

Create a Secure EC2 Instance with S3 Access using IAM

1. S3 Console — List of Buckets

The screenshot shows the AWS S3 console interface. On the left, there's a sidebar with navigation links like 'General purpose buckets', 'Directory buckets', 'Table buckets', etc. The main area displays a table of 'General purpose buckets' with the following data:

Name	AWS Region	Creation date
bucket1-klh-a5	US East (N. Virginia) us-east-1	September 10, 2025, 13:06:24 (UTC+05:30)
bucket2-klh-a5-backup	US East (N. Virginia) us-east-1	September 10, 2025, 13:06:53 (UTC+05:30)
my-bucket1-2025	US East (N. Virginia) us-east-1	October 9, 2025, 13:10:31 (UTC+05:30)
registrationvar	US East (N. Virginia) us-east-1	September 24, 2025, 14:01:39 (UTC+05:30)
varshi-website	US East (N. Virginia) us-east-1	September 10, 2025, 13:23:41 (UTC+05:30)
varshitha-klh-cse-a5	Europe (Stockholm) eu-north-1	August 20, 2025, 13:18:34 (UTC+05:30)

On the right, there are two cards: 'Account snapshot' and 'External access'.

2. Create S3 Bucket — Enter bucket name (nida-lab-bucket)

The screenshot shows the 'Create bucket' wizard. It has several sections:

- General configuration**:
 - AWS Region**: Europe (Stockholm) eu-north-1
 - Bucket type**: General purpose (selected)
- Bucket name**: nida-lab-bucket
- Copy settings from existing bucket - optional**: Only the bucket settings in the following configuration are copied.
 - Choose bucket**: s3://bucket/prefix
- Object Ownership**

3. Bucket Created Successfully

The screenshot shows the AWS S3 Buckets page. At the top, a green banner displays the message "Successfully created bucket 'nida-lab-bucket'". Below the banner, there are two tabs: "General purpose buckets" (selected) and "Directory buckets". A search bar and a "Create bucket" button are also present. The main area lists "General purpose buckets (7)" with the following details:

Name	AWS Region	Creation date
bucket1-klh-a5	US East (N. Virginia) us-east-1	September 10, 2025, 13:06:24 (UTC+05:30)
bucket2-klh-a5-backup	US East (N. Virginia) us-east-1	September 10, 2025, 13:06:53 (UTC+05:30)
my-bucket1-2025	US East (N. Virginia) us-east-1	October 9, 2025, 13:10:31 (UTC+05:30)
nida-lab-bucket	Europe (Stockholm) eu-north-1	November 2, 2025, 11:54:02 (UTC+05:30)
registrationvar	US East (N. Virginia) us-east-1	September 24, 2025, 14:01:39 (UTC+05:30)
varshi-website	US East (N. Virginia) us-east-1	September 10, 2025, 13:23:41 (UTC+05:30)
varshitha-klh-cse-a5		August 20, 2025, 13:18:34 (UTC+05:30)

On the right side, there are two cards: "Account snapshot" and "External access summary - new".

4. Upload File to S3 — Add sample.rtf

The screenshot shows the AWS S3 Upload page for the "nida-lab-bucket". The URL in the address bar is "eu-north-1.console.aws.amazon.com/s3/upload/nida-lab-bucket?region=eu-north-1". The page has a header with the AWS logo, search bar, and account information. The main content area is titled "Upload" and contains the following sections:

- Upload**: A large input field with the placeholder "Drag and drop files and folders you want to upload here, or choose Add files or Add folder."
- Files and folders (1 total, 389.0 B)**: A table showing one file: "sample.rtf" (text/rtf, 389.0 B). There are buttons for "Remove", "Add files", and "Add folder".
- Destination**: Set to "s3://nida-lab-bucket".
- Destination details**: Describes bucket settings for new objects.
- Permissions**: A section for managing object permissions.

5. Upload Status — Upload succeeded

The screenshot shows the AWS S3 console interface. At the top, there is a green success message: "Upload succeeded. For more information, see the Files and folders table." Below this, a summary section indicates "Succeeded" (1 file, 389.0 B (100.00%)) and "Failed" (0 files, 0 B (0%)). The "Files and folders" tab is selected, showing a table with one item: "sample.rtf" (text/rtf, 389.0 B, Succeeded). A "Close" button is visible in the top right corner of the main content area.

6. IAM Console

The screenshot shows the AWS IAM console. The left sidebar includes sections for Identity and Access Management (IAM), Access management, and Access reports. The main area displays a list of roles with 11 items. The columns are Role name, Trusted entities, and Last activity. The "Create role" button is highlighted with a blue circle and cursor. A search bar is also present at the top of the list table.

