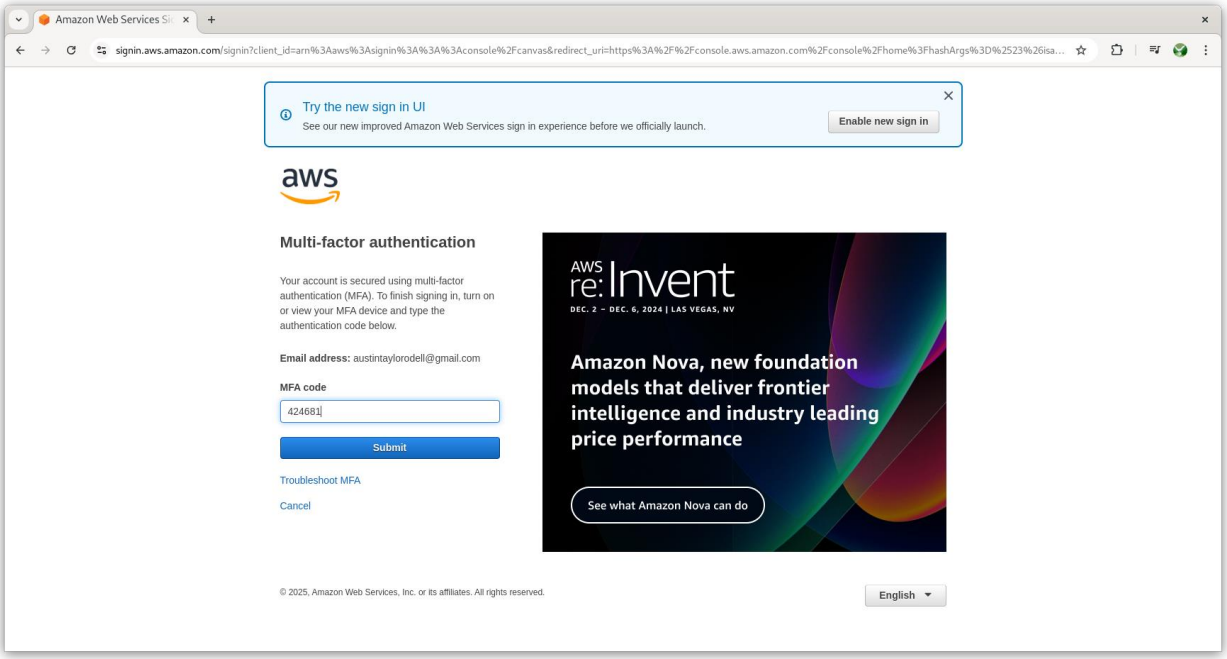
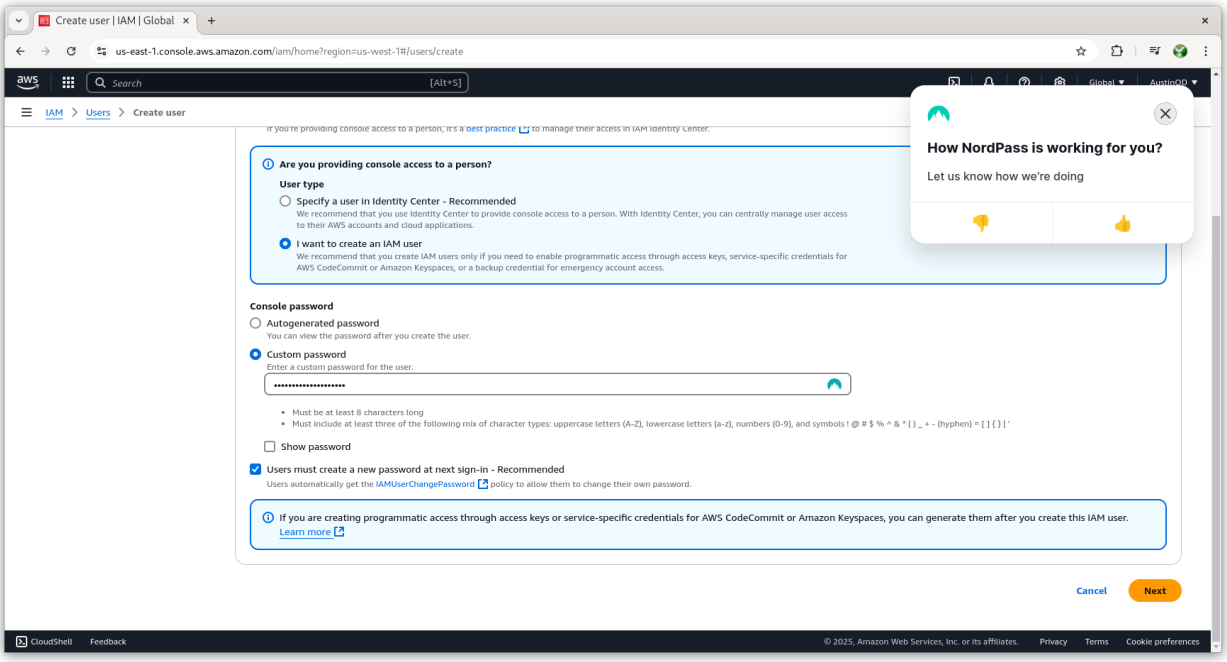
Security with AWS: Identity and Access Management

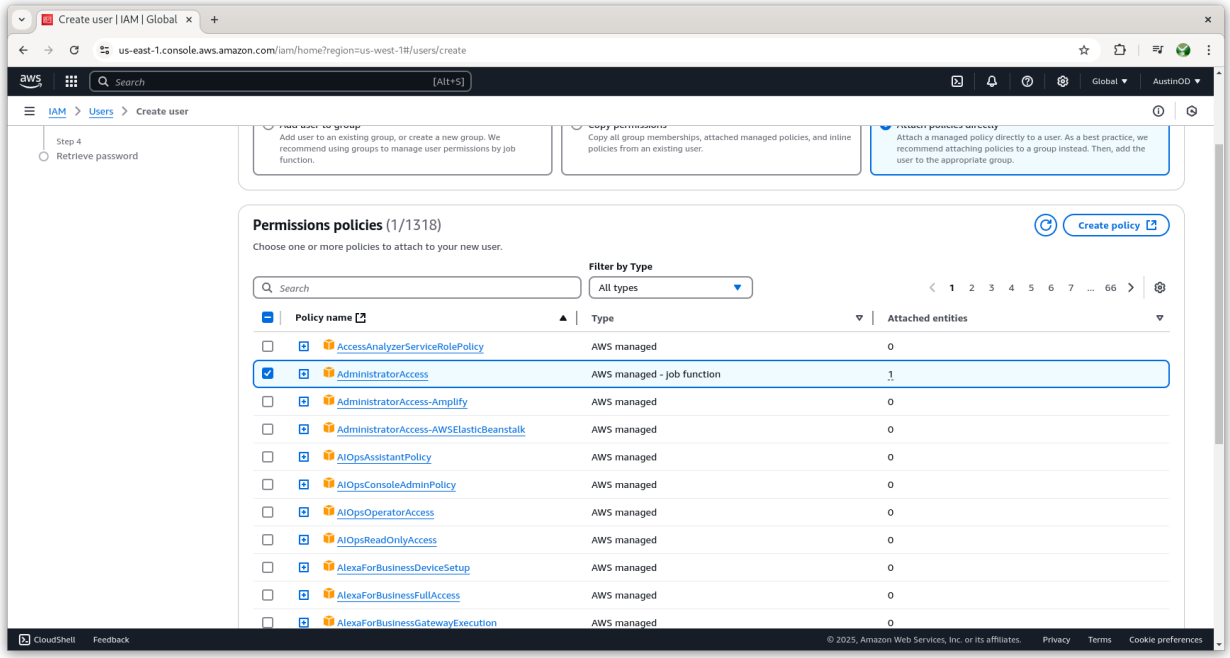
By Austin O'Dell



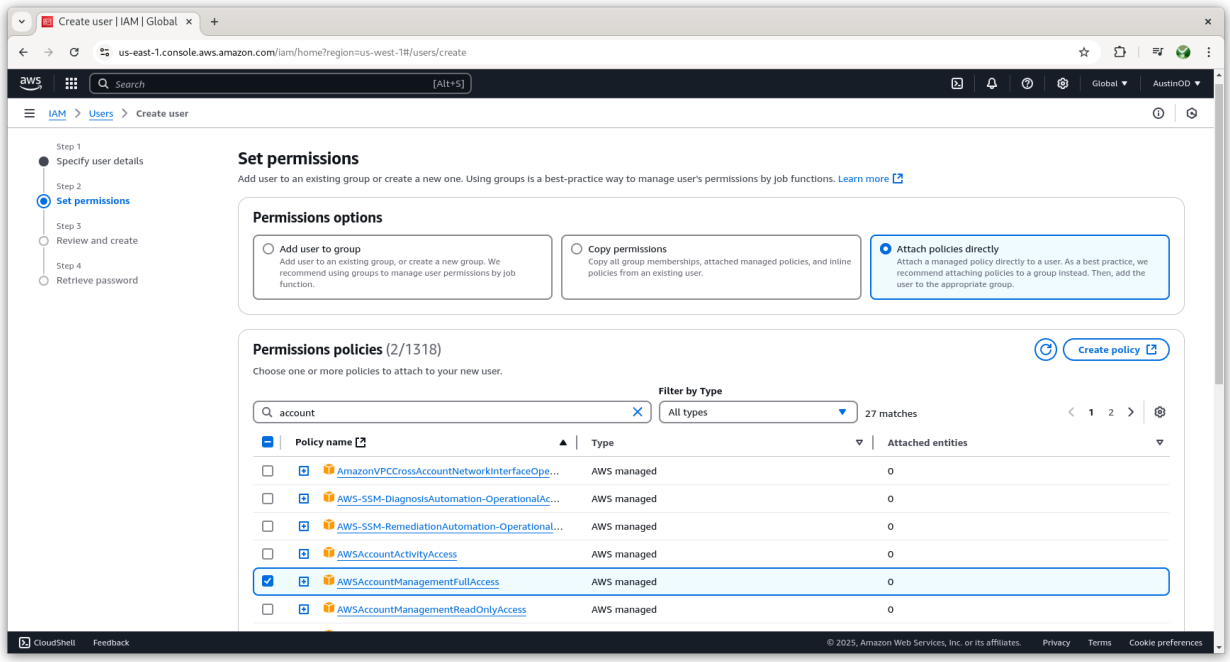
Login to root user using MFA(Never allow root logins without MFA)



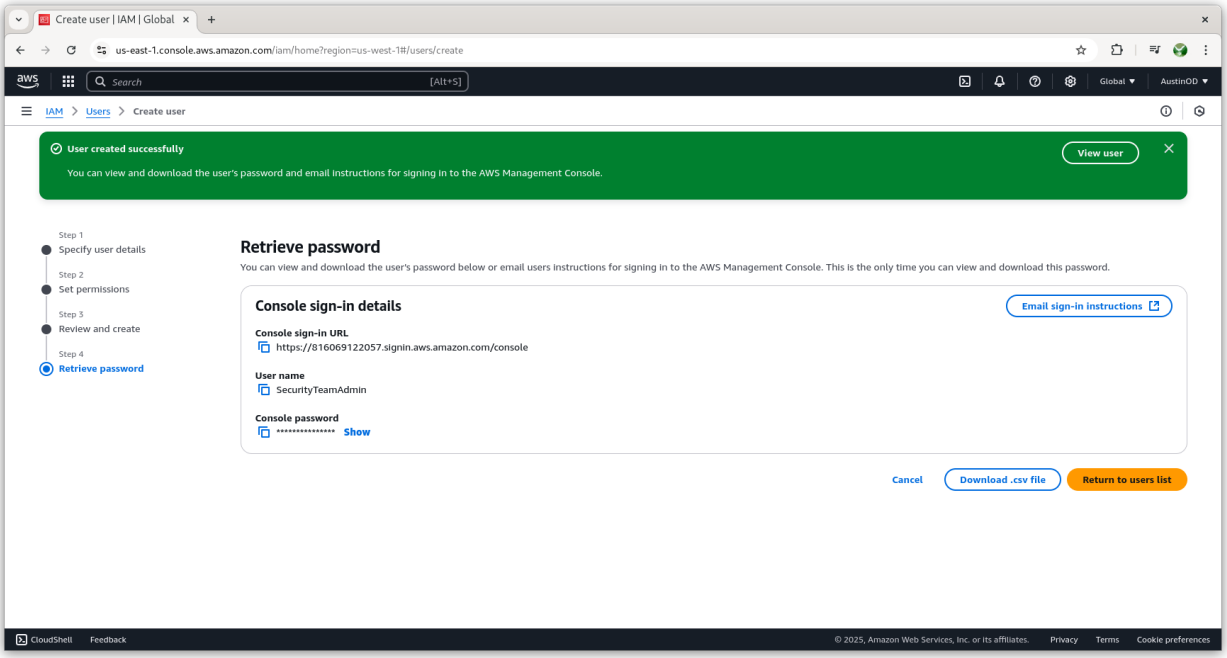
Go to iAM>Users>Create User and create a user: SecurityTeamAdmin I chose to use a custom password generated by my password manager Nordpass for ease of use.



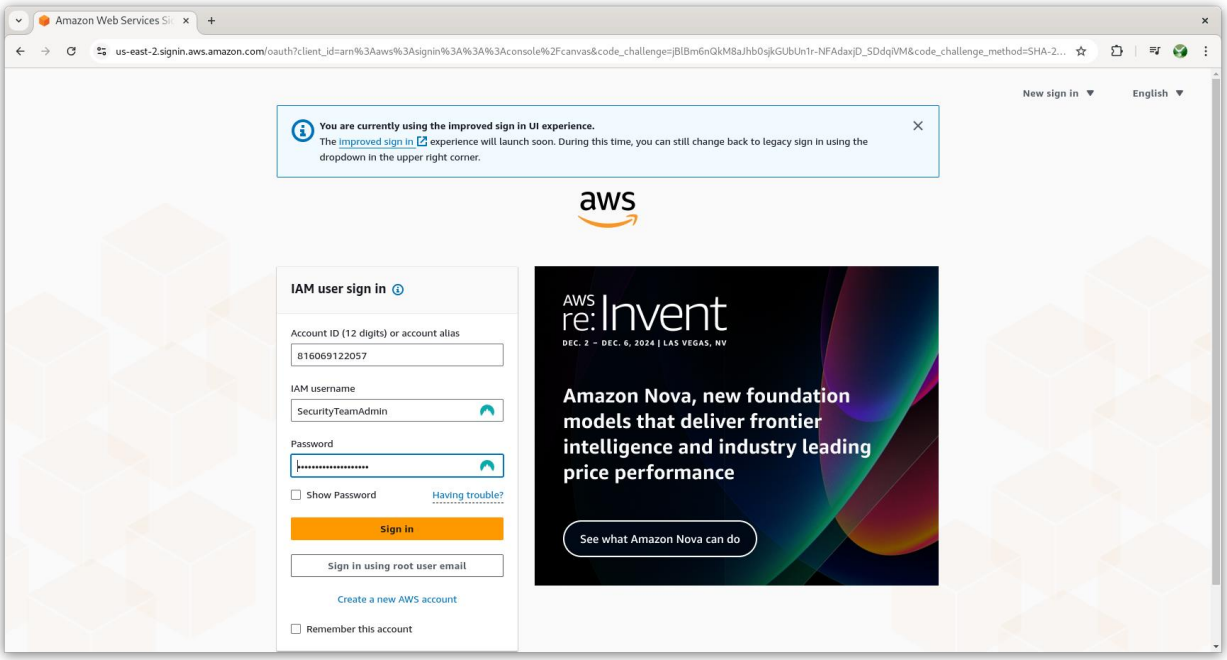
Provide the user with necessary permissions. This user will act as an administrator for the security team we are creating in this project.



