

# EC2

The screenshot shows the 'Name and tags' step of the 'Launch an instance' wizard. The 'Name' field contains 'dotnet server'. Below it is a search bar for 'Application and OS Images (Amazon Machine Image)'. A 'Quick Start' section includes tabs for Amazon, macOS, Ubuntu, Windows, Red Hat, and SUSE Linux.

The screenshot shows the success message 'Successfully initiated launch of instance (i-01cb528858ec93dd2)' after launching the instance. It also shows the 'Next Steps' section with links to 'Create billing and free tier usage alerts', 'Connect to your instance', 'Connect an RDS database', and 'Create EBS snapshot policy'.

The screenshot shows the AWS EC2 Instances page. The left sidebar includes options like EC2 Dashboard, EC2 Global View, Events, Console-to-Code (Preview), Instances (selected), Instances Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations (New), and Images (AMIs, AMI Catalog). The main content area displays 'Instances (1/2) Info' with a search bar and filters for Name, Instance ID, Instance state, Instance type, Status check, and Alarm status. Two instances are listed: 'dotnet server' (Instance ID i-01cb528858ec93dd2, State: Running, Type: t2.micro) and 'dotnet server' (Instance ID i-0f3342efe99c73c81, State: Terminated, Type: t2.micro). Below this is the 'Instance: i-01cb528858ec93dd2 (dotnet server)' details page, which includes tabs for Details, Status and alarms (New), Monitoring, Security, Networking, Storage, and Tags. The instance summary shows Public IPv4 address 3.85.107.66, Private IPv4 address 172.31.81.59, and Public IPv4 DNS ec2-3-85-107-66.compute-1.amazonaws.com. The status bar at the bottom indicates the date as 18-01-2024.

This screenshot is similar to the previous one but shows a different state for the first instance. The 'dotnet server' instance (Instance ID i-01cb528858ec93dd2) is now shown as 'Shutting-d...' (State: Shutting-down, Type: t2.micro). The second instance (Instance ID i-0f3342efe99c73c81) remains 'Terminated'. The rest of the interface and status bar are identical to the first screenshot.

S3

The screenshot shows the 'General configuration' step of creating an S3 bucket. The 'AWS Region' is set to 'US East (N. Virginia) us-east-1'. The 'Bucket type' dropdown is open, showing two options: 'General purpose' (selected) and 'Directory - New'. The 'Bucket name' field contains 'nethrabucket03'. Below it, a note states: 'Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming'. The 'Copy settings from existing bucket - optional' section is present but empty. The status bar at the bottom shows 'CloudShell Feedback' and the date '18-01-2024'.

The screenshot shows the continuation of the bucket creation process for the 'ap-south-1' region. The 'Object Ownership' section is displayed, with 'ACLs disabled (recommended)' selected. The status bar at the bottom shows 'CloudShell Feedback' and the date '18-01-2024'.

Bucket Versioning

Disable

Enable

**Tags - optional (1)**

You can use bucket tags to track storage costs and organize buckets. [Learn more](#)

Key Value - optional

Nethra TN [Remove](#)

[Add tag](#)

**Default encryption** [Info](#)

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type [Info](#)

Server-side encryption with Amazon S3 managed keys (SSE-S3)

Server-side encryption with AWS Key Management Service keys (SSE-KMS)

Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)

Secure your objects with two separate layers of encryption. For details on pricing, see [DSSE-KMS pricing](#) on the Storage tab of the [Amazon S3 pricing page](#).

CloudShell Feedback Type here to search

Encryption type [Info](#)

Server-side encryption with Amazon S3 managed keys (SSE-S3)

Server-side encryption with AWS Key Management Service keys (SSE-KMS)

Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)

Secure your objects with two separate layers of encryption. For details on pricing, see [DSSE-KMS pricing](#) on the Storage tab of the [Amazon S3 pricing page](#).

Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#)

Disable

Enable

**Advanced settings**

After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.

Cancel [Create bucket](#)

CloudShell Feedback Type here to search

The screenshot shows the AWS S3 console interface. At the top, a green banner indicates that a bucket has been successfully created. Below this, the 'Account snapshot' section provides an overview of storage usage and activity trends. The main area displays 'General purpose buckets' with one entry: 'nethrabit03'. This entry includes details such as the AWS Region (Asia Pacific (Mumbai) ap-south-1), Access level (Objects can be public), and Creation date (January 18, 2024, 22:27:31 (UTC+05:30)).

Name	AWS Region	Access	Creation date
nethrabit03	Asia Pacific (Mumbai) ap-south-1	Objects can be public	January 18, 2024, 22:27:31 (UTC+05:30)

At the bottom of the page, there are links for CloudShell, Feedback, and various system status indicators.

The screenshot shows the 'Upload objects - S3 bucket nethrabit03' page. It displays a large dashed box for dragging and dropping files. Below this, a table lists the uploaded file 'Task6.pdf' with its type identified as 'application/pdf'. The 'Destination' field is set to 's3://nethrabit03'. The 'Destination details' section is expanded, showing the destination path.

Name	Type
Task6.pdf	application/pdf

At the bottom of the page, there are links for CloudShell, Feedback, and various system status indicators.

The screenshot shows two consecutive screenshots of the AWS S3 console interface.

**Screenshot 1: Upload objects - S3 bucket nethrabucket03**

The top bar shows tabs for "Sign in", "Task/Task at master - Nethrasree", "Registration", "aws console login - Search", and "Upload objects - S3 bucket nethrabucket03". The main content area displays a green success message: "Upload succeeded" with "View details below." Below this is a "Summary" table:

Destination	Succeeded	Failed
s3://nethrabucket03	1 file, 933.1 KB (100.00%)	0 files, 0 B (0%)

Below the summary is a "Files and folders" section:

**Files and folders (1 Total, 933.1 KB)**

Name	Folder	Type	Size	Status	Error
Task6.pdf	-	application/...	933.1 KB	Succeeded	-

**Screenshot 2: Delete objects - S3 bucket nethrabucket03**

The top bar shows tabs for "Sign in", "Task/Task at master - Nethrasree", "Registration", "aws console login - Search", and "Delete objects - S3 bucket nethrabucket03". The main content area displays a green success message: "Successfully deleted objects" with "View details below." Below this is a table:

Source	Successfully deleted	Failed to delete
s3://nethrabucket03	1 object, 933.1 KB	0 objects

Below the table is a "Failed to delete" section:

**Failed to delete (0)**

Name	Folder	Type	Last modified	Size	Error
No objects failed to delete.					

The bottom of each screenshot shows a Windows taskbar with various pinned icons and system status information.

Successfully emptied bucket "nethrabucket03"  
View details below. If you want to delete this bucket, use the [delete bucket configuration](#).

The details below are no longer available after you navigate away from this page.

Source	Successfully deleted	Failed to delete
s3://nethrabucket03	2 objects, 933.1 KB	0 objects

**Failed to delete (0)**

Name	Prefix	Version ID	Type	Last modified	Size	Error
No failed object deletions						

CloudShell Feedback © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Successfully deleted bucket "nethrabucket03"

**Create a bucket**

Every object in S3 is stored in a bucket. To upload files and folders to S3, you'll need to create a bucket where the objects will be stored.

[Create bucket](#)

**Pricing**

With S3, there are no minimum fees. You only pay for what you use. Prices are based on the location of your S3 bucket.

Waiting for ap-south-1.prod.pr.analytics.console.aws.a2z.com... © 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

## RDS

The screenshot shows the AWS RDS console for creating a new database instance. The 'Easy create' option is selected, which provides recommended best-practice configurations. The configuration section includes options for engine type (Aurora MySQL-Compatible, Aurora PostgreSQL-Compatible, MySQL, MariaDB, PostgreSQL, Oracle) and character set. On the right, the 'Aurora MySQL-Compatible Edition' details page is displayed, highlighting its enterprise-class features like up to 128 TB of storage and six-way replication across three Availability Zones.