Role name	Trusted entities	Last activity
AWSServiceRoleForAPIGateway	AWS Service: ops.apigateway (Service)	-
AWSServiceRoleForResourceExplorer	AWS Service: resource-explorer-2 (Service)	17 minutes ago
AWSServiceRoleForSupport	AWS Service: support (Service-Linked)	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service)	-
Ec2s3Access	AWS Service: ec2	-
registrationFunction-role-75zvckz3	AWS Service: lambda	-
registrationFunction-role-801u962l	AWS Service: lambda	38 days ago
registrationFunction-role-ljzkgygx	AWS Service: lambda	-
s3cr_rrole_for_bucket1-klh-a5	AWS Service: s3	-
s3cr_rrole_for_bucket1-klh-a5_1	AWS Service: s3	52 days ago

7. Create Role — Select trusted entity (AWS service → EC2)

The screenshot shows the AWS IAM 'Create role' wizard at Step 1: Select trusted entity. The 'Trusted entity type' section is displayed, showing five options:

- AWS service**: Allow AWS services like EC2, Lambda, or others to perform actions in this account.
- AWS account**: Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
- Web identity**: Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.
- SAML 2.0 federation**: Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.
- Custom trust policy**: Create a custom trust policy to enable others to perform actions in this account.

Below this, the 'Use case' section indicates that the role will be used for EC2. A dropdown menu shows 'EC2'.

8. Add Permissions — Attach AmazonS3FullAccess policy

The screenshot shows the AWS IAM 'Create role' wizard at Step 2: Add permissions. The 'Permissions policies' section is displayed, showing a list of policies:

- AmazonS3FullAccess**: AWS managed. Provides full access to all buckets via t...

Below this, a note says 'Set permissions boundary - optional'. At the bottom, there are 'Cancel', 'Previous', and 'Next' buttons.

9. Role Details — Enter role name and review trust policy

The screenshot shows the 'Create role' wizard in the AWS IAM console. The current step is 'Name, review, and create'. The 'Role name' field contains 'Awsec2s3'. The 'Description' field contains 'Allows EC2 Instances to call AWS services on your behalf.' A preview of the 'Trust policy' is shown as JSON code:

```
1 "Version": "2012-10-17",
2 "Statement": [
3     {
4         "Effect": "Allow",
5         "Action": [
6             "sts:AssumeRole"
7         ],
8         "Principal": [
9             "Service": [
10                 "ec2.amazonaws.com"
11             ]
12         ]
13     }
14 ]
```

10. Role Created — Confirmation in Roles list

The screenshot shows the 'Roles' list page in the AWS IAM console. The sidebar shows 'Identity and Access Management (IAM)'. The main area displays a green confirmation message: 'Role Awsec2s3 created.' Below it, the 'Roles (12)' list is shown. The table includes columns for 'Role name', 'Trusted entities', and 'Last activity'. The newly created role 'Awsec2s3' is listed with 'AWS Service: ec2' as the trusted entity and no last activity.

Role name	Trusted entities	Last activity
Awsec2s3	AWS Service: ec2	-
AWSServiceRoleForAPIGateway	AWS Service: ops.apigateway	-
AWSServiceRoleForResourceExplorer	AWS Service: resource-explorer-2	18 minutes ago
AWSServiceRoleForSupport	AWS Service: support	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor	-
Ec2s3Access	AWS Service: ec2	-
registrationFunction-role-75zvckz3	AWS Service: lambda	-
registrationFunction-role-801u962l	AWS Service: lambda	38 days ago