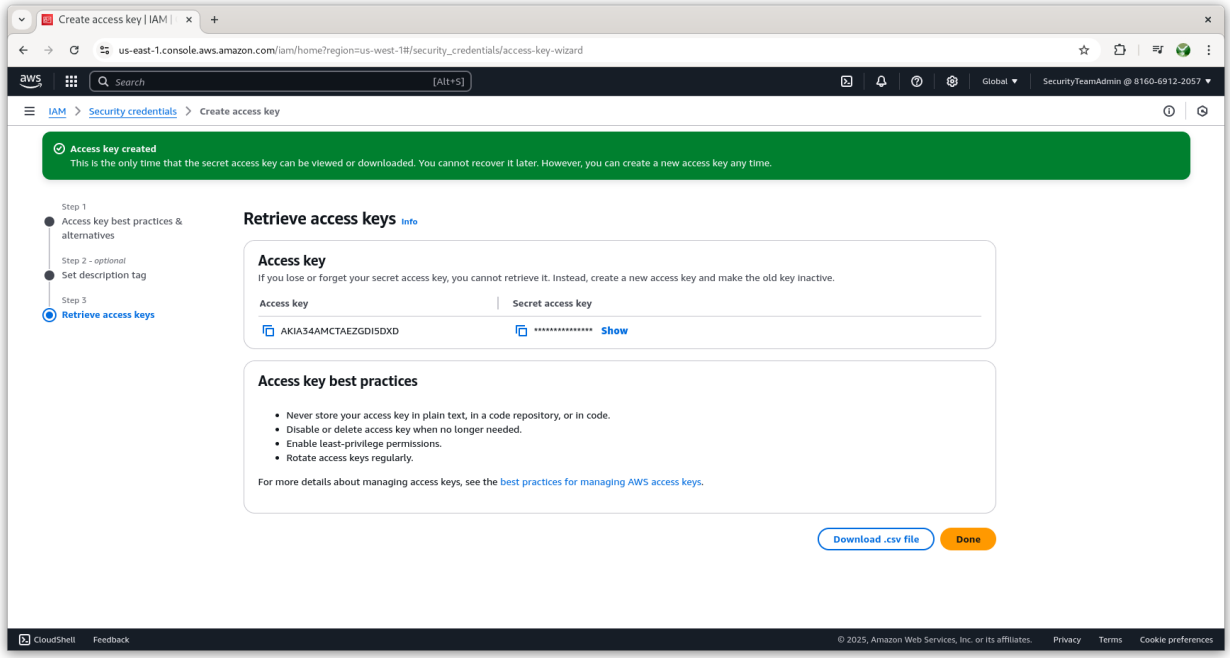
Next we assign permissions, however since we are creating this user prior to the creation of a group we will attach permissions directly. AWSAccountManagementFullAccess policy allows full management of AWS accounts including users and groups.



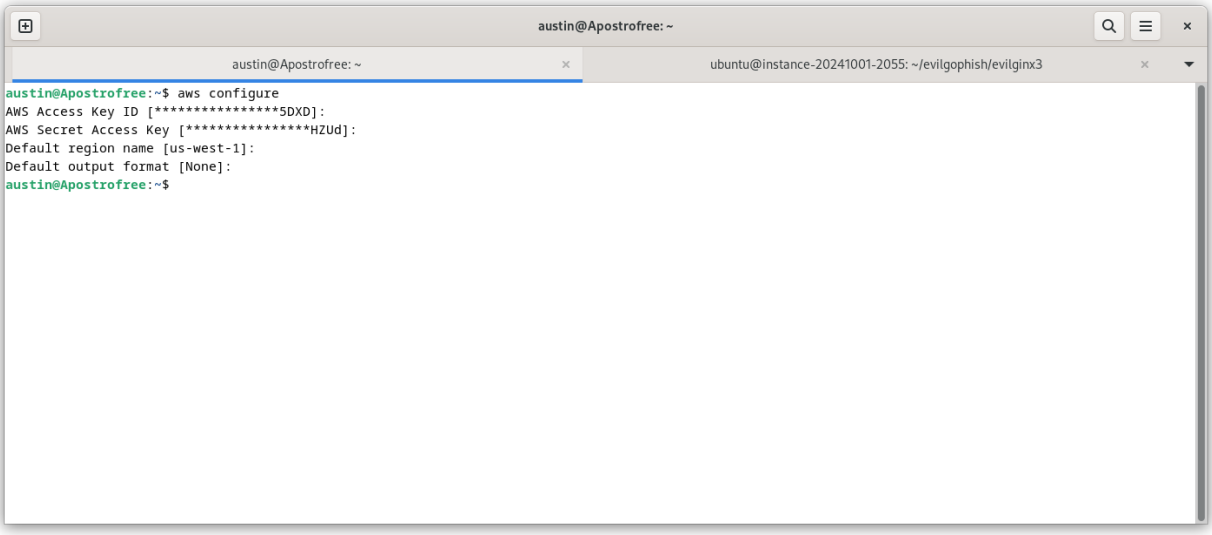
The SecurityTeamAdmin user is created and we're given the sign-in console URL.



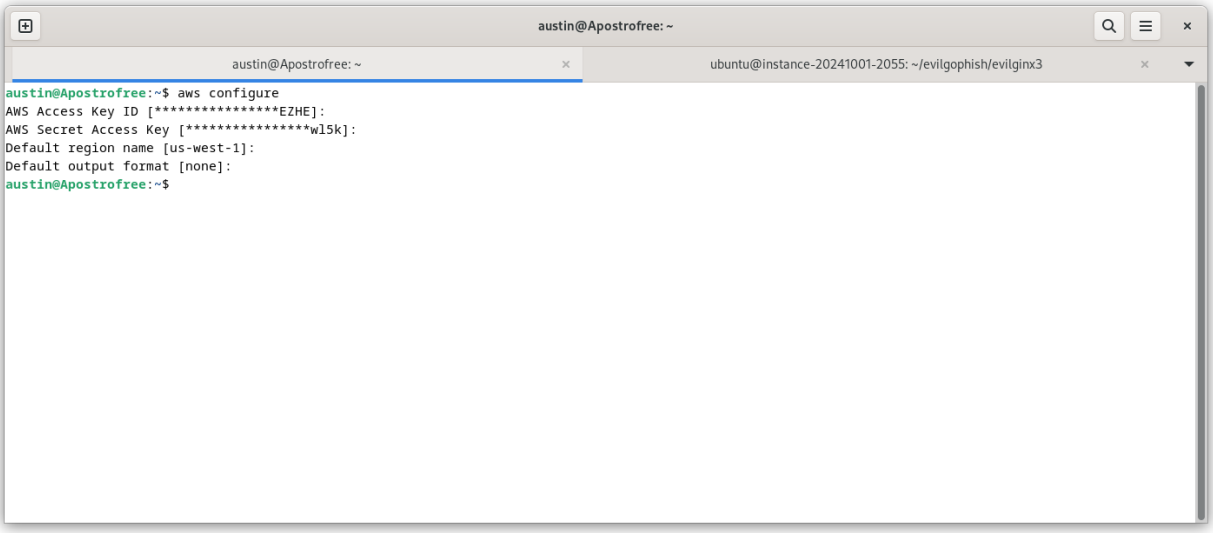
Opening the URL we're presented with the login page for the newly created user.



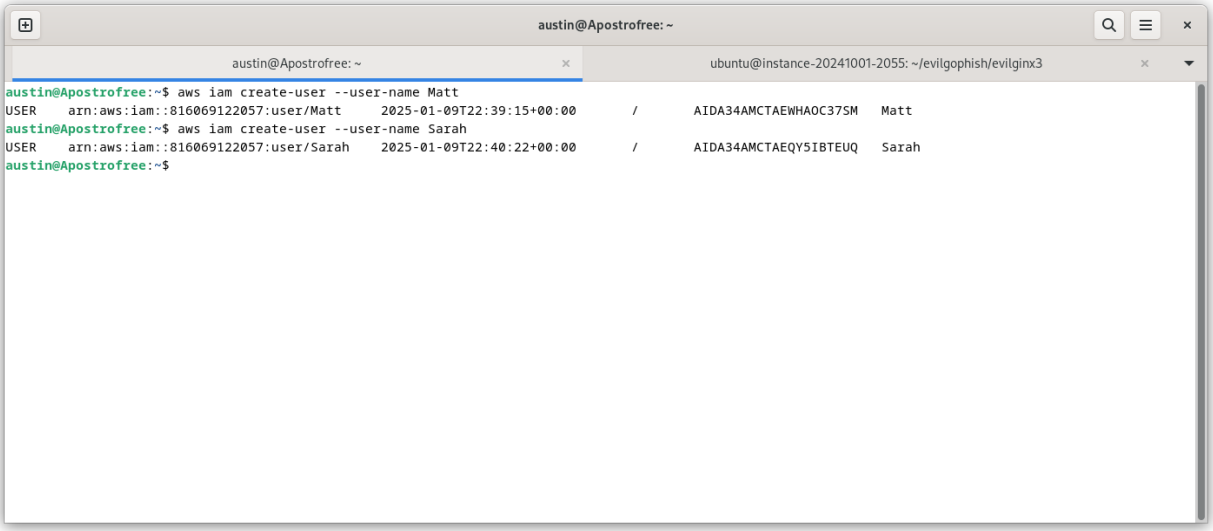
Next we go to IAM > Security Credentials > Create access key and we will generate an access key to use with the AWS CLI so that we can manage resources from our terminal as the newly created SecurityTeamAdmin user.



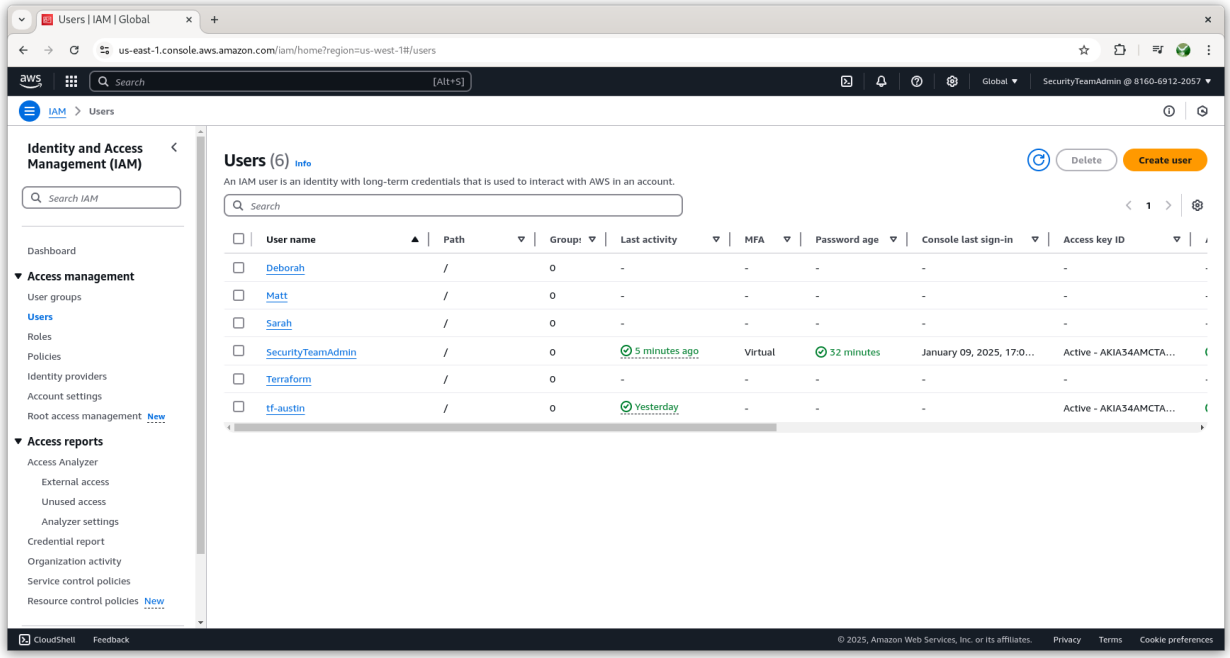
Run *aws configure* and input the access key and secret access key we created above for the SecurityTeamAdmin user. The results of running *aws configure* above show that the keys have already been configured and any further aws commands will be run as the SecurityTeamAdmin.



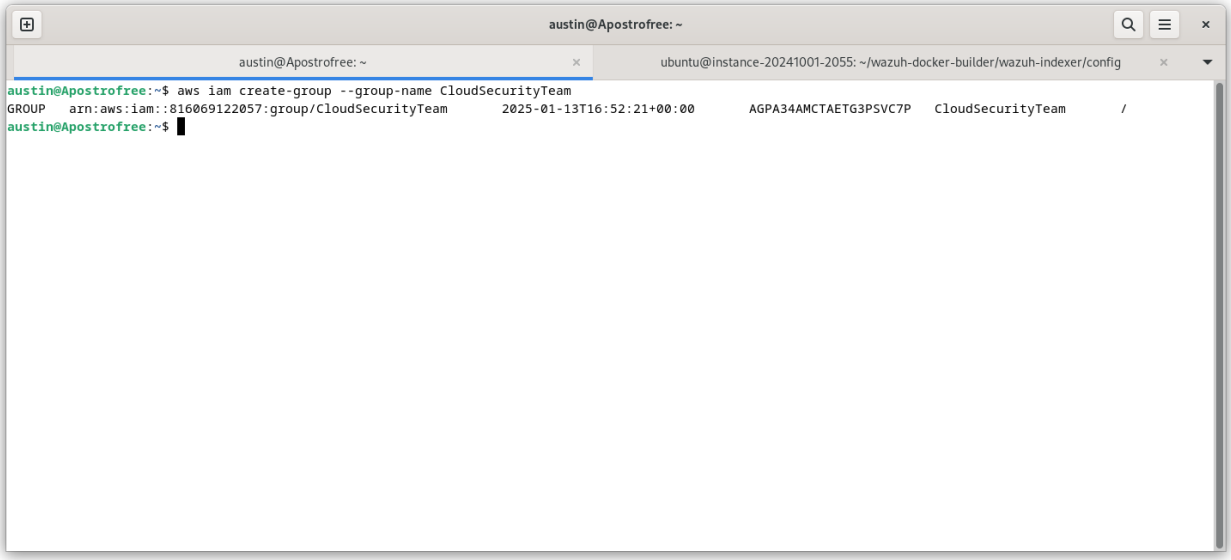
These keys can be revoked and regenerated, and new keys can be configured for the same user as shown above.



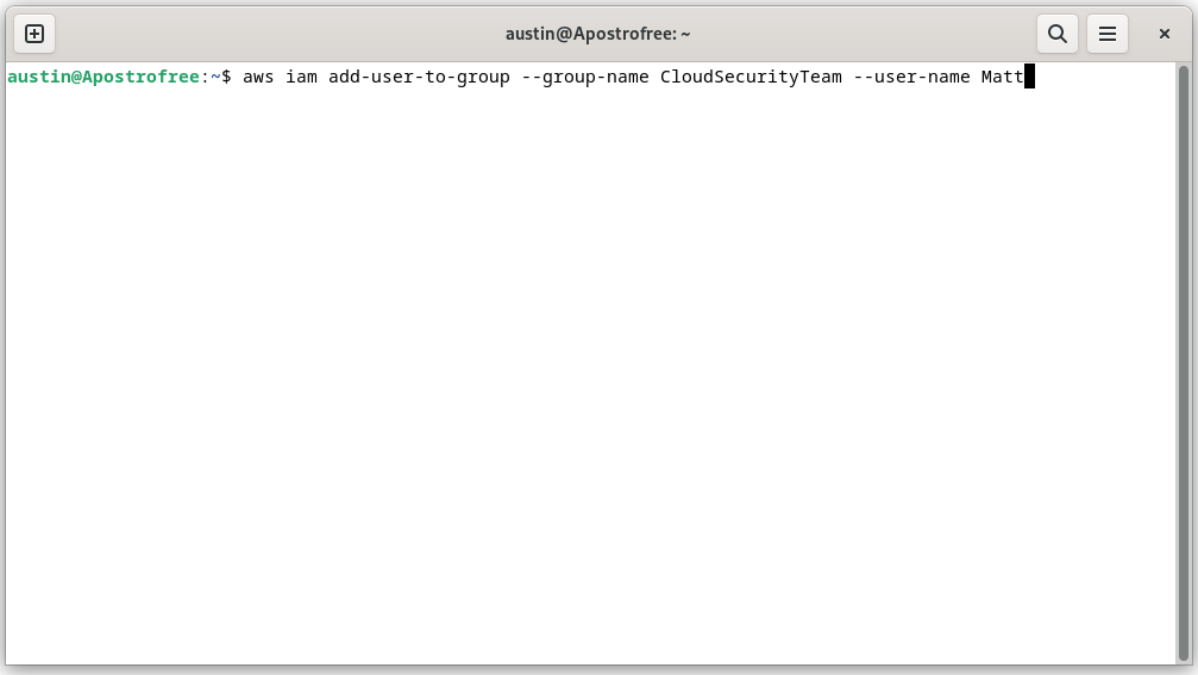
Next we use *aws create-user --user-name <user>* as SecurityTeamAdmin to create three users Matt, Sarah, and Deborah.