This screenshot shows the continuation of the RDS instance creation process, specifically for the MySQL engine. It requires a master password (Nethra0307) and a confirmation. A note about setting up an EC2 connection is present. The MySQL details page on the right highlights its popularity and rich features, including support for various instance classes and automated backup.

Sign in

Task/Task at master - Nethrasree | Registration | aws console login - Search | Databases | RDS | us-east-1 | +

Aws Services Search [Alt+S]

Amazon RDS

Introducing Aurora I/O-Optimized

Aurora's I/O-Optimized is a new cluster storage configuration that offers predictable pricing for all applications and improved price-performance, with up to 40% costs savings for I/O-intensive applications.

RDS > Databases

Consider creating a Blue/Green Deployment to minimize downtime during upgrades

You may want to consider using Amazon RDS Blue/Green Deployments and minimize your downtime during upgrades. A Blue/Green Deployment provides a staging environment for changes to production databases.

RDS User Guide | Aurora User Guide

Databases (1)

Group resources Modify Actions Restore from S3 Create database

Filter by databases

DB identifier	Status	Role	Engine	Region & AZ	Size
nethradb	Creating	Instance	MySQL Community	-	db.t3.micro

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Type here to search

25°C 22:35 ENG 18-01-2024

Sign in

Task/Task at master - Nethrasree | Registration | aws console login - Search | RDS | us-east-1 | +

Aws Services Search [Alt+S]

Amazon RDS

Endpoint & port	Networking	Security
Endpoint	Availability Zone	VPC security groups default (sg-03864c4cc07dd8c0d) Active
Port	VPC vpc-099f09b82b6b7b968	Publicly accessible No
	Subnet group default-vpc-099f09b82b6b7b968	Certificate authority Info rds-ca-2019
	Subnets subnet-0c60fee0a160d72d7 subnet-026127f9f561c82bf subnet-06f2b081693579b5a subnet-097db471abc27a701 subnet-0hef0r4f0659971c1	Certificate authority date August 22, 2024, 22:38 (UTC+05:30)

CloudShell Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Type here to search

25°C 22:35 ENG 18-01-2024

The screenshot shows the AWS RDS console with the URL <https://us-east-1.console.aws.amazon.com/rds/home?region=us-east-1#databases>. A modal window titled "Introducing Aurora I/O-Optimized" is displayed, stating: "Aurora's I/O-Optimized is a new cluster storage configuration that offers predictable pricing for all applications and improved price-performance, with up to 40% costs savings for I/O-intensive applications." Below it, another modal window titled "Consider creating a Blue/Green Deployment to minimize downtime during upgrades" provides information about minimizing downtime during database upgrades.

The main RDS interface shows a table titled "Databases (1)". The single entry is "nethradb", which is marked as "Deleting". The table includes columns for DB identifier, Status, Role, Engine, Region & AZ, and Size. The status is listed as "Deleting".

## IAM

The screenshot shows the AWS IAM console with the URL <https://us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/groups/create>. The left sidebar shows the "Identity and Access Management (IAM)" menu with "User groups" selected. The main area is titled "Create user group" and contains two sections: "Name the group" and "Add users to the group - Optional (0) Info".

In the "Name the group" section, the "User group name" field is filled with "Nethra@0307".

In the "Add users to the group" section, there is a search bar and a table with columns for "User name" (empty), "Groups" (empty), "Last activity" (empty), and "Creation time" (empty).

