

AWS Cloud Console Configuration & Hardening Project.

Project Overview

This project demonstrates how to securely configure and harden the AWS Management Console using IAM best practices. The goal is to reduce unauthorized access, apply least privilege permissions, and validate access control through testing.

Key security concepts covered:

- AWS Management Console Root Account set up (secured)
- Setting up 2 EC2 instances
- Creating an IAM Policy
- Creating an AW Alias
- Creating IAM group & User
- Test the IAM user access

Setting Up the AWS Management Console

I went on <https://aws.amazon.com/> to create an account. I signed up as a new user. I completed the identity and billing set up.



The image shows the AWS sign-up page. At the top right, it says "Sign up for AWS". Below that, there's a field for "Root user email address" with a note about account recovery and a link to the "AWS Privacy Notice". There's also a field for "AWS account name" with a note about changing it later. A large orange button labeled "Verify email address" is present. Below these fields, there's a section for "Sign in to an existing AWS account". At the bottom, a small note mentions essential cookies and a link to the "Cookie Notice". On the left side of the form, there's a section titled "Try AWS at no cost for up to 6 months" with a note about starting with \$100 in credits and earning more. An illustration of a rocket launching is shown next to this text.

I logged in as a root user and was immediately prompted to add a Multi Factory authentication which I did.

Select MFA device Info

MFA device name

Device name
This name will be used within the identifying ARN for this device.
 Maximum 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , . @ _ - (hyphen)

MFA device

Device options
In addition to username and password, you will use this device to authenticate into your account.

- Passkey or security key**
Authenticate using your fingerprint, face, or screen lock. Create a passkey on this device or use another device, like a FIDO2 security key.
- Authenticator app**
Authenticate using a code generated by an app installed on your mobile device or computer.
- Hardware TOTP token**
Authenticate using a code generated by Hardware TOTP token or other hardware devices.

Set up device Info

Authenticator app
A virtual MFA device is an application running on your device that you can configure by scanning a QR code.

- 1** Install a compatible application such as Google Authenticator, Duo Mobile, or Authy app on your mobile device or computer.
[See a list of compatible applications ↗](#)
- 2**  Open your authenticator app, choose **Show QR code** on this page, then use the app to scan the code. Alternatively, you can type a secret key. [Show secret key](#)
- 3** Type two consecutive MFA codes below
Enter a code from your virtual app below

Wait 30 seconds, and enter a second code entry.

[Cancel](#) [Previous](#) [Add MFA](#)

Multi Factory authentication was activated.

Multi-factor authentication (MFA) (1)			
Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. Learn more ↗			
Type	Identifier	Certifications	Created on
<input type="radio"/> Virtual	arn:aws:iam::950298808068:mfa/Afeezezg8	Not Applicable	Mon Jan 12 2026

Setting up 2 EC2 instances

In order to be able to create the 2 EC2 (Audit & Sales) I clicked on the search bar and typed EC2. I selected the first one that says “Virtual Servers in the cloud”.

The screenshot shows the AWS Services page with a search bar at the top containing "ec2". The main content area is titled "Services" and features three cards: "EC2 Virtual Servers in the Cloud", "EC2 Image Builder A managed service to automate build, customize and deploy OS images", and "Recycle Bin Protect resources from accidental deletion". On the left, there is a sidebar with links to "Services", "Features", "Resources", "Documentation", "Knowledge articles", "Marketplace", "Blog posts", "Events", and "Tutorials".

I clicked on “Launch Instances”

The screenshot shows the "Launch instance" page. It has a heading "Launch instance" and a sub-instruction "To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud." Below this are two buttons: "Launch instance" (highlighted in orange) and "Migrate a server". A note at the bottom states: "Note: Your instances will launch in the United States (N. Virginia) Region".

I input the name and added a new tag to indicate which environment it belongs, which is the Audit environment.

The screenshot shows the "Launch an instance" - "Name and tags" section. It displays two sets of tags: one for "Name" with key "Name" and value "Cybertech-Audit", and another for "Environment" with key "Environment" and value "Audit". Both tags have "Instances" buttons next to them. At the bottom, there is a link "Add new tag" and a note "You can add up to 48 more tags."