The created users are shown graphically in the aws console above. We can also delete users with *aws iam delete-user --user-name Deborah* for instance to delete the user Deborah.

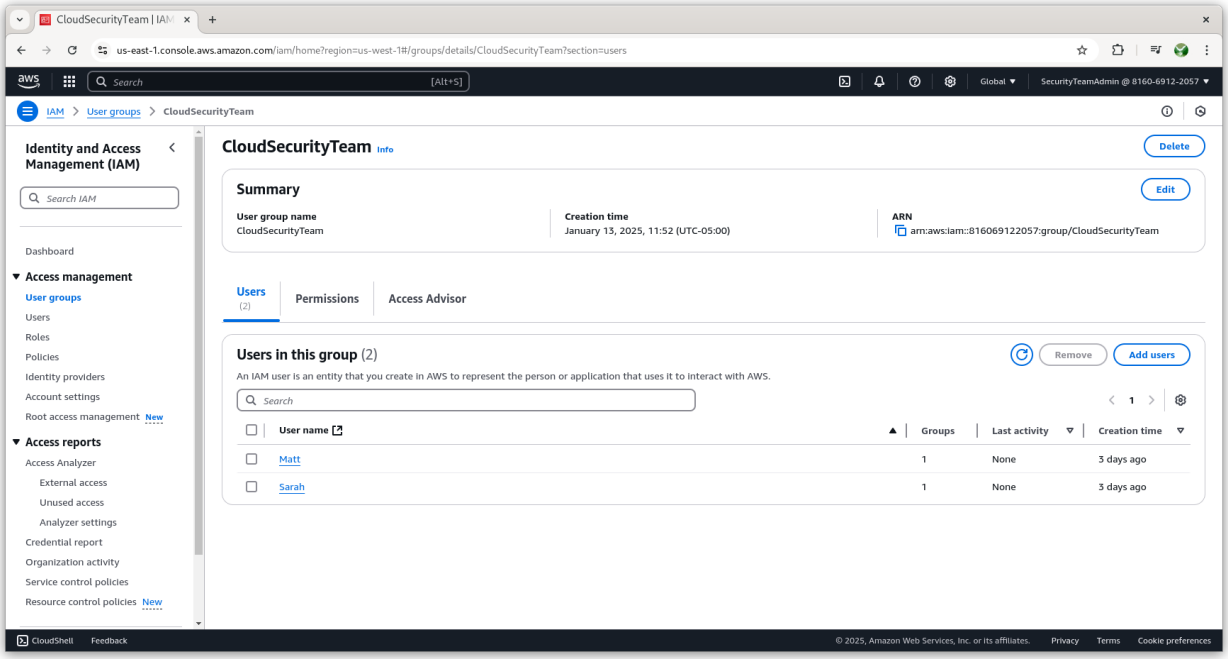


Next we run *aws iam create-group --group-name CloudSecurityTeam* to create a group named CloudSecurityTeam



We add Matt and Sarah to the CloudSecurityTeam group with:

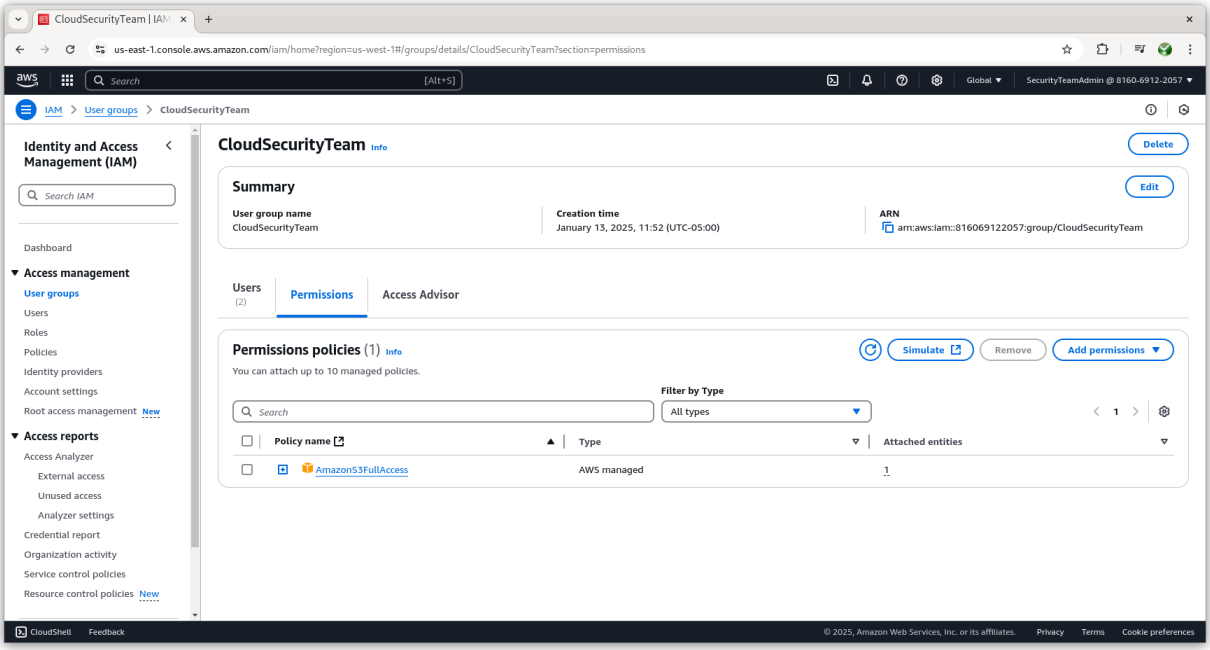
aws iam add-user-to-group --group-name CloudSecurityTeam --user-name <user name>



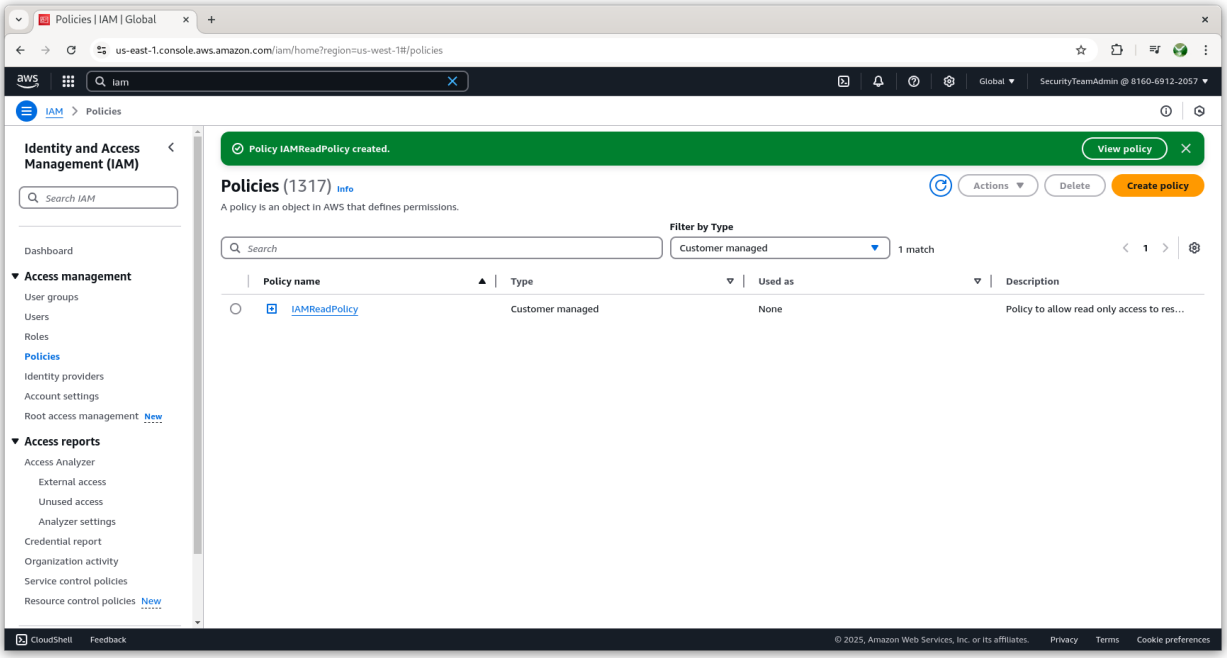
Observe that both users have been added to the group at IAM>User Groups>CloudSecurityTeam in the AWS console.

We can attach the AmazonS3FullAccess managed policy to the group by using

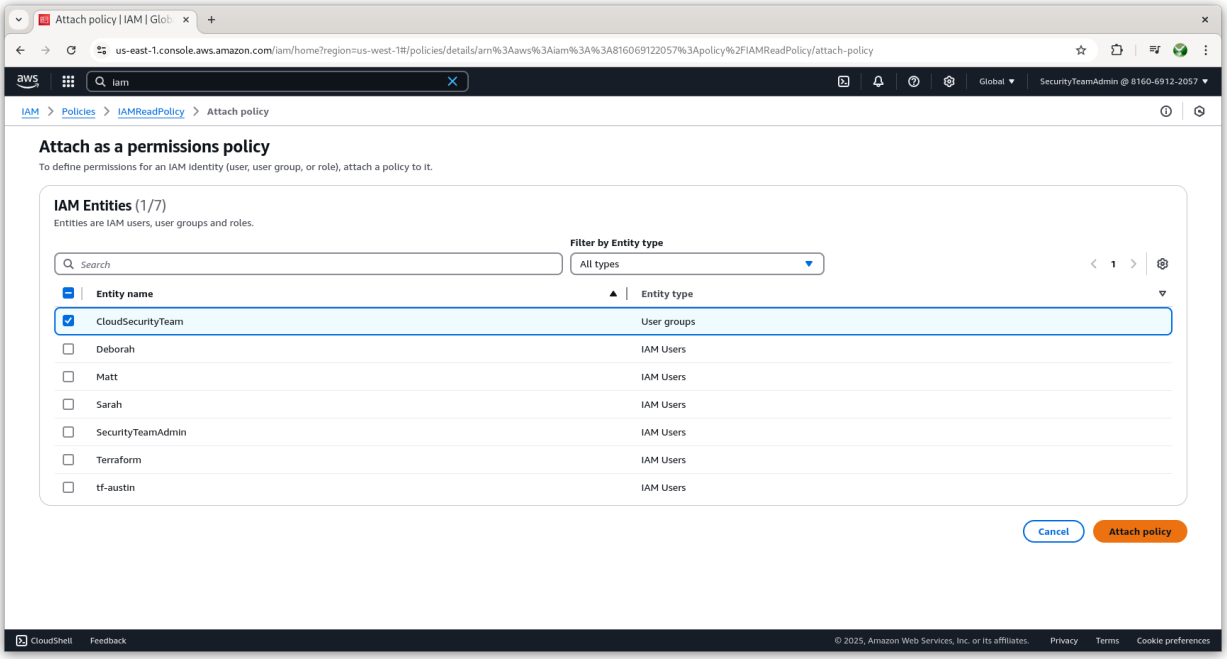
aws iam attach-group-policy --group-name CloudSecurityTeam --policy-arn arn:aws:iam::aws:policy/AmazonS3FullAccess or we can add the policy to the group in the AWS console.



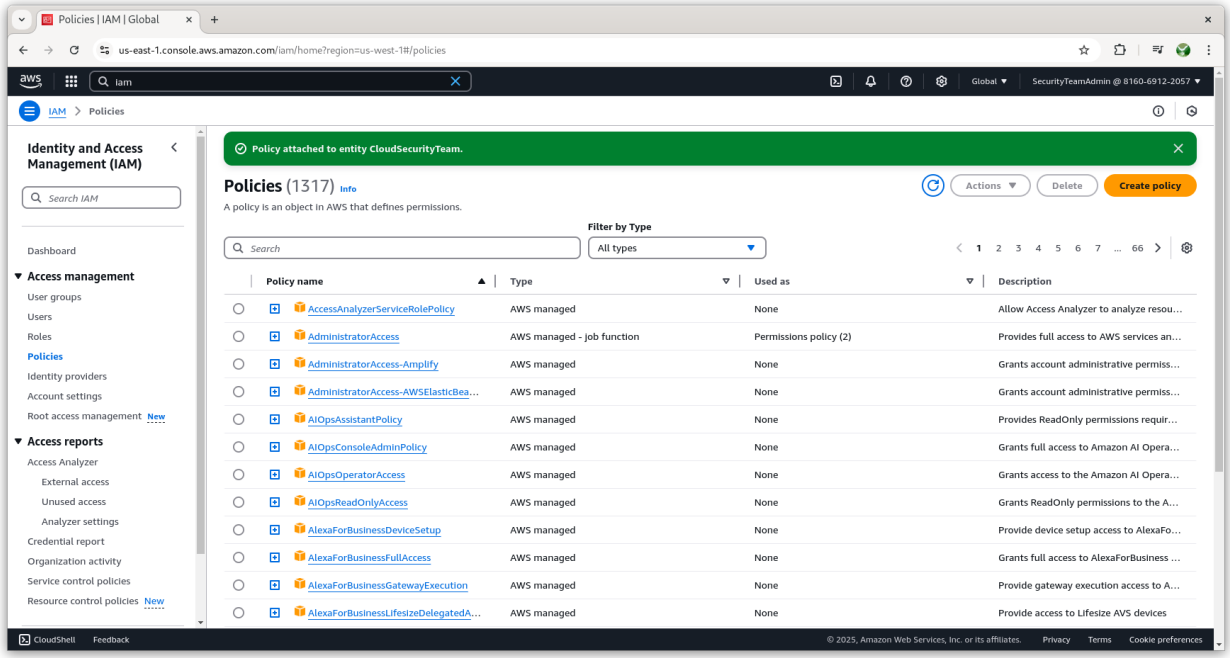
We see that that the AmazonS3FullAccess permissions policy is attached to the CloudSecurityTeam group on the AWS console above.

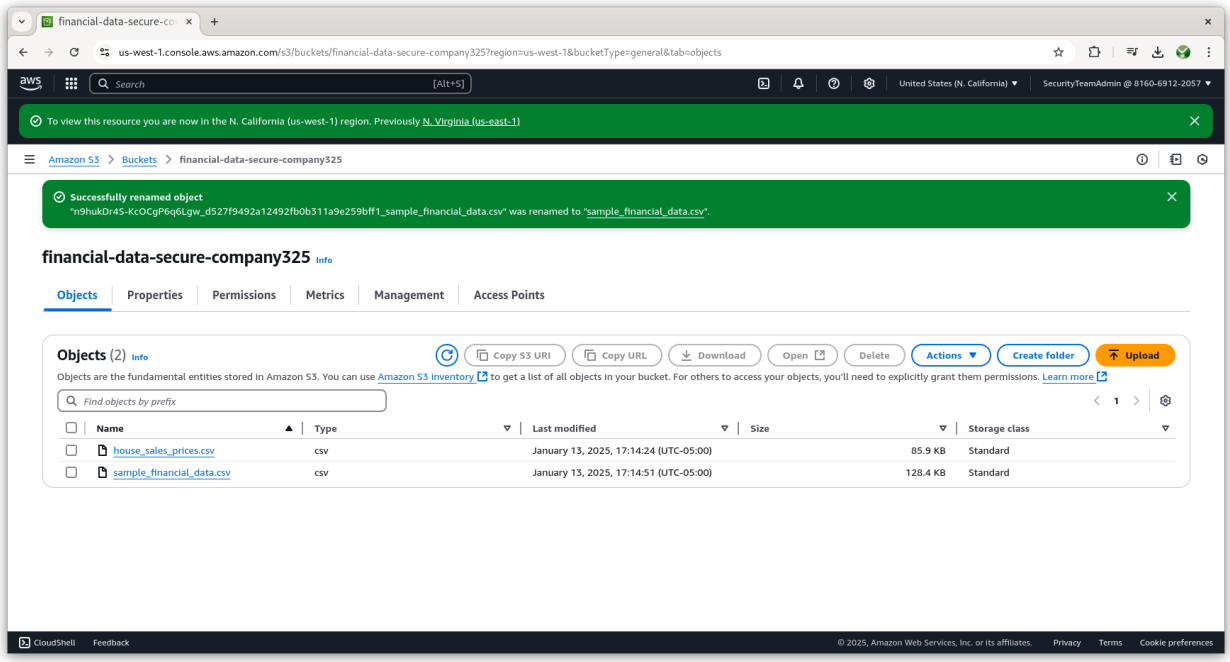


Next we go to IAM>Policies> Create Policy and create a customer managed policy to allow read only access to all IAM objects and components.