The screenshot shows the AWS Identity and Access Management (IAM) console. The left sidebar is titled "Identity and Access Management (IAM)" and includes sections for Dashboard, Access management (User groups, Users, Roles, Policies, Identity providers, Account settings), Access reports (Access Analyzer, External access), and Global navigation (Global, Nethrasree). A search bar at the top right contains the query "EC2". Below it, a table lists 30 matches for "Policy name". The table columns are Policy name, Type, Used as, and Description. One policy, "AmazonEC2FullAcc...", has a checked checkbox next to it.

Policy name	Type	Used as	Description
AmazonEC2Contai...	AWS managed	None	Provides administrative access to Amazon EC2
AmazonEC2Contai...	AWS managed	None	Provides full access to Amazon EC2
AmazonEC2Contai...	AWS managed	None	Provides read-only access to Amazon EC2
AmazonEC2Contai...	AWS managed	None	Policy to enable Task AutoScaling
AmazonEC2Contai...	AWS managed	None	Policy to enable CloudWatch Events
AmazonEC2Contai...	AWS managed	None	Default policy for the Amazon ECS
AmazonEC2Contai...	AWS managed	None	Default policy for Amazon EC2
AmazonEC2Contai...	AWS managed	None	Provides full access to Amazon EC2
AmazonEC2ReadO...	AWS managed	None	Provides read only access to Amazon EC2
AmazonEC2Rolefo...	AWS managed	None	Provides EC2 access to S3 buckets
AmazonEC2Rolefo...	AWS managed	None	Provides EC2 limited access to S3 buckets

The screenshot shows the AWS IAM User groups page. The left sidebar is identical to the previous one. A green banner at the top center says "Nethra@0307 user group created." Below it, the "User groups (1) Info" section is displayed. It states: "A user group is a collection of IAM users. Use groups to specify permissions for a collection of users." A table lists the single user group "Nethra@0307". The table columns are Group name, Users, Permissions, and Creation time. The "Users" column shows a warning icon and "0" users assigned.

Group name	Users	Permissions	Creation time
Nethra@0307	0	Defined	Now

The screenshot shows the 'Specify user details' step of the AWS IAM 'Create user' wizard. The user name is set to 'Nethra@0307'. A note indicates that the user name can have up to 64 characters and must consist of valid characters (A-Z, a-z, 0-9, and + = . @ \_ - (hyphen)). An optional checkbox for providing access to the AWS Management Console is present. A callout box provides instructions for generating programmatic access keys or service-specific credentials for AWS CodeCommit or Amazon Keypairs.

User details

User name  
Nethra@0307

The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = . @ \_ - (hyphen)

Provide user access to the AWS Management Console - *optional*  
If you're providing console access to a person, it's a [best practice](#) to manage their access in IAM Identity Center.

If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keypairs, you can generate them after you create this IAM user. [Learn more](#)

Cancel Next

**Review and create**

Add user to group (radio button selected): Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.

Copy permissions (radio button): Copy all group memberships, attached managed policies, and inline policies from an existing user.

Attach policies directly (radio button): 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.

User groups (1/1)

Group name	Attached policies	Created
Nethra	AmazonEC2FullAccess	2024-01-1...

Set permissions boundary - *optional*

Cancel Previous Next

The screenshot shows the AWS Identity and Access Management (IAM) console. A green success message at the top states: "User created successfully" and "You can view and download the user's password and email instructions for signing in to the AWS Management Console." Below this, the "Users" section is displayed with one user listed: "Nethra@0307". The left sidebar includes options for Dashboard, Access management (User groups, Roles, Policies, Identity providers, Account settings), and Access reports (Access Analyzer, External access). The bottom navigation bar shows CloudShell, Feedback, and various system icons.