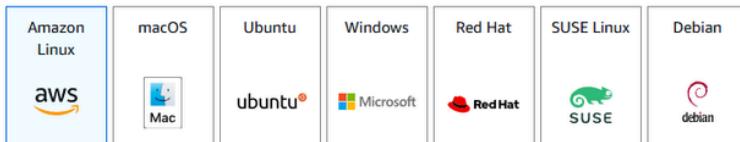
The next step is the selection of the AMI which is the operating system I want the server to run on. I chose the Amazon Linux AMI.

▼ Application and OS Images (Amazon Machine Image) [Info](#)

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose [Browse more AMIs](#).

Search our full catalog including 1000s of application and OS images

Quick Start



[Browse more AMIs](#)

Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)

Amazon Linux 2023 kernel-6.1 AMI
ami-07ff62358b87c7116 (64-bit (x86), uefi-preferred) / ami-059afa9e3a9c7af0c (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs

Free tier eligible

Description

Amazon Linux 2023 (kernel-6.1) is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.

Amazon Linux 2023 AMI 2023.10.20260105.0 x86_64 HVM kernel-6.1

Architecture	Boot mode	AMI ID	Publish Date	Username	(i)
64-bit (x86) ▾	uefi-preferred	ami-07ff62358b87c7116	2026-01-02	ec2-user	Verified provider

I moved on to the instance type which refers to the number of core and Ram I want to assign to it. I selected t3.micro which can run 2vCPU and has 1GB of memory.

▼ Instance type [Info](#) | [Get advice](#)

Instance type

t3.micro Free tier eligible
Family: t3 2 vCPU 1 GiB Memory Current generation: true
On-Demand Ubuntu Pro base pricing: 0.0139 USD per Hour On-Demand SUSE base pricing: 0.0104 USD per Hour
On-Demand Linux base pricing: 0.0104 USD per Hour On-Demand RHEL base pricing: 0.0392 USD per Hour
On-Demand Windows base pricing: 0.0196 USD per Hour

All generations

[Compare instance types](#)

[Additional costs apply for AMIs with pre-installed software](#)

I then launch my instances and created the second one following the same steps and settings and my two EC2 has been created.

[EC2](#) > Instances

Instances (2) Info											
<input type="text"/> Find Instance by attribute or tag (case-sensitive)											
	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	
<input type="checkbox"/>	Cybertech-Audit	i-03c6bb9233f4075ee	Running View alarms +	t3.micro	3/3 checks passed	View alarms +	us-east-1c	ec2-54-196-148-24.co...	54.196.148.24	-	
<input type="checkbox"/>	Cybertech-Sales	i-0bdd2e5de5fc8b6e9	Running View alarms +	t3.micro	Initializing	View alarms +	us-east-1c	ec2-54-221-188-144.co...	54.221.188.144	-	

Creating an IAM Policy

Moving on in the cloud security project, I want to create a policy.

Policies are role which tells who has access to what on our cloud infrastructure.

To do that, I will search for IAM in the search bar and select the first option.

The screenshot shows the AWS search interface with the search bar containing 'iam'. The results section is titled 'Services' and lists three options: 'IAM' (selected), 'IAM Identity Center', and 'Resource Access Manager'. The IAM card includes the description 'Manage access to AWS resources'.

I navigate to policy and I click on “create policy” in order to create my own policy.

The screenshot shows the 'Policies' list page with 1440 policies. A search bar and filter by type ('All types') are at the top. The table lists policies like 'AccessAnalyzerServiceRolePolicy', 'AccountManagementFromVercel', and 'AdministratorAccess'. A 'Create policy' button is visible in the top right.

I added the policy I created and gave it a name and clicked on “create policy”.

The screenshot shows the 'Review and create' step of the 'Create policy' wizard. It includes fields for 'Policy name' (set to 'Cybertech-AuditEnvPolicy'), 'Description - optional' (set to 'IAM policy for user in the audit environment'), and a 'Permissions defined in this policy' section. The permissions table shows an explicit deny for EC2 and an allow for EC2 with a condition. A green success message at the bottom states 'Policy Cybertech-AuditEnvPolicy created.'

The screenshot shows the 'Policies' list page again, but now with a green success message at the bottom stating 'Policy Cybertech-AuditEnvPolicy created.' The policy name 'Cybertech-AuditEnvPolicy' is visible in the list.

