateDate": "2021-01-20T06:01:45Z"
        },
        ...
    ]
}
```

<pre>{ "Path": "/", "GroupName": "ad-Development", "GroupId": "AGPAUAWOPGE5MUURO5EM6", "Arn": "arn:aws:iam::276384657722:group/ad-Development", "CreateDate": "2021-01-20T06:01:47Z" }, { "Path": "/", "GroupName": "ad-Production", "GroupId": "AGPAUAWOPGE5AP2H5GNNP", "Arn": "arn:aws:iam::276384657722:group/ad-Production", "CreateDate": "2021-01-20T06:01:45Z" }</pre>	Access Credentials to your AWS Account				
	<table border="1"> <tr> <td>Login URL</td> <td>https://23.83.84.16577</td> </tr> <tr> <td>Region</td> <td>Asia Pacific (Singapore)</td> </tr> </table>	Login URL	https://23.83.84.16577	Region	Asia Pacific (Singapore)
Login URL	https://23.83.84.16577				
Region	Asia Pacific (Singapore)				

Step 12: Check which policies are attached to the group to enumerate permissions.

Commands:

```
aws iam list-group-policies --group-name ad-admin
aws iam list-attached-group-policies --group-name ad-admin
```

<pre>x \$ root@Kali ~ ➔ aws iam list-group-policies --group-name ad-admin { "PolicyNames": [] } \$ root@Kali ~ ➔ aws iam list-attached-group-policies --group-name ad-admin { "AttachedPolicies": [{ "PolicyName": "AdministratorAccess", "PolicyArn": "arn:aws:iam::aws:policy/AdministratorAccess" }] }</pre>	Access Credentials to your AWS Account
---	--

Step 13: Enumerate policies for the AWS account.

Command: aws iam list-policies

<pre>{ "PolicyName": "AWSCodeArtifactReadOnlyAccess", "PolicyId": "ANPAZKAPJZG4PVTKOJHFB", "Arn": "arn:aws:iam::aws:policy/AWSCodeArtifactReadOnlyAccess", "Path": "/", "DefaultVersionId": "v1", "AttachmentCount": 0, "PermissionsBoundaryUsageCount": 0, "IsAttachable": true, "CreateDate": "2020-06-25T21:23:52Z", "UpdateDate": "2020-06-25T21:23:52Z" },</pre>	Access Credentials to your AWS Account
---	--

```

{
    "PolicyName": "AmazonRedshiftDataFullAccess",
    "PolicyId": "ANPAZKAPJZG4PX5LA5SG6",
    "Arn": "arn:aws:iam::aws:policy/AmazonRedshiftDataFullAccess",
    "Path": "/",
    "DefaultVersionId": "v1",
    "AttachmentCount": 0,
    "PermissionsBoundaryUsageCount": 0,
    "IsAttachable": true,
    "CreateDate": "2020-09-09T19:23:55Z",
    "UpdateDate": "2020-09-09T19:23:55Z"
}
]
> root@Kali ~ ~ ~

```

Without any filter the command will return all the IAM managed policies (AWS Managed Policy + Customer Managed Policies)

Step 14: Searching for customer managed policies

Command: aws iam list-policies --scope Local | grep -A2 PolicyName

```

"PolicyName": "AWSLambdaBasicExecutionRole-17ebe2f2-d0e4-4f65-a7e4-096fa5b0fed",
"PolicyId": "ANPAUAWOPGE5LGN4WP4YI",
"Arn": "arn:aws:iam::276384657722:policy/service-role/AWSLambdaBasicExecutionRole-17ebe2f2-d0e4-4f65-a7e4-096fa5b0fed",
"PolicyName": "ad-customer-managed-policy",
"PolicyId": "ANPAUAWOPGE5LUNL2W12P",
"Arn": "arn:aws:iam::276384657722:policy/ad-customer-managed-policy",
"PolicyName": "AWSLambdaBasicExecutionRole-799421eb-f9f8-4297-88a8-7529e717aefe",
"PolicyId": "ANPAUAWOPGE5N36IJ7YAV",
"Arn": "arn:aws:iam::276384657722:policy/service-role/AWSLambdaBasicExecutionRole-799421eb-f9f8-4297-88a8-7529e717aefe",
"PolicyName": "AWSLambdaBasicExecutionRole-d0a7f065-308f-45e3-bca3-ba8db70082a8",
"PolicyId": "ANPAUAWOPGE5NKA30YAPY",
"Arn": "arn:aws:iam::276384657722:policy/service-role/AWSLambdaBasicExecutionRole-d0a7f065-308f-45e3-bca3-ba8db70082a8",

```

The command “aws iam list-policies --scope Local” will return the customer managed policies. “grep -A2 PolicyName” is used to search for the PolicyName string and the next two lines after the string match.

Step 15: Check for policy details of ad-customer-managed-policy.

Command: aws iam get-policy --policy-arn
arn:aws:iam::276384657722:policy/ad-customer-managed-policy