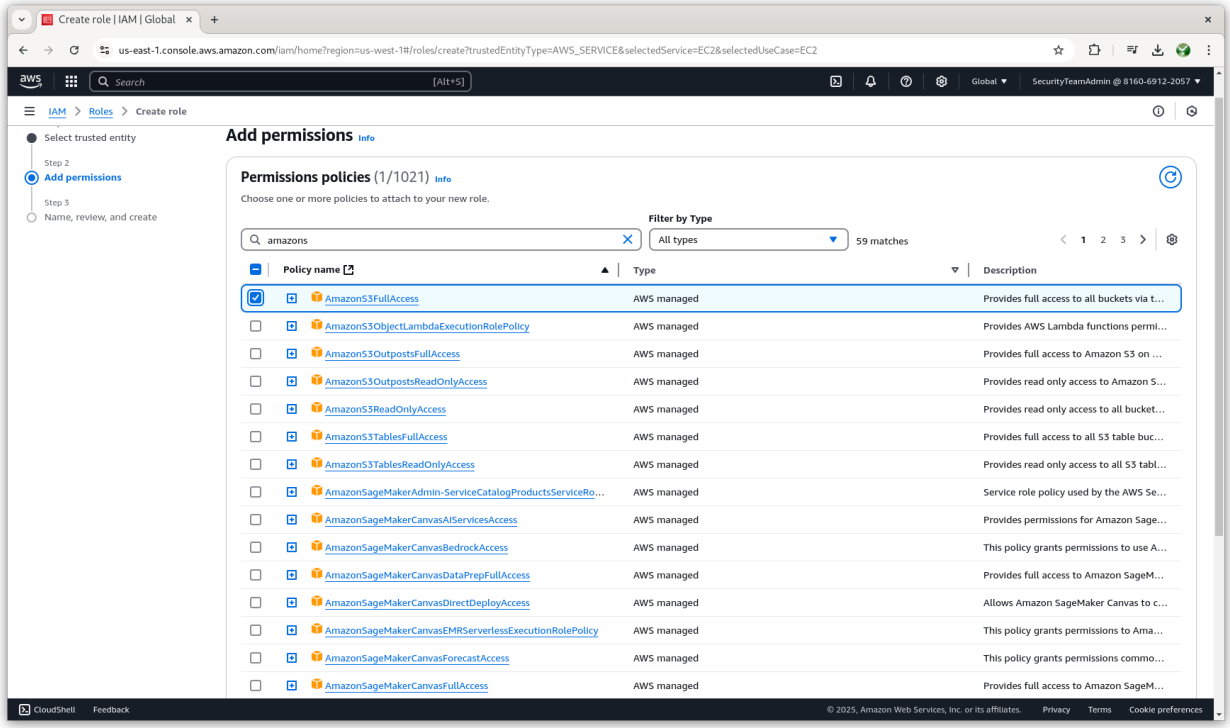


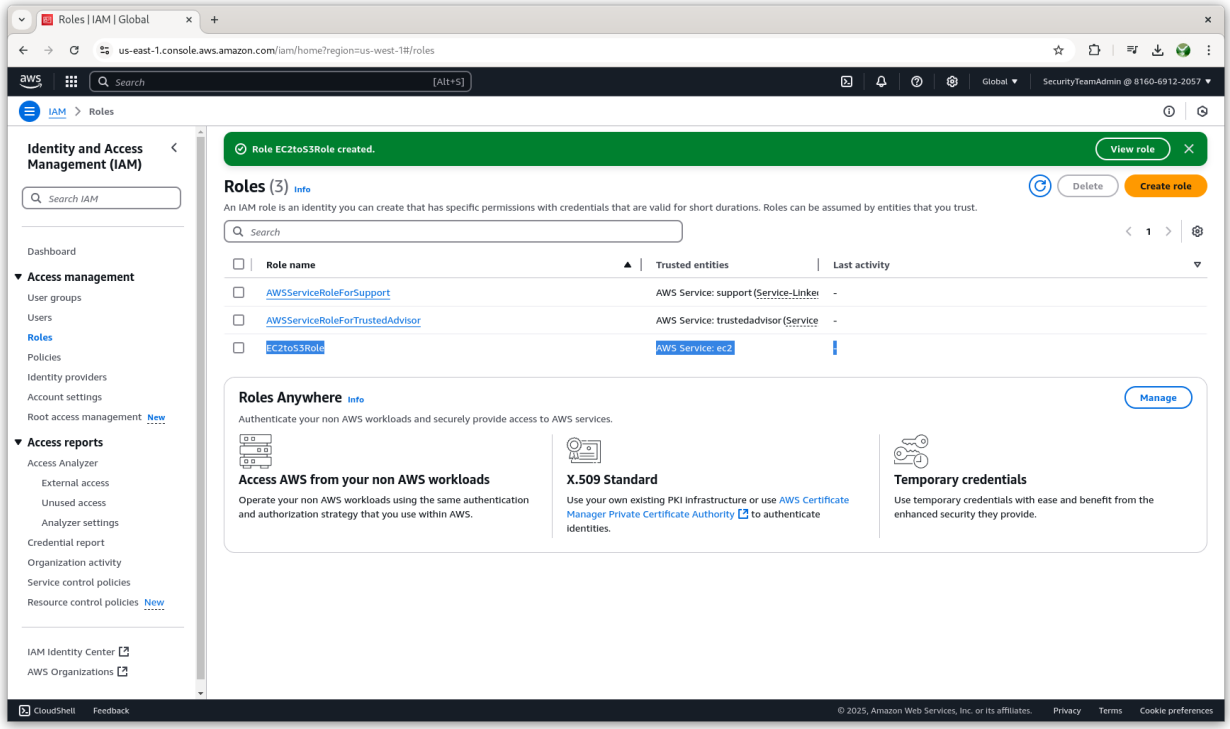
Attach this policy to the CloudSecurityTeam group.



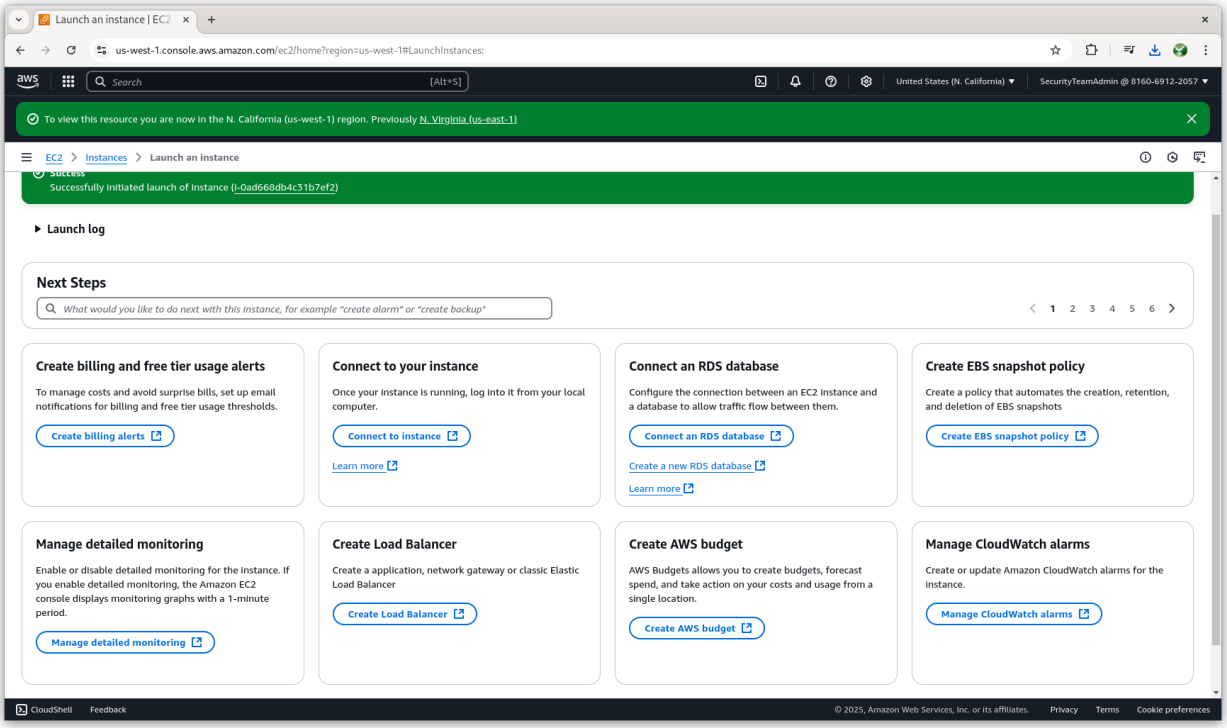


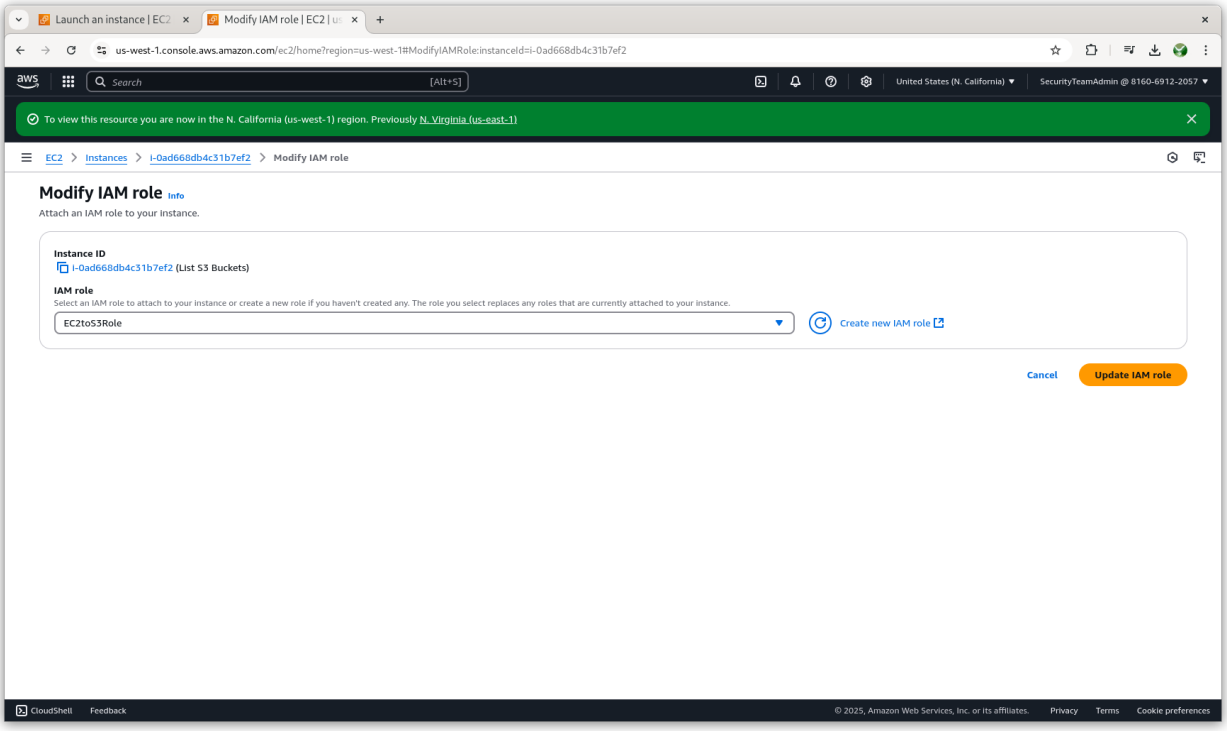
Next we upload and rename the sample confidential financial data that we will be working with.



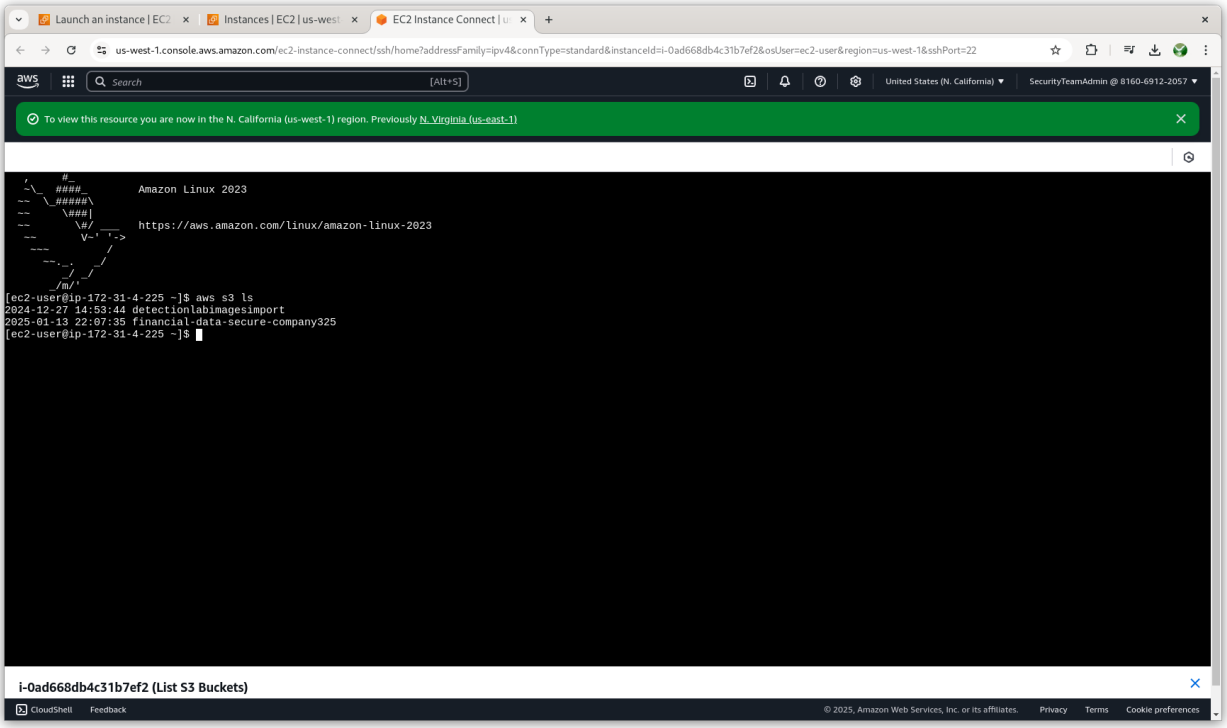


Next we create an IAM role to allow an EC2 instance to access S3 data named EC2toS3 role.