The screenshot shows the "Create role" page in the AWS IAM console. A callout box at the top right says: "Create a custom trust policy to enable others to perform actions in this account." Below this, the "Use case" section is shown with the heading "Use case" and the sub-instruction "Allow an AWS service like EC2, Lambda, or others to perform actions in this account." A dropdown menu under "Service or use case" is set to "EC2". The "Use case" section contains several radio button options: "EC2" (selected), "EC2 Role for AWS Systems Manager", "EC2 Spot Fleet Role", "EC2 - Spot Fleet Auto Scaling", and "EC2 - Spot Fleet Tagging". The bottom navigation bar shows CloudShell, Feedback, and various system icons.

The screenshot shows the AWS IAM Roles page. A green success banner at the top states "Role Nethra@0307 created." Below it, a search bar and a table header with columns for "Role name" and "Trusted entities". The main content area is titled "Roles Anywhere" with a brief description: "Authenticate your non AWS workloads and securely provide access to AWS services." It features three cards: "Access AWS from your non AWS workloads" (X.509 Standard), "X.509 Standard" (use own PKI or AWS Certificate Manager Private Certificate Authority), and "Temporary credentials" (use temporary credentials with enhanced security). The left sidebar shows the IAM navigation menu.

The screenshot shows the AWS IAM Roles page after a role has been deleted. A green success banner at the top states "Role deleted Nethra@0307." Below it, a search bar and a table header with columns for "Role name" and "Trusted entities". The table lists one role: "rds-monitoring-role" (AWS Service: monitoring.rds). The main content area is titled "Roles Anywhere" with a brief description: "Authenticate your non AWS workloads and securely provide access to AWS services." It features three cards: "Access AWS from your non AWS workloads" (X.509 Standard), "X.509 Standard" (use own PKI or AWS Certificate Manager Private Certificate Authority), and "Temporary credentials" (use temporary credentials with enhanced security). The left sidebar shows the IAM navigation menu.

The screenshot shows the AWS Identity and Access Management (IAM) console. The left sidebar is collapsed, and the main area displays a green success message: "User 'Nethra@0307' deleted." Below this, the "Users" section is shown with a table header: "User (0) Info". The table has columns for "User name", "Path", "Group", "Last activity", and "MFA". A note states: "An IAM user is an identity with long-term credentials that is used to interact with AWS in an account." A search bar and a "Create user" button are also present.

This screenshot is similar to the first one but shows two consecutive deletions. The first message says "User 'Nethra@0307' deleted." and the second message says "User group deleted." The "User groups" section is displayed with a table header: "User groups (0) Info". The table has columns for "Group name", "Users", "Permissions", and "Creation time". A note states: "A user group is a collection of IAM users. Use groups to specify permissions for a collection of users." A search bar and a "Create group" button are also present.

SNS

New Feature  
Amazon SNS now supports in-place message archiving and replay for FIFO topics. [Learn more](#)

## Application Integration

# Amazon Simple Notification Service

## Pub/sub messaging for microservices and serverless applications.

Amazon SNS is a highly available, durable, secure, fully managed pub/sub messaging service that enables you to decouple microservices, distributed systems, and event-driven serverless applications. Amazon SNS provides topics for high-throughput, push-based, many-to-many messaging.

**Create topic**

**Topic name**  
A topic is a message channel. When you publish a message to a topic, it fans out the message to all subscribed endpoints.

**Next step**

[Start with an overview](#)

▶ **Delivery policy (HTTP/S) - optional** [Info](#)  
The policy defines how Amazon SNS retries failed deliveries to HTTP/S endpoints. To modify the default settings, expand this section.

▶ **Delivery status logging - optional** [Info](#)  
These settings configure the logging of message delivery status to CloudWatch Logs.

▶ **Tags - optional**  
A tag is a metadata label that you can assign to an Amazon SNS topic. Each tag consists of a key and an optional value. You can use tags to search and filter your topics and track your costs. [Learn more](#)

▶ **Active tracing - optional** [Info](#)  
Use AWS X-Ray active tracing for this topic to view its traces and service map in Amazon CloudWatch. Additional costs apply.