```

$ root@Kali ~ ➔ aws iam get-policy --policy-arn arn:aws:iam::276384657722:policy/ad-customer-managed-policy
{
    "Policy": {
        "PolicyName": "ad-customer-managed-policy",
        "PolicyId": "ANPAUAWOPGE5LUNL2WI2P",
        "Arn": "arn:aws:iam::276384657722:policy/ad-customer-managed-policy",
        "Path": "/",
        "DefaultVersionId": "v1",
        "AttachmentCount": 0,
        "PermissionsBoundaryUsageCount": 0,
        "IsAttachable": true,
        "CreateDate": "2021-01-20T07:51:42Z",
        "UpdateDate": "2021-01-20T07:51:42Z"
    }
}

```

Access Credentials to your AWS IaC

Step 16: Get the policy version document to check permissions that the policy grants.

Command: aws iam get-policy-version --policy-arn
arn:aws:iam::276384657722:policy/ad-customer-managed-policy --version-id v1

```

$ root@Kali ~ ➔ aws iam get-policy-version --policy-arn arn:aws:iam::276384657722:policy/ad-customer-managed-policy --version-id v1
{
    "PolicyVersion": {
        "Document": {
            "Version": "2012-10-17",
            "Statement": [
                {
                    "Sid": "VisualEditor0",
                    "Effect": "Allow",
                    "Action": [
                        "s3:GetObjectAcl",
                        "s3:GetObject",
                        "s3:GetBucketPolicy",
                        "s3:GetObjectVersion"
                    ],
                    "Resource": [
                        "arn:aws:s3:::file-uploader-saved-files",
                        "arn:aws:s3:::file-uploader-saved-files/*"
                    ]
                }
            ],
            "VersionId": "v1",
            "IsDefaultVersion": true,
            "CreateDate": "2021-01-20T07:51:42Z"
        }
    }
}

```

Login URL	https://276384657722.sigin.aws.amazon.com/console
Region	Asia Pacific (Singapore) ap-southeast-1
Username	qIVMHVAcHfwkveFcsASM
Password	Um7uYTtIKeMJU35Jc
Access Key ID	AKIAUAWOPGE5KB43QAEB
Secret Access Key	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

Step 17: Enumerate roles on the AWS account.

Command: aws iam list-roles

```

{
    "Path": "/service-role/",
    "RoleName": "xxe-demo-role-8u65x4rn",
    "RoleId": "AROAUAWOPGE5MA2YQRFXB",
    "Arn": "arn:aws:iam::276384657722:role/service-role/xxe-demo-role-8u65x4rn",
    "CreateDate": "2020-12-12T13:50:11Z",
    "AssumeRolePolicyDocument": {
        "Version": "2012-10-17",
        "Statement": [
            {
                "Effect": "Allow",
                "Principal": {
                    "Service": "lambda.amazonaws.com"
                },
                "Action": "sts:AssumeRole"
            }
        ],
        "MaxSessionDuration": 3600
    }
}

```

root@Kali ➜ ~

Step 18: Check details for roles.

Command: aws iam get-role --role-name ad-loggingrole

```

{
    "Role": {
        "Path": "/",
        "RoleName": "ad-LoggingRole",
        "RoleId": "AROAUAWOPGE5JQT23CRUN",
        "Arn": "arn:aws:iam::276384657722:role/ad-LoggingRole",
        "CreateDate": "2021-01-20T06:15:40Z",
        "AssumeRolePolicyDocument": {
            "Version": "2012-10-17",
            "Statement": [
                {
                    "Effect": "Allow",
                    "Principal": {
                        "AWS": "*"
                    },
                    "Action": "sts:AssumeRole"
                }
            ]
        },
        "MaxSessionDuration": 3600,
        "RoleLastUsed": {
            "LastUsedDate": "2021-02-12T15:03:54Z",
            "Region": "us-east-1"
        }
    }
}

```

root@Kali ➜ ~

Access Credentials to your AWS Account

Access Key ID	AKIAUAWOPGE5KB43QAEB
Secret Access Key	w1OqQUFmc4uQaeJ4iHresM78bTFimXboQp
Username	qIVMHVachHfwkvnFcsASM
Password	Um7uYTlKeMJU35Jc

Note: This role can be assumed by any AWS resource.

Step 19: Check for policies attached to roles.

Commands:

```
aws iam list-attached-role-policies --role-name ad-loggingrole  
aws iam list-role-policies --role-name ad-loggingrole
```

```
x $ root@Kali ➤ aws iam list-role-policies --role-name ad-loggingrole  
{  
    "PolicyNames": []  
}  
$ root@Kali ➤ aws iam list-attached-role-policies --role-name ad-loggingrole  
{  
    "AttachedPolicies": [  
        {  
            "PolicyName": "AmazonS3ReadOnlyAccess",  
            "PolicyArn": "arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess"  
        },  
        {  
            "PolicyName": "IAMReadOnlyAccess",  
            "PolicyArn": "arn:aws:iam::aws:policy/IAMReadOnlyAccess"  
        }  
    ]  
}
```

References:

1. AWS CLI (<https://docs.aws.amazon.com/cli/latest/reference/>)