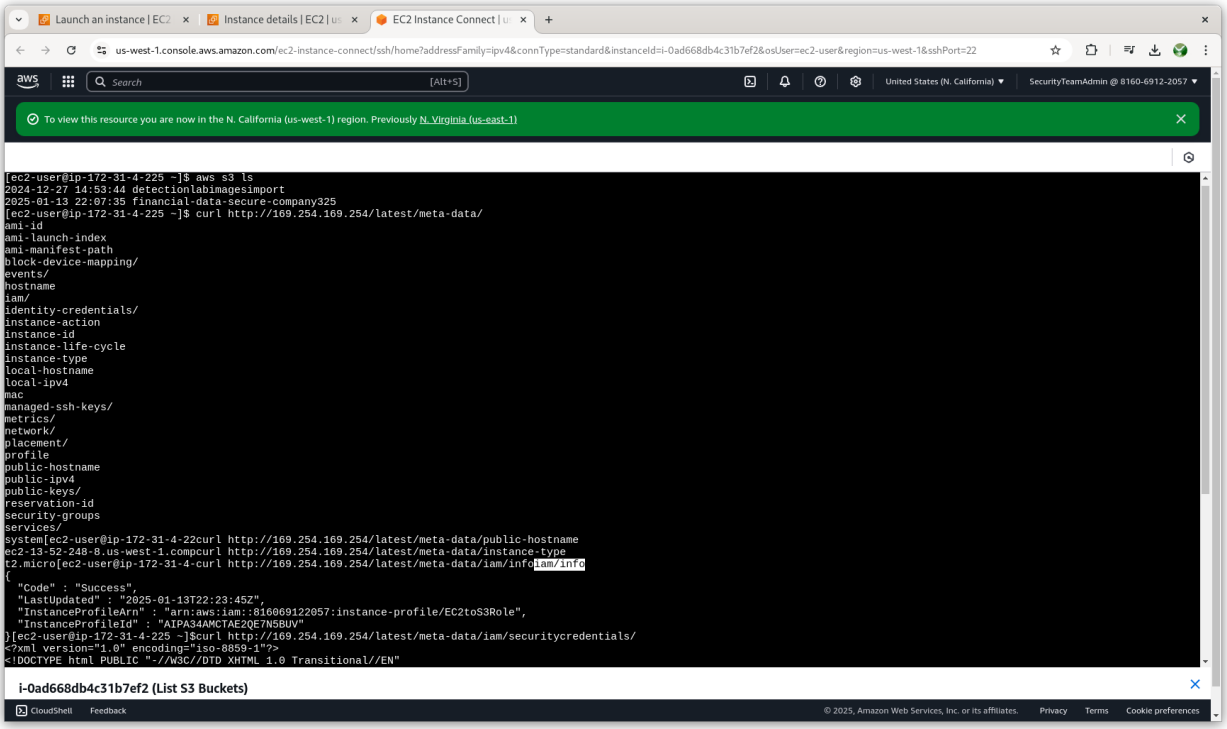




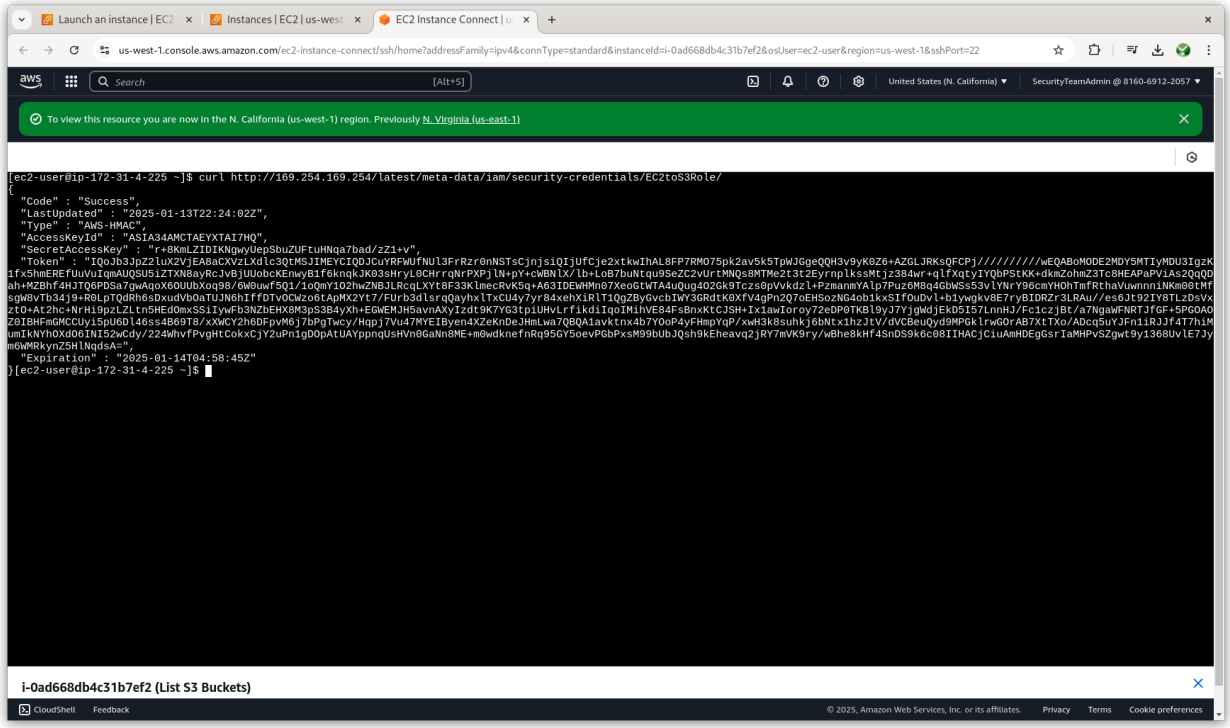
Next we create an EC2 instance and attach the role we previously created to it.



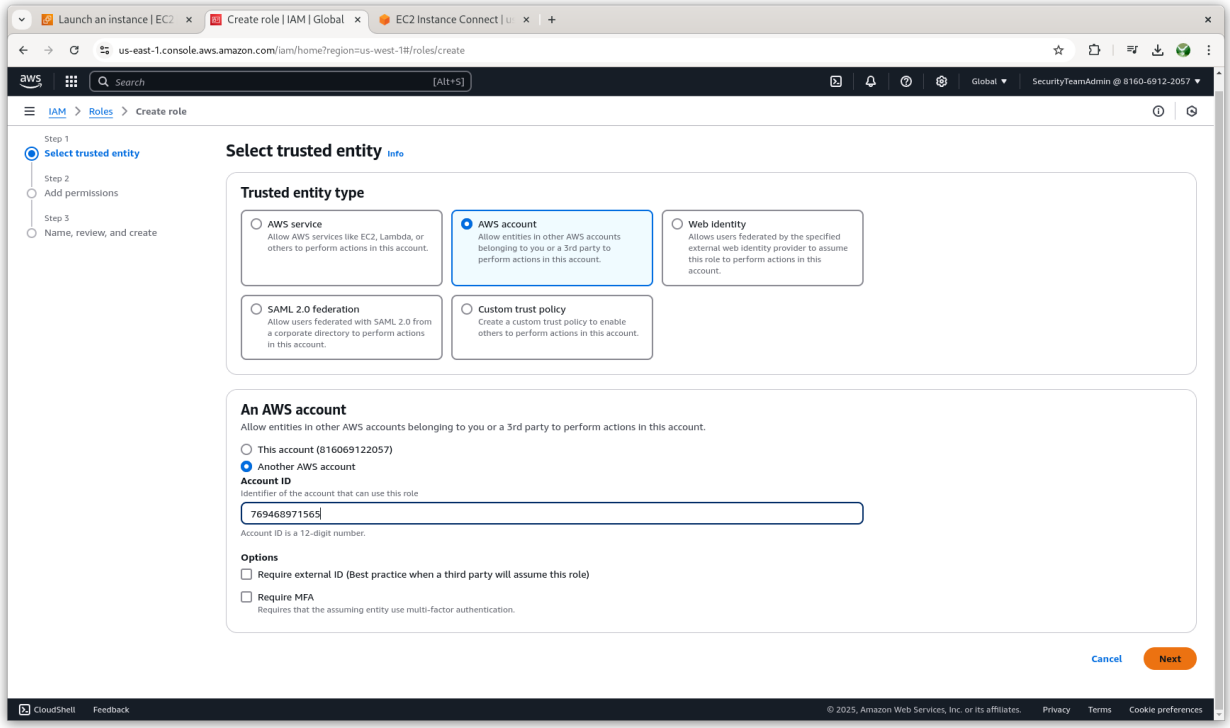
Here we see that the EC2 instance can access the S3 data by running *aws s3 ls* from the instance.



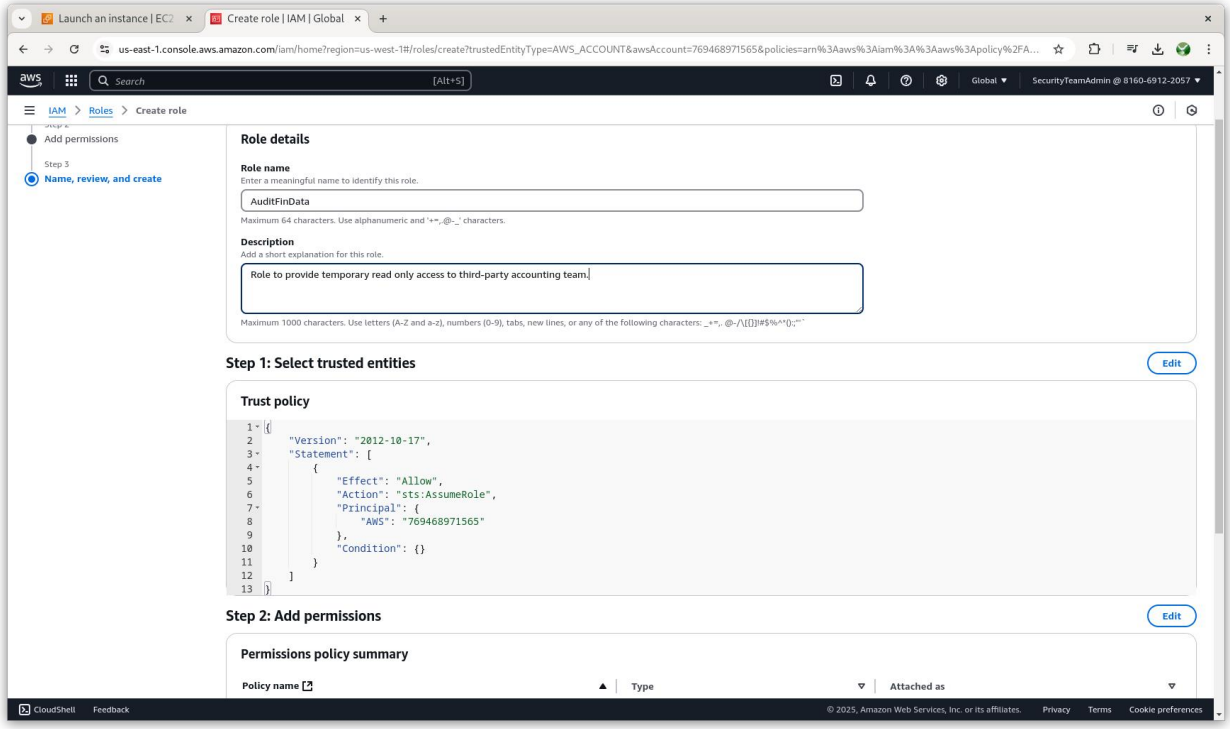
We can run *curl http://169.254.169.254/latest/meta-data/iam/info* to see the role that the EC2 has assumed.



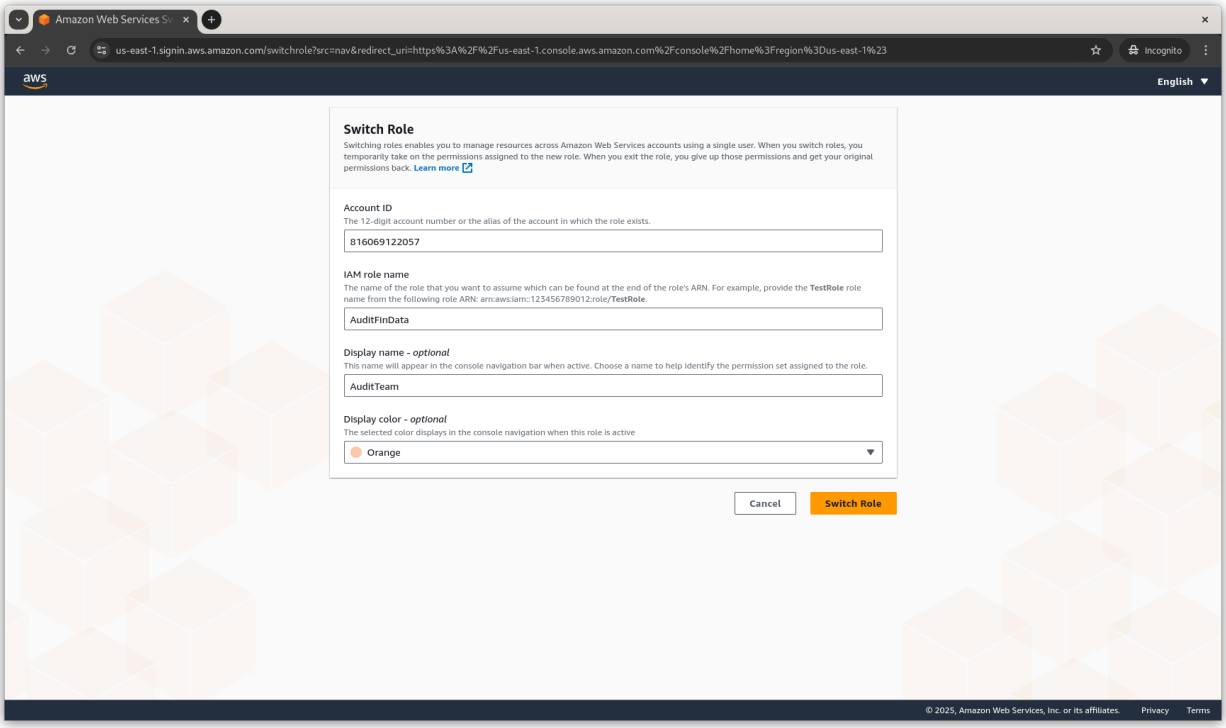
We can run curl http://169.254.169.254/latest/meta-data/iam/security-credentials/ to see the temporary security token used by the EC2 instance for its assumed role. Using IAM roles is a security best practice because roles are easy to revoke and access tokens are temporary.



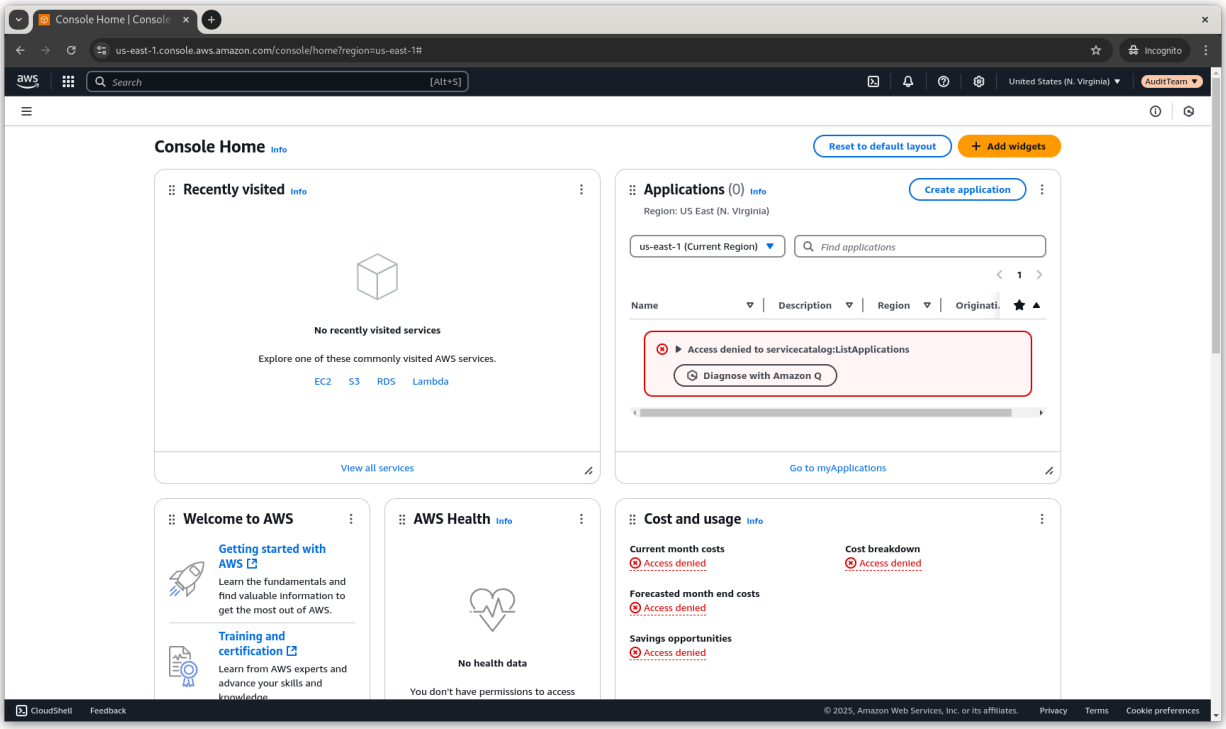
Next we create a role for an external auditor to assume to access the financial data uploaded to our S3 without having to provide access to our AWS account. Under create role > Select trusted entity we select AWS account and select Another AWS account and input the account ID of the external auditor.