**Create topic**

The screenshot shows the AWS SNS console with a green success message: "Topic Developer created successfully. You can create subscriptions and send messages to them from this topic." Below this, the "Developer" topic details are displayed, including Name (Developer), ARN (arn:aws:sns:us-east-1:767398111163:Developer), and Type (Standard). There are "Edit", "Delete", and "Publish message" buttons.

The screenshot shows the "Create subscription" wizard. In the "Protocol" step, "Email" is selected as the endpoint type. The "Endpoint" field contains "nethrasree.logu@aspiresys.com". A note says "After your subscription is created, you must confirm it." In the "Subscription filter policy - optional" step, there is a note about filtering messages. In the "Redrive policy (dead-letter queue) - optional" step, there is a note about sending undeliverable messages to a dead-letter queue. At the bottom are "Cancel" and "Create subscription" buttons.

The screenshot shows the AWS SNS console with a green success message: "Subscription to Developer created successfully. The ARN of the subscription is arn:aws:sns:us-east-1:76739811163:Developer:69738af8-d971-462b-bf54-058ced657ff7." Below this, the "Subscription: 69738af8-d971-462b-bf54-058ced657ff7" page is displayed with "Edit" and "Delete" buttons. The "Details" section shows the ARN, Status (Pending confirmation), Protocol (EMAIL), Endpoint (nethrasree.logu@aspiresys.com), and Topic. The navigation bar includes "Amazon SNS > Topics > Developer > Subscription: 69738af8-d971-462b-bf54-058ced657ff7". The bottom status bar shows "CloudShell Feedback" and system information like "25°C Mostly sunny 09:40 19-01-2024".

The screenshot shows the "Message body" configuration page. It has two options for "Message structure": "Identical payload for all delivery protocols" (selected) and "Custom payload for each delivery protocol". Under "Message body to send to the endpoint", there is a text area containing the message: "1. Hi Nethra,  
2. Please send me your feedback." The bottom status bar shows "CloudShell Feedback" and system information like "25°C Mostly sunny 09:44 19-01-2024".

The screenshot shows the AWS SNS console. On the left, the navigation menu includes 'Amazon SNS', 'Dashboard', 'Topics' (which is selected), 'Subscriptions', and 'Mobile' (with options for Push notifications, Text messaging (SMS), and Origination numbers). The main content area displays a green success message: 'Message published to topic Developer successfully.' with details: Message ID: 42e487b9-80d1-5c28-a9d-d99a79b56e0c, Request ID: 2334fc4d-0027-5772-8705-a26ef9437556. Below this, the 'Developer' topic details are shown, including Name: Developer, ARN: arn:aws:sns:us-east-1:767398111163:Developer, Topic owner: 767398111163, and Type: Standard. Action buttons for 'Edit', 'Delete', and 'Publish message' are available. The bottom of the screen shows the Windows taskbar with various pinned icons.

The screenshot shows the AWS SNS 'Topics' page. The left sidebar has the same navigation as the previous screen. The main content area shows a green success message: 'Topic Developer deleted successfully.' Below this, the 'Topics (0)' section is displayed, featuring a search bar and a table header with columns for 'Name', 'Type', and 'ARN'. A message at the bottom says 'No topics' and 'To get started, create a topic.', with a 'Create topic' button. The bottom of the screen shows the Windows taskbar.

The screenshot shows the AWS SNS Subscriptions page. On the left, a sidebar lists 'Amazon SNS' services: Dashboard, Topics, Subscriptions (selected), and Mobile (Push notifications, Text messaging (SMS), Origination numbers). The main content area displays a green success message: 'Subscription deleted successfully.' Below this, the 'Subscriptions (0)' section includes buttons for Edit, Delete, Request confirmation, Confirm subscription, and Create subscription. A search bar and a table header ('ID', 'Endpoint', 'Status', 'Protocol', 'Topic') are also present. The status bar at the bottom indicates 'No subscriptions found' and a 'Create subscription' button.