The screenshot shows the AWS IAM Policies page. The left sidebar has 'Identity and Access Management (IAM)' selected under 'Access Management'. The main area displays a table of policies. A search bar at the top right contains the text 'cyber'. The table has columns for 'Policy name', 'Type', 'Used as', and 'Description'. One row is visible, showing 'Cyber-AuditEnvPolicy' as a 'Customer managed' policy with no usage and a description starting with 'IAM policy for user in the audit environment...'. There are buttons for 'Actions', 'Delete', and 'Create policy'.

Creating an AW Alias

As I am currently logged in as a root user, I want to make logging for users to be easy, so I will create an alias login for them.

I will click on the search bar and search for IAM, and click on “create” under account alias. I gave it a name and clicked on create alias.

AWS Account

Account ID

950298808068

Account Alias

[Create](#)

The dialog box is titled 'Create alias for AWS account 950298808068'. It has a 'Preferred alias' field containing 'cybertechusers', with a note below stating 'Must be not more than 63 characters. Valid characters are a-z, 0-9, and - (hyphen)'. Below this is a 'New sign-in URL' field with the value 'https://cybertechusers.signin.aws.amazon.com/console'. A blue callout box contains the note: 'IAM users will still be able to use the default URL containing the AWS account ID.' At the bottom are 'Cancel' and 'Create alias' buttons.

Creating IAM group & User

The next step I want to do is to create the groups and users.

A group consist of different IAM users. The importance of creating a group is to make it easy when we want to apply a policy, it will affect all users in that group. Which makes it easier to assign policy to users.

IAM user is a member of a group.

To create a group, I search for IAM in the search bar and I clicked on “User group”, then I clicked on create group.

This screenshot shows the 'Create user group' page in the AWS IAM console. The left sidebar shows the IAM navigation menu with 'User groups' selected. The main form has a 'Name the group' section where 'CyberTech-Audit-Group' is typed into the input field. Below it is an 'Add users to the group - Optional' section which is currently empty. At the bottom is an 'Attach permissions policies - Optional' section where the 'cyber' policy is selected from a dropdown. A large orange 'Create user group' button is at the bottom right.

I gave my group a name and I attached the policy I previously created and I clicked on “create user group”.

This screenshot shows the 'Create user group' page after the group has been created. The 'Name the group' field now contains 'CyberTech-Audit-Group'. The 'Add users to the group - Optional' section is still empty. The 'Attach permissions policies - Optional' section shows the 'cyber' policy selected. The orange 'Create user group' button is visible at the bottom right.

The screenshot shows the AWS IAM User groups page. A green banner at the top indicates "CyberTech-Audit-Group user group created." The main table lists one user group:

Group name	Users	Permissions	Creation time
CyberTech-Audit-Group	0	Defined	Now

Navigation and search bars are visible on the left and top right respectively.

The next thing I want to do is to create user. In the same IAM tab, I clicked on “users”, then i clicked on create user.

The screenshot shows the AWS IAM Users page with a green banner indicating "CyberTech-Audit-Group user group created." Below it, the "Create user" step 1 details page is shown. The steps are:

- Step 1: Specify user details (selected)
- Step 2: Set permissions
- Step 3: Review and create

The "Specify user details" section includes fields for "User name" and "Provide user access to the AWS Management Console - optional". A note states: "In addition to console access, users with SignInLocalDevelopmentAccess permissions can use the same console credentials for programmatic access without the need for access keys." A link "Learn more" is provided.

Buttons for "Cancel" and "Next" are at the bottom right.

I added the user to the group I previously created.

The screenshot shows the AWS IAM Set permissions step. The steps are:

- Step 1: Specify user details
- Step 2: Set permissions (selected)
- Step 3: Review and create
- Step 4: Retrieve password

The "Set permissions" section includes a note: "Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions." A link "Learn more" is provided. The "Permissions options" section has three choices:

- Add user to group: Selected. Description: "Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function."
- Copy permissions: Description: "Copy all group memberships, attached managed policies, and inline policies from an existing user."
- Attach policies directly: Description: "Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group."