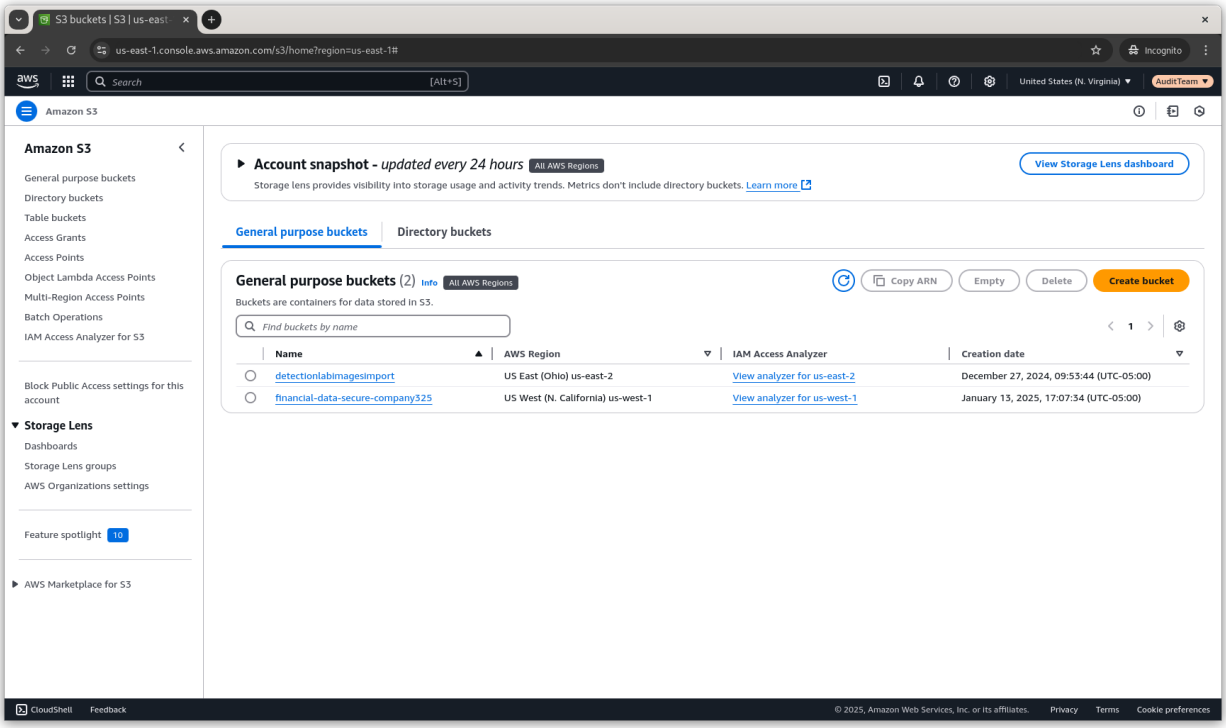
We name the role AuditFinData and provide read only access to S3 data.



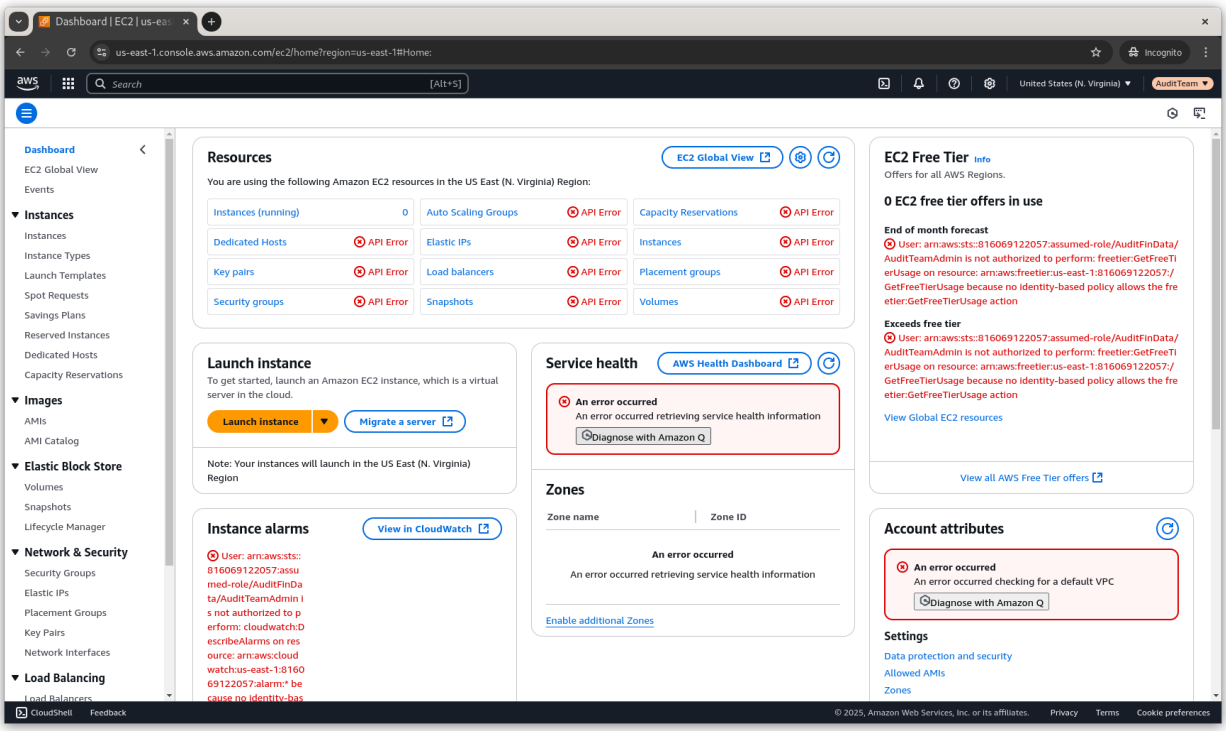
After logging in to the auditor account in an incognito window we can switch roles to the newly created AuditFinData role.



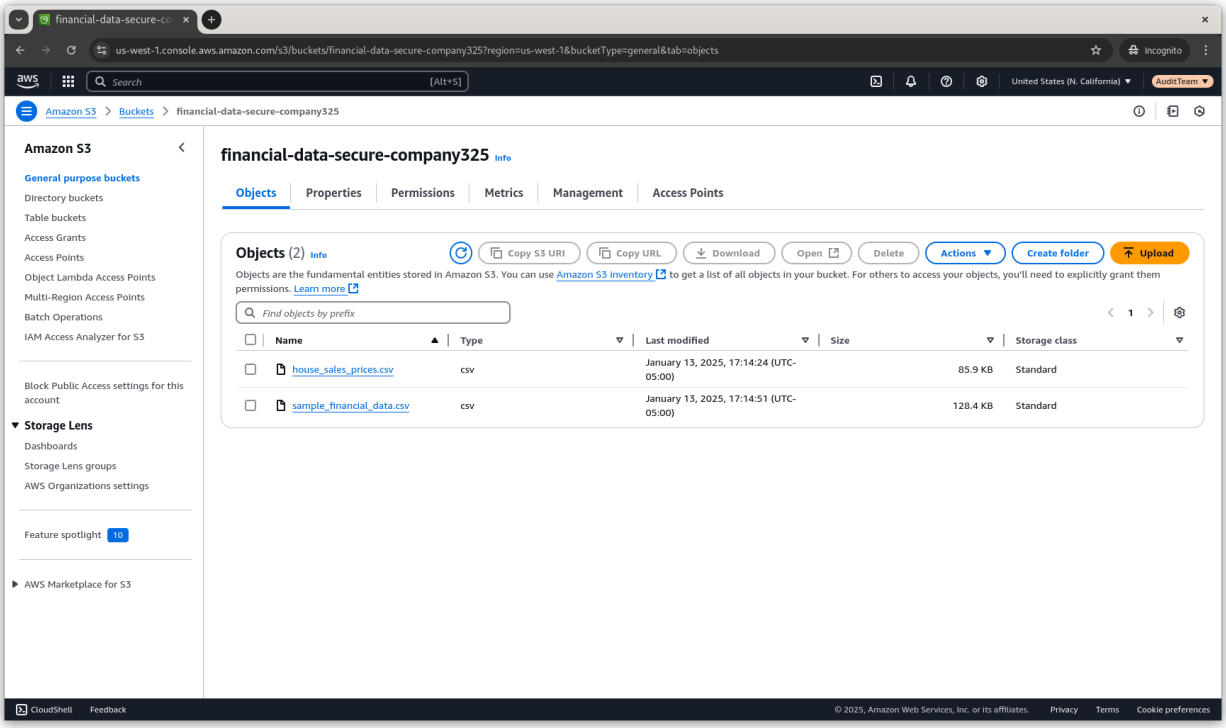
We see that the console for the auditor is filled with Access Denied warning because we have limited their access to specific resources.



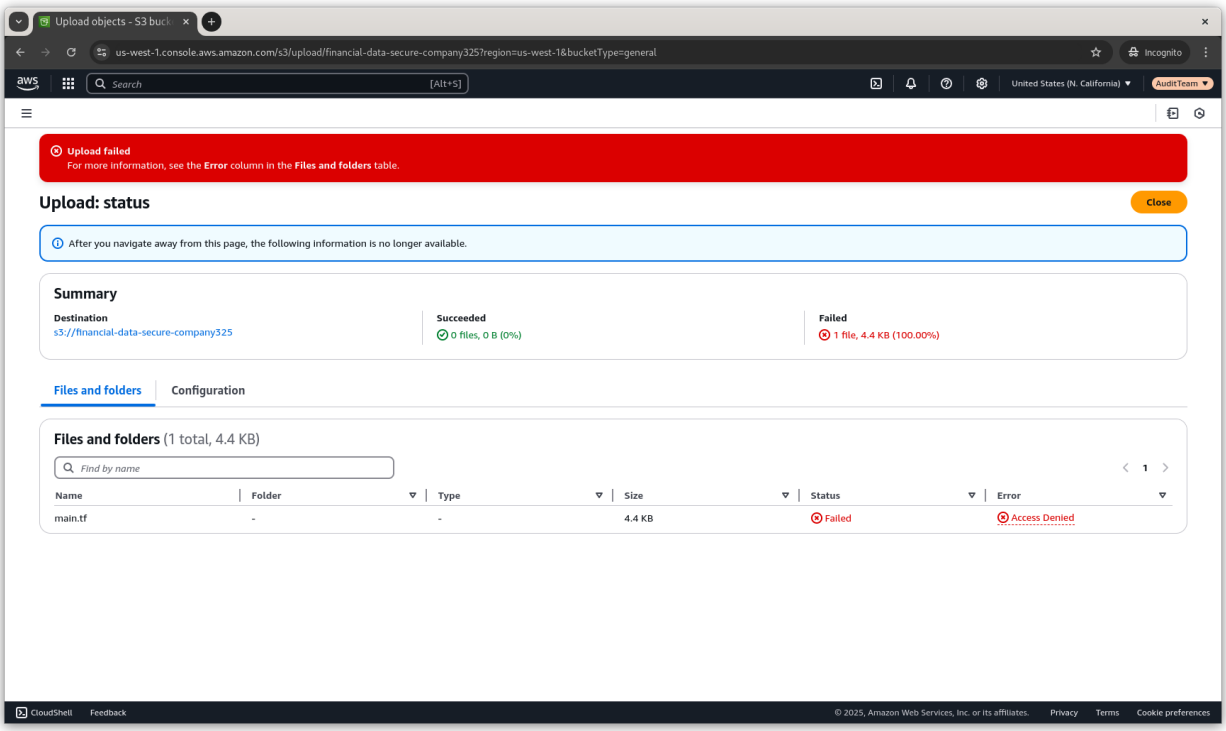
However the auditor is able to list files on S3.



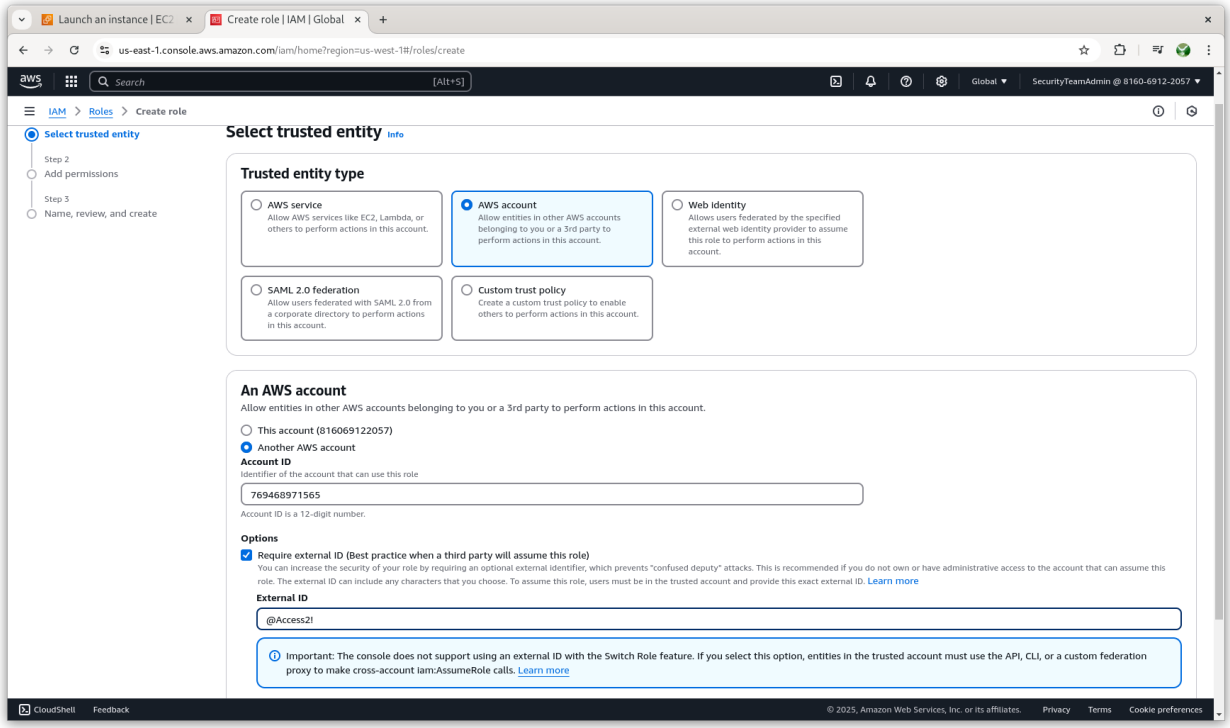
They are not allowed to modify, access, or launch EC2 instances.