11. EC2 Console

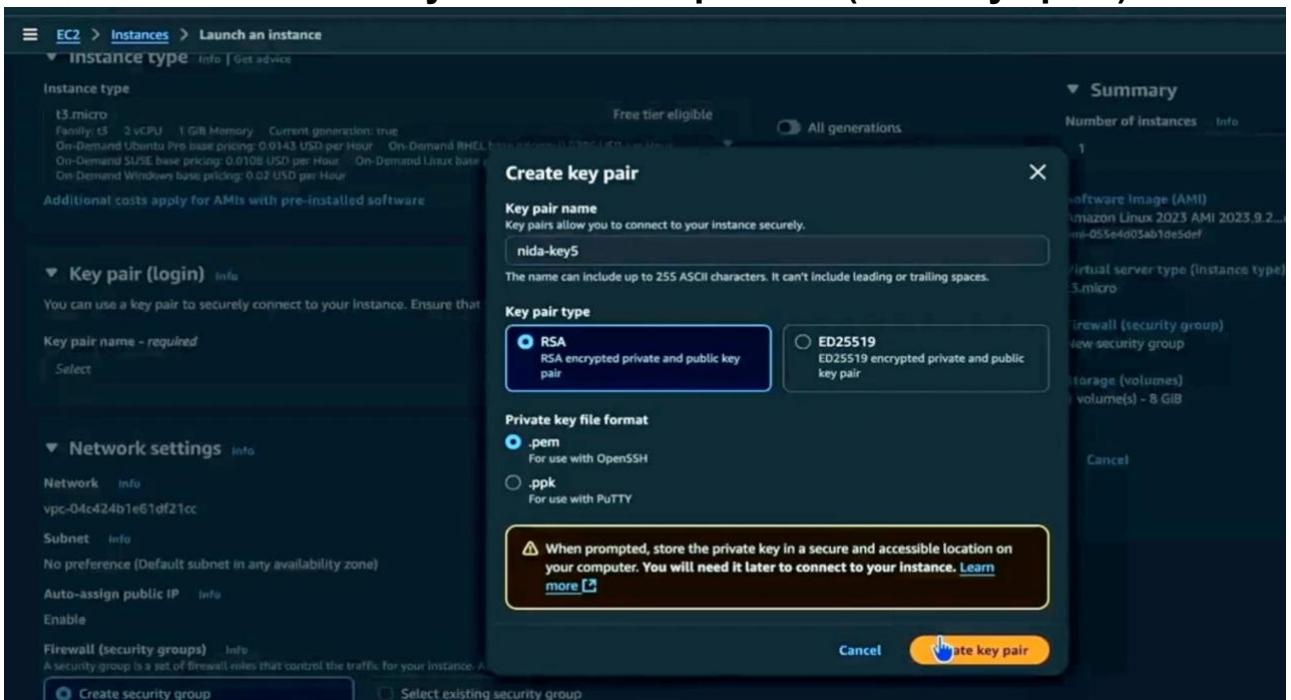
The screenshot shows the AWS EC2 Instances console. On the left, there's a navigation sidebar with options like Dashboard, AWS Global View, Events, Instances (selected), Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Capacity Manager, Images (AMIs, AMI Catalog), and Elastic Block Store (Volumes). The main area is titled 'Instances (1) info' and shows a table with one row. The row details an instance with the ID i-074b10c0c4dc5046f, which is 'Running', of type 't3.micro', has a status check of '3/3 checks passed', is in 'eu-north-1a' availability zone, and has a public IPv4 DNS of ec2-51-20-44-142.eu-n. There are buttons for 'Connect', 'Actions', and 'Launch instances'.

12. Launch Instance — Name, AMI selection and basic settings

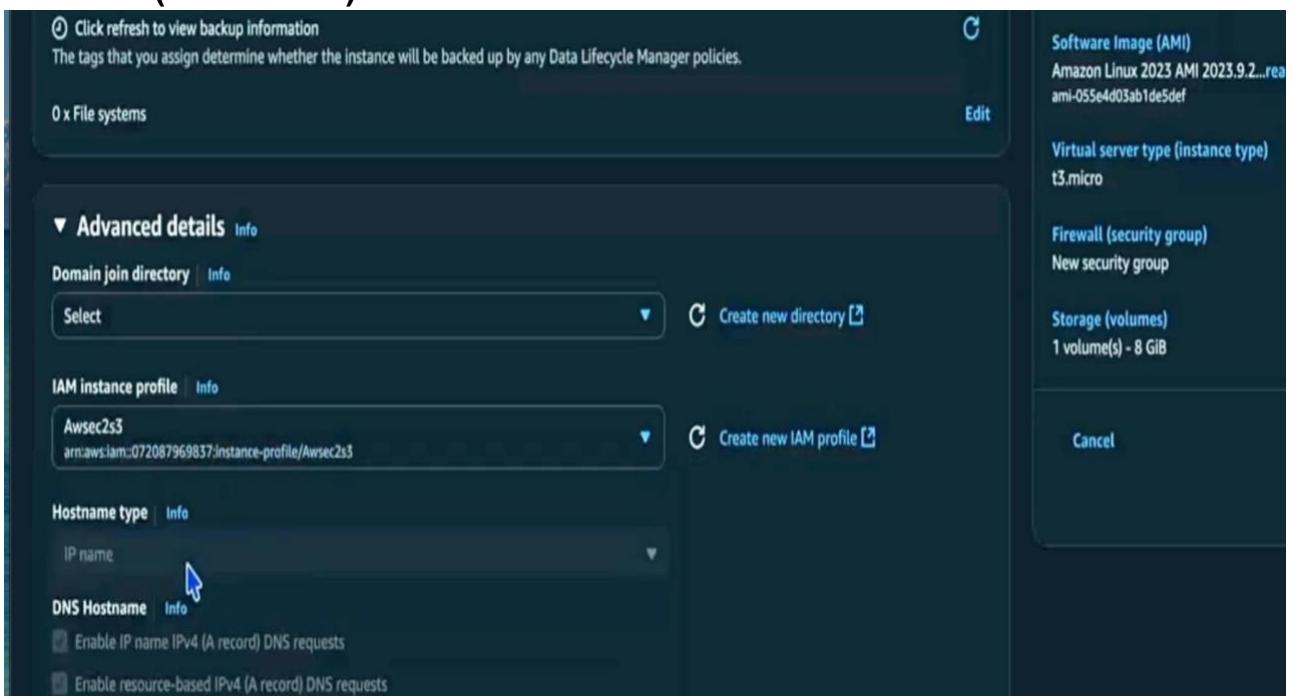
The screenshot shows the 'Launch an instance' wizard. At the top, there's a message: 'It seems like you may be new to launching instances in EC2. Take a walkthrough to learn about EC2, how to launch instances and about best practices'. Buttons for 'Take a walkthrough' and 'Do not show me this message again' are present. The main steps are:

- Name and tags**: A field where 'nida-ec2-lab' is entered, with a link to 'Add additional tags'.
- Application and OS Images (Amazon Machine Image)**: A search bar with placeholder 'Search our full catalog including 1000s of application and OS images' and a 'Quick Start' tab. Below it are buttons for 'Amazon Linux', 'macOS', 'Ubuntu', 'Windows', 'Red Hat', 'SUSE Linux', and 'Debian'.
- Summary**: Shows 'Number of instances' (1), 'Software Image (AMI)' (Amazon Linux 2023 AMI 2023.9.2...), 'Virtual server type (instance type)' (t3.micro), 'Firewall (security group)' (New security group), and 'Storage (volumes)' (1 volume(s) - 8 GiB).
- Buttons at the bottom**: 'Cancel', 'Launch instance' (highlighted in yellow), and 'Preview code'.

13. Create Key Pair — Save .pem file (nida-key5.pem)



14. Advanced Details — IAM instance profile (Awsec2s3) attached



15. SSH Terminal — Successful SSH connection to EC2 (ec2-user)

The screenshot shows an SSH terminal session. The title bar reads "Downloads — ec2-user@ip-172-31-26-127:~ — ssh -i nida-key5.pem ec...". The terminal output is as follows:

```
Last login: Sun Nov  2 11:13:05 on console
[(base) syednida@192 ~ % cd Downloads
[(base) syednida@192 Downloads % chmod 400 nida-key5.pem
[(base) syednida@192 Downloads % ssh -i "nida-key5.pem" ec2-user@51.20.132.242
The authenticity of host '51.20.132.242 (51.20.132.242)' can't be established.
ED25519 key fingerprint is SHA256:EkpmkjckO3XaRZbbF9Nt/lzK8C1sFgfvYV2a/bLdyM8.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '51.20.132.242' (ED25519) to the list of known hosts.

# 
--\_\_ #####_          Amazon Linux 2023
-- \_\#\#\#\_\_
-- \#\#\#
--   \#/__-- https://aws.amazon.com/linux/amazon-linux-2023
--     V-' '-->
--   /_/
--  /_/
-- /m/
[ec2-user@ip-172-31-26-127 ~]$
```

16. AWS CLI — Verify S3 access (aws s3 ls and list object)

The screenshot shows an AWS CLI terminal session. The terminal output is as follows:

```
[ec2-user@ip-172-31-26-127 ~]$ aws s3 ls
2025-09-10 07:42:13 bucket1-klh-a5
2025-09-10 07:42:02 bucket2-klh-a5-backup
2025-10-09 07:40:32 my-bucket1-2025
2025-11-02 06:24:03 nida-lab-bucket
2025-09-24 08:31:41 registrationvar
2025-09-10 08:09:58 varshi-website
2025-08-20 07:48:37 varshitha-klh-cse-a5
[ec2-user@ip-172-31-26-127 ~]$ aws s3 ls s://nida-lab-bucket
2025-11-02 06:24:32      389 sample.rtf
[ec2-user@ip-172-31-26-127 ~]$
```

Conclusion

All steps were completed: an S3 bucket was created and a sample file uploaded. An IAM role with S3 access was created and attached to an EC2 instance. The EC2 instance successfully accessed the S3 bucket using the AWS CLI, confirming the configuration