The "User groups (1/1)" table shows one group assigned:

Group name	Users	Attached policies	Created
CyberTech-Audit-Group	0	CyberTech-AuditEnvPolicy	2026-01-14 (10 minutes ago)

User created successfully and added to a group.

Screenshot of the AWS IAM Users page showing a successful user creation.

User created successfully

You can view and download the user's password and email instructions for signing in to the AWS Management Console.

View user

Users (1) Info

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

User name	Path	Group	Last activity	MFA	Password age	Console last sign-in	Access key ID	Active key age	Access key status
Cybertech-Audit-Ali	/	1	-	-	1 minute	-	-	-	-

Actions: Copy | Delete | Create user

I signed in as the new user created.

Screenshot of the AWS IAM user sign-in page.

IAM user sign in Info

Account ID or alias (Don't have?)

Remember this account

IAM username

Password

Show Password Having trouble?

Sign in

[Sign in using root user email](#)

[Create a new AWS account](#)

By continuing, you agree to [AWS Customer Agreement](#) or other agreement for AWS services, and the [Privacy Notice](#). This site uses essential cookies. See our [Cookie Notice](#) for more information.

aws

Run models at scale with cost-effective chips

Our new AI chip, EC2 Trn3 UltraServer, delivers the best token economics for AI applications.

Explore Trn3 >

Screenshot of the AWS CloudWatch Metrics console showing the "Console Home" dashboard. The dashboard includes sections for "Recently visited" services (EC2, S3, Aurora and RDS, Lambda), "Welcome to AWS" (Getting started with AWS, Training and certification), "AWS Health" (No health data), "Applications" (0), and "Cost and usage". A red box highlights an error message: "Access denied to servicecatalog>ListApplications" with a "Diagnose with Amazon Q" button.

Test the IAM user access.

Finally, I want to test if the policy works.

After signing in as the user, we can see the policy works due to the “access denied” message it displays.

Screenshot of the AWS CloudWatch Metrics console showing the "Console Home" dashboard. The dashboard includes sections for "Recently visited" services (EC2, S3, Aurora and RDS, Lambda), "Welcome to AWS" (Getting started with AWS, Training and certification), "AWS Health" (No health data), "Applications" (0), and "Cost and usage". A red box highlights an error message: "Access denied to servicecatalog>ListApplications" with a "Diagnose with Amazon Q" button.

Security recommendations 0

Access denied to iam>ListMFADevices

You don't have permission to `iam>ListMFADevices`. To request access, copy the following text and send it to your AWS administrator. [Learn more about troubleshooting access denied errors.](#)

User: arn:aws:iam::950298808068:user/Cybertech-Audit-Ali
Action: iam>ListMFADevices
Context: no identity-based policy allows the action

[Diagnose with Amazon Q](#)

Access denied to iam>ListAccessKeys

You don't have permission to `iam>ListAccessKeys`. To request access, copy the following text and send it to your AWS administrator. [Learn more about troubleshooting access denied errors.](#)

User: arn:aws:iam::950298808068:user/Cybertech-Audit-Ali
Action: iam>ListAccessKeys
Context: no identity-based policy allows the action

[Diagnose with Amazon Q](#)

IAM resources

Resources in this AWS Account

Access denied to iam:GetAccountSummary

You don't have permission to `iam:GetAccountSummary`. To request access, copy the following text and send it to your AWS administrator. [Learn more about troubleshooting access denied errors.](#)

User: arn:aws:iam::950298808068:user/Cybertech-Audit-Ali
Action: iam:GetAccountSummary
Context: no identity-based policy allows the action

AWS Account

Access denied to iam>ListAccountAliases

You don't have permission to `iam>ListAccountAliases`. To request access, copy the following text and send it to your AWS administrator. [Learn more about troubleshooting access denied errors.](#)

User: arn:aws:iam::950298808068:user/Cybertech-Audit-Ali
Action: iam>ListAccountAliases
Context: no identity-based policy allows the action

[Diagnose with Amazon Q](#)

Quick Links

[My security credentials](#)

Manage your access keys, multi-factor authentication (MFA) and other credentials.

Tools

[Policy simulator](#)

The simulator evaluates the policies that you choose and determines the effective permissions for each of the actions that you specify.

[Additional documentation](#)