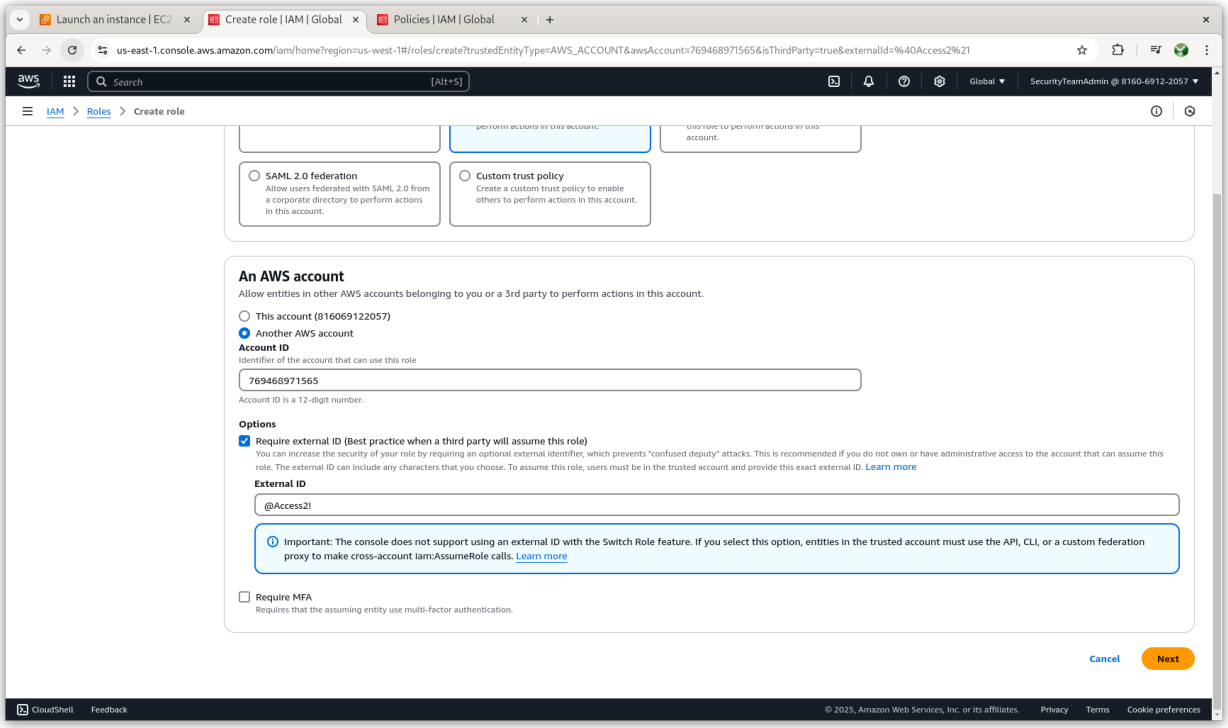
But they can access the required financial data.

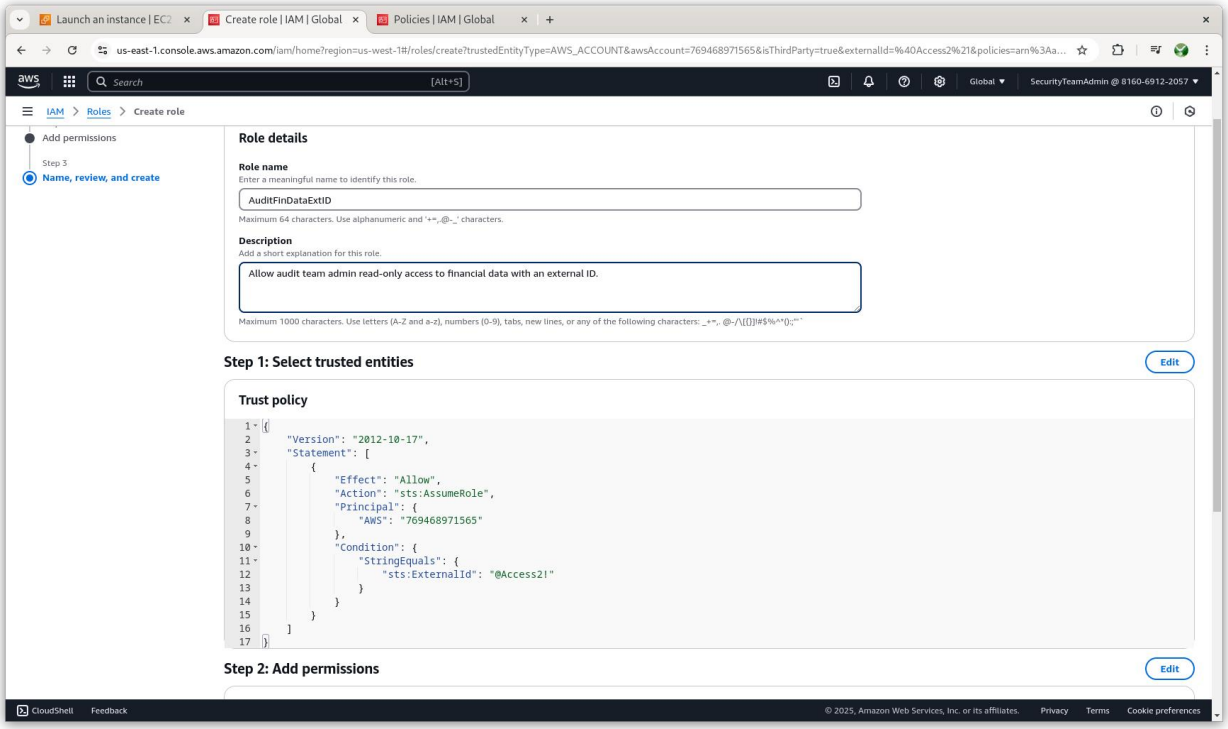


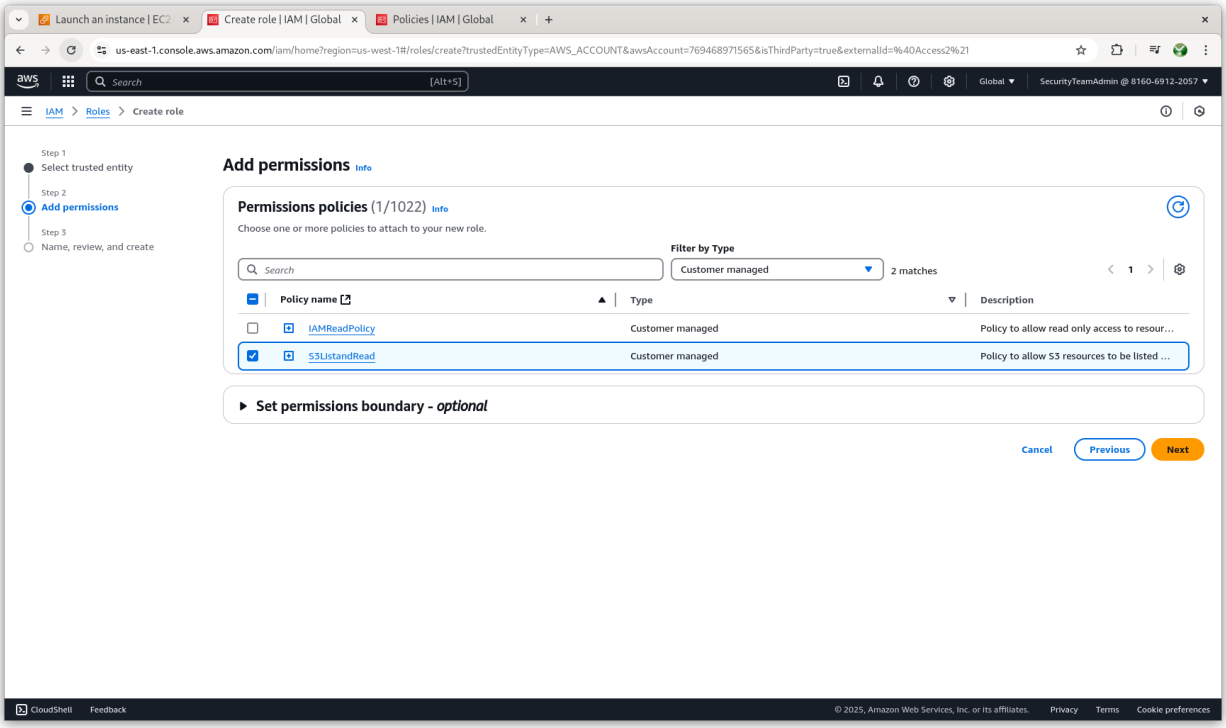
However, as shown, the role cannot upload files to S3.

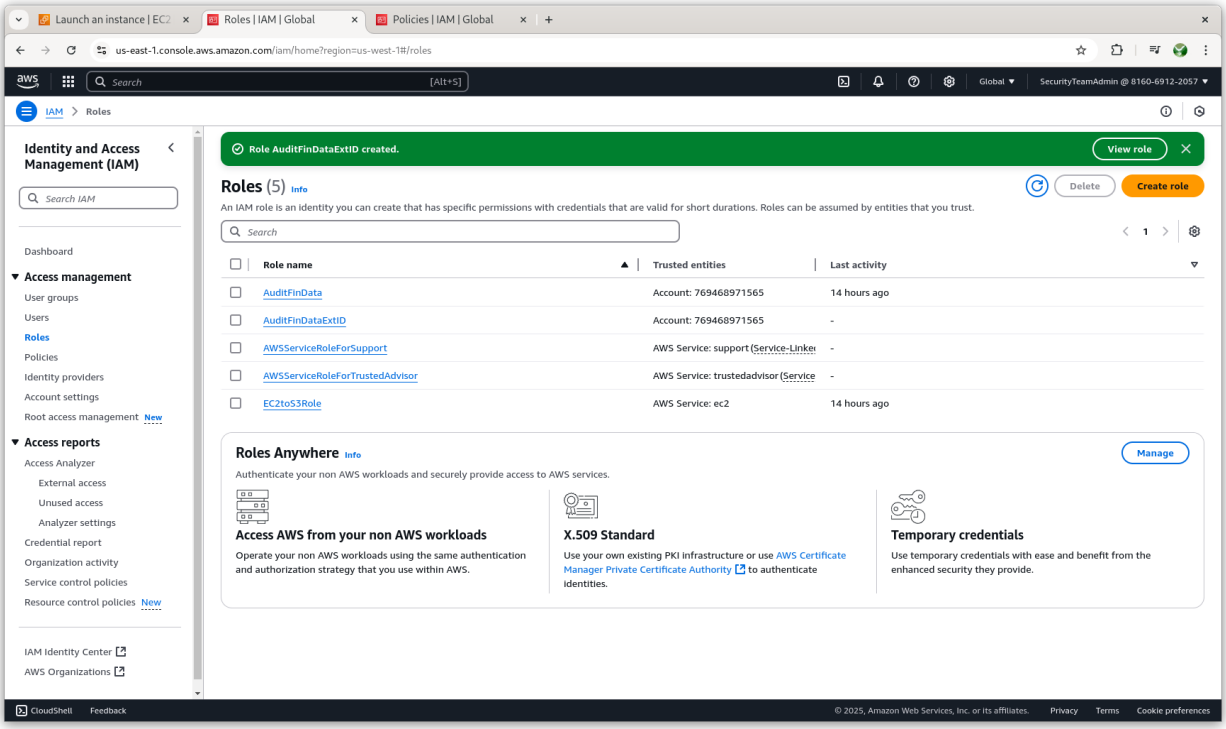


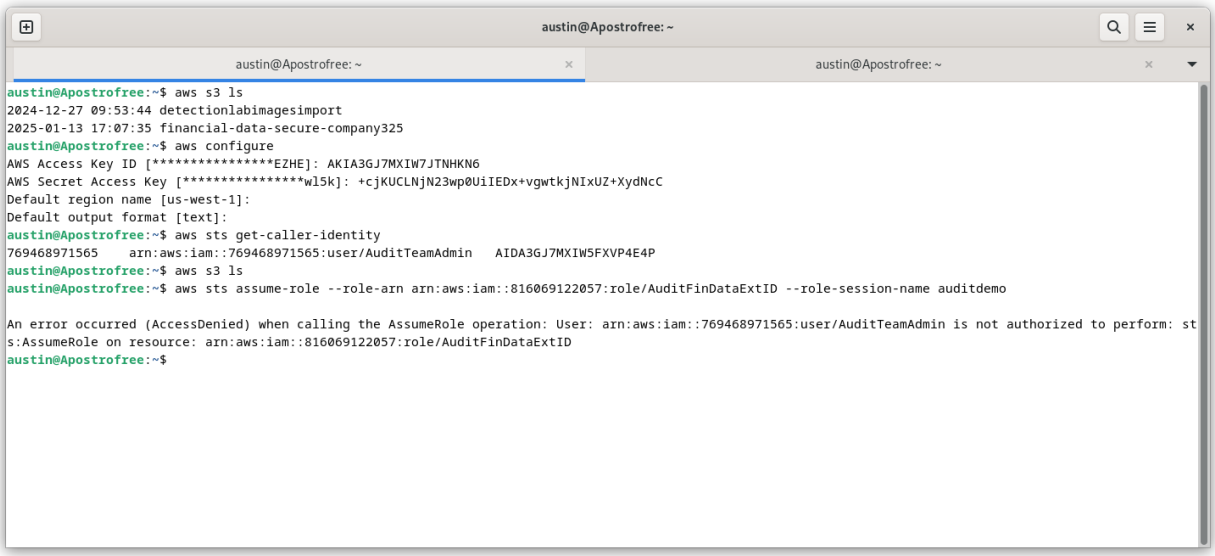
Next we create another role with an External ID set that will need to be provided when assuming the role. This is considered best practice when providing access to a third party.


Launch an instance | EC2
Create role | IAM | Global
Policies | IAM | Global
us-east-1.console.aws.amazon.com/iam/home?region=us-west-1#/roles/create?trustedEntityType=AWS_ACCOUNT&awsAccount=769468971565&isThirdParty=true&externalId=%40Access2%21
Search [Alt+S]
Global
SecurityTeamAdmin @ 8160-6912-2057
IAM > Roles > Create role
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.
An AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.
This account (816069122057)
Another AWS account
Account ID
Identifier of the account that can use this role
769468971565
Account ID is a 12-digit number.
Options
Require external ID (Best practice when a third party will assume this role)
You can increase the security of your role by requiring an optional external identifier, which prevents "confused deputy" attacks. This is recommended if you do not own or have administrative access to the account that can assume this role. The external ID can include any characters that you choose. To assume this role, users must be in the trusted account and provide this exact external ID. Learn more
External ID
@Access2!
Important: The console does not support using an external ID with the Switch Role feature. If you select this option, entities in the trusted account must use the API, CLI, or a custom federation proxy to make cross-account iam:AssumeRole calls. Learn more
Require MFA
Requires that the assuming entity use multi-factor authentication.
Cancel
Next
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

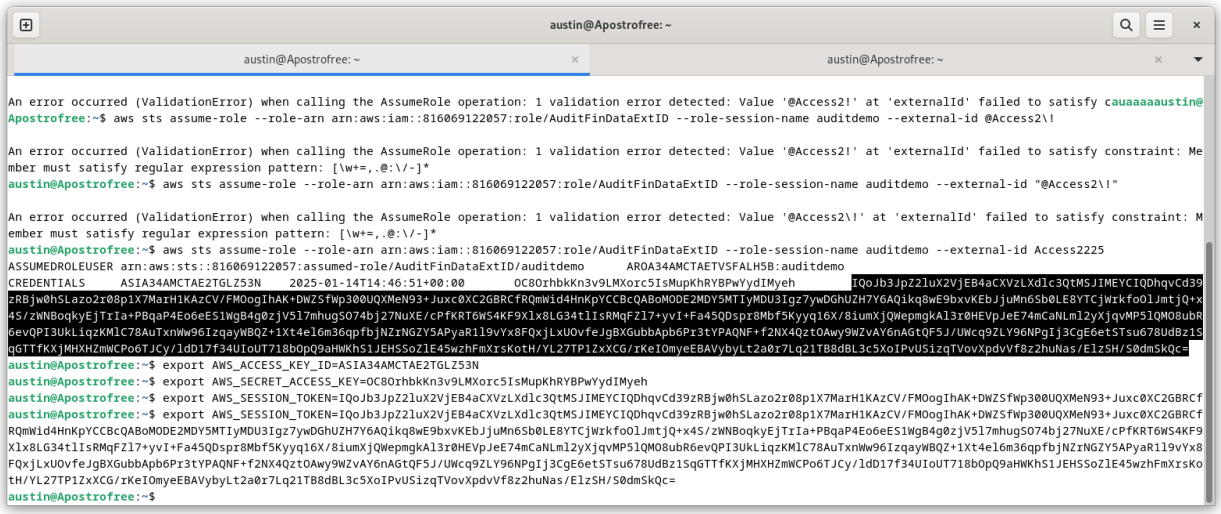

Launch an instance | EC2
Create role | IAM | Global
Policies | IAM | Global
us-east-1.console.aws.amazon.com/iam/home?region=us-west-1#/roles/create?trustedEntityType=AWS_ACCOUNT&awsAccount=769468971565&isThirdParty=true&externalId=%40Access2%21&policies=arn%3Aa...
Search [Alt+S]
Global
SecurityTeamAdmin @ 8160-6912-2057
IAM > Roles > Create role
Add permissions
Step 3
Name, review, and create
Role details
Role name
Enter a meaningful name to identify this role.
AuditFinDataExtID
Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.
Description
Add a short explanation for this role.
Allow audit team admin read-only access to financial data with an external ID.
Maximum 1000 characters. Use letters (A-Z and a-z), numbers (0-9), tabs, new lines, or any of the following characters: _+=,. @-/\[{}]!#$%^*();:"'
Step 1: Select trusted entities
Edit
Trust policy
{
"Version": "2012-10-17",
"Statement": [
{
"Effect": "Allow",
"Action": "sts:AssumeRole",
"Principal": {
"AWS": "769468971565"
},
"Condition": {
"StringEquals": {
"sts:ExternalId": "@Access2!"
}
}
}
]
}
Step 2: Add permissions
Edit
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences


Launch an instance | EC2
Create role | IAM | Global
Policies | IAM | Global
us-east-1.console.aws.amazon.com/iam/home?region=us-west-1#/roles/create?trustedEntityType=AWS_ACCOUNT&awsAccount=769468971565&isThirdParty=true&externalId=%40Access2%21
Search [Alt+S]
Global
SecurityTeamAdmin @ 8160-6912-2057
IAM > Roles > Create role
Step 1
Select trusted entity
Step 2
Add permissions
Step 3
Name, review, and create
Add permissions Info
Permissions policies (1/1022) Info
Choose one or more policies to attach to your new role.
Filter by Type
Search
Customer managed
2 matches
1
Policy name
Type
Description
IAMReadPolicy
Customer managed
Policy to allow read only access to resour...
S3ListandRead
Customer managed
Policy to allow S3 resources to be listed ...
Set permissions boundary - optional
Cancel
Previous
Next
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

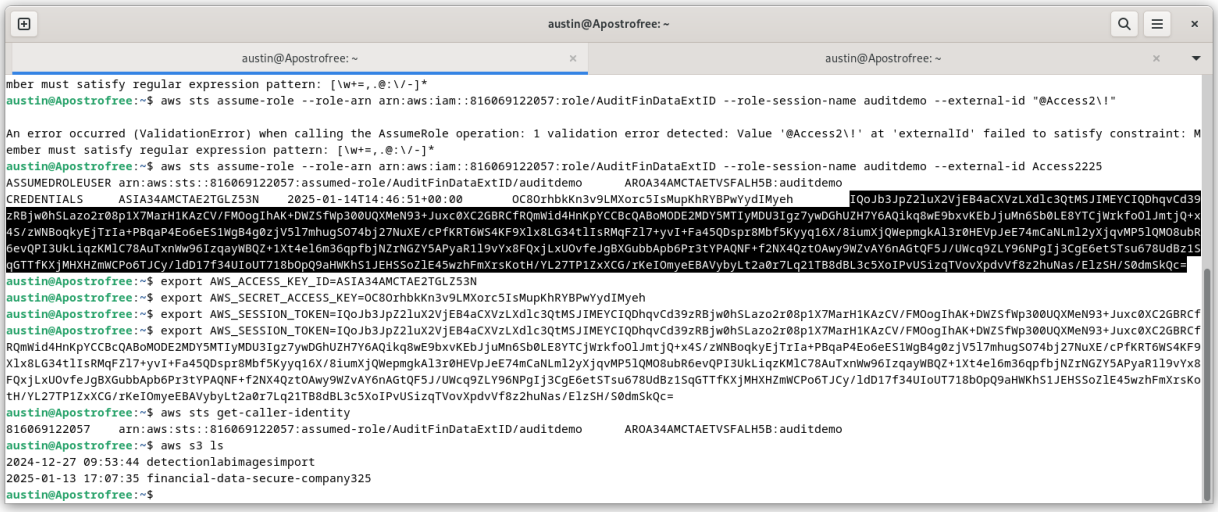

Launch an instance | EC2
Roles | IAM | Global
Policies | IAM | Global
us-east-1.console.aws.amazon.com/iam/home?region=us-west-1#/roles
Search [Alt+S]
Global
SecurityTeamAdmin @ 8160-6912-2057
IAM > Roles
Identity and Access Management (IAM)
Search IAM
Dashboard
Access management
User groups
Users
Roles
Policies
Identity providers
Account settings
Root access management New
Access reports
Access Analyzer
External access
Unused access
Analyzer settings
Credential report
Organization activity
Service control policies
Resource control policies New
IAM Identity Center
AWS Organizations
Role AuditFinDataExtID created.
View role
Roles (5) Info
An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.
Delete
Create role
Search
1
Role name
Trusted entities
Last activity
AuditFinData
Account: 769468971565
14 hours ago
AuditFinDataExtID
Account: 769468971565
-
AWSServiceRoleForSupport
AWS Service: support (Service-Linke
-
AWSServiceRoleForTrustedAdvisor
AWS Service: trustedadvisor (Service
-
EC2toS3Role
AWS Service: ec2
14 hours ago
Roles Anywhere Info
Manage
Authenticate your non AWS workloads and securely provide access to AWS services.
Access AWS from your non AWS workloads
Operate your non AWS workloads using the same authentication and authorization strategy that you use within AWS.
X.509 Standard
Use your own existing PKI infrastructure or use AWS Certificate Manager Private Certificate Authority to authenticate identities.
Temporary credentials
Use temporary credentials with ease and benefit from the enhanced security they provide.
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences



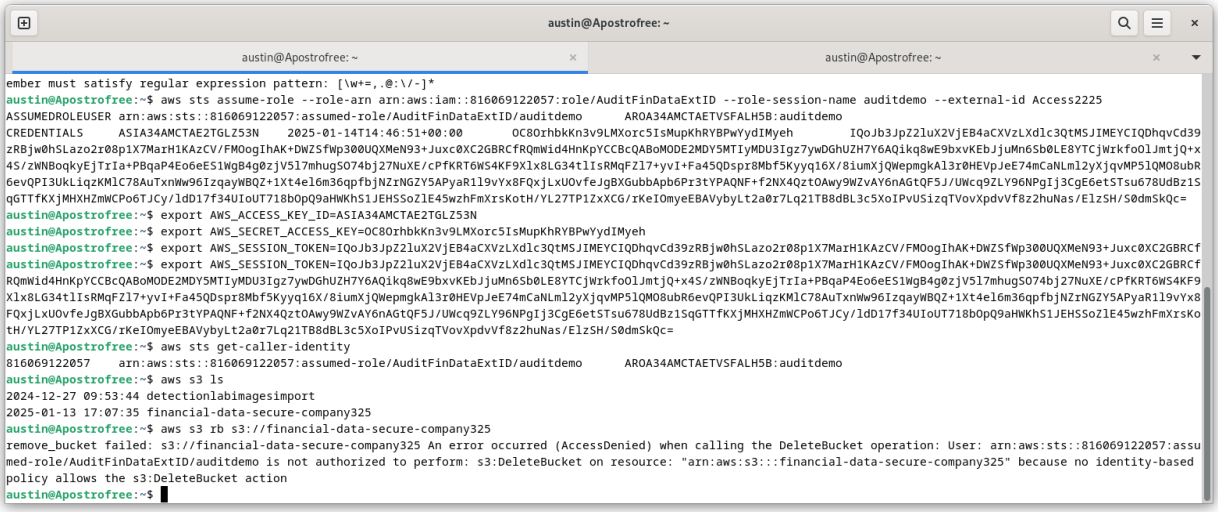
After setting the appropriate access tokens for the auditing team as shown by *aws sts get-caller-identity* in the AWS CLI we first attempt to assume the newly created role without the external ID and are met with an Access Denied error.



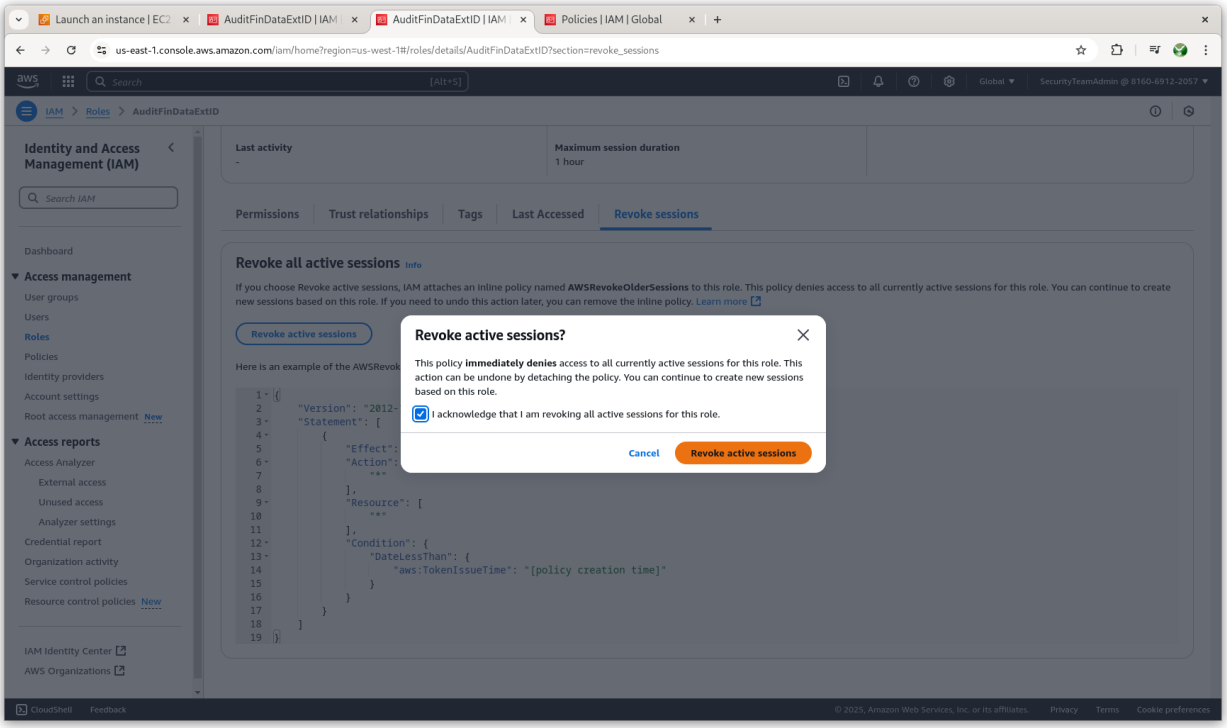
When we try again with the - -external-id set we are provided with the appropriate credentials. We can then use export to set these corresponding variables and use the AWS CLI as the IAM role we created.



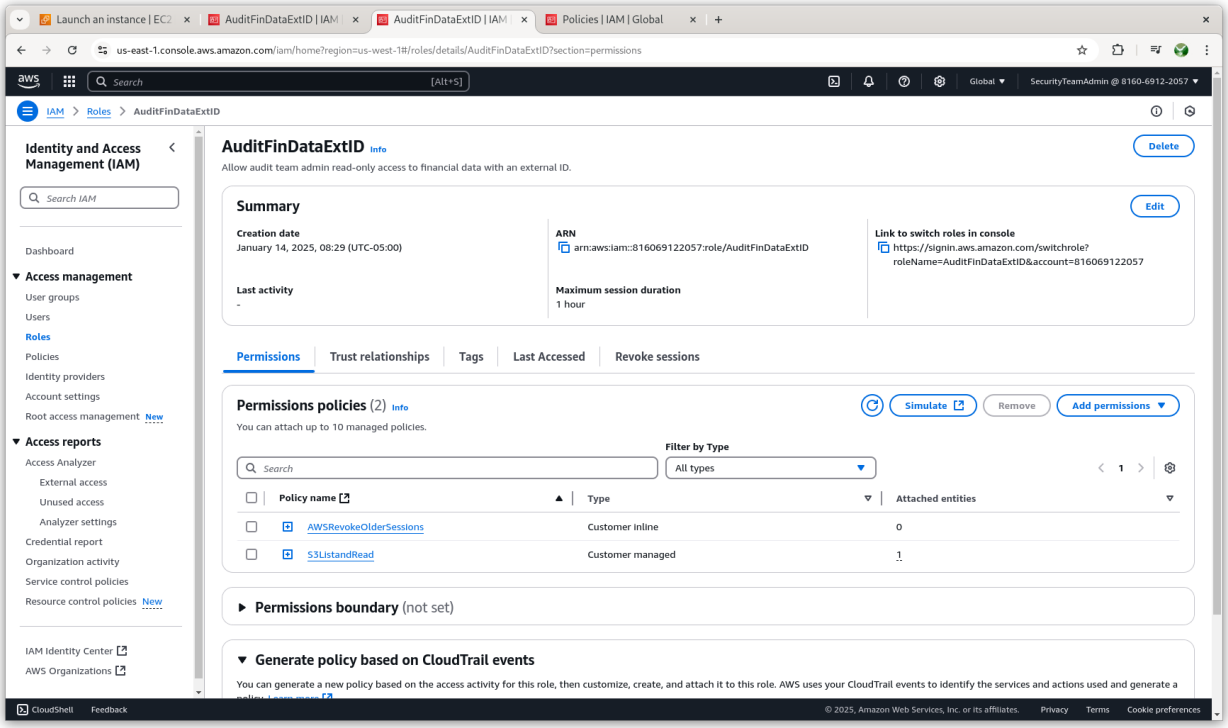
Here we see that the role has been assumed using *aws sts get-caller-identity* and we see that the auditing team is now able to perform *aws s3 ls* and list the buckets. Read only access to S3 has been provided.

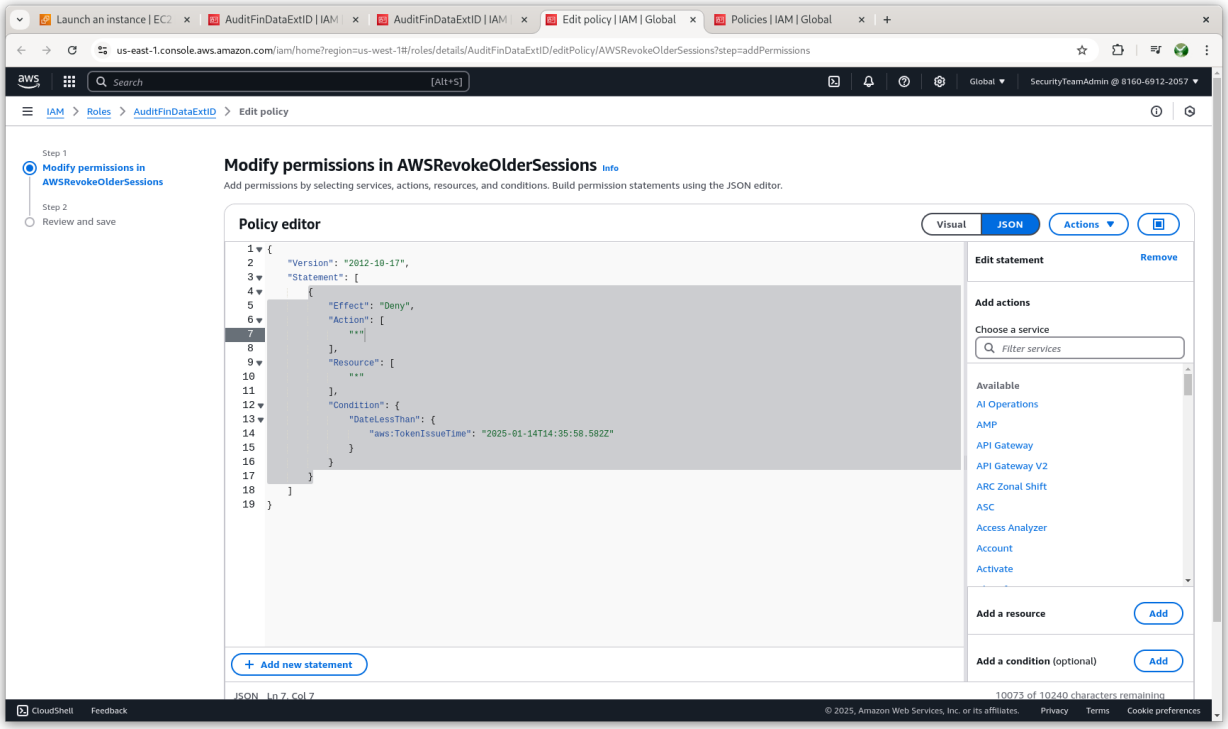


Above we show that the auditing team role is unable to delete S3 resources as *aws s3 rb s3://financial-data-secure-company325 returns a not authorized to perform error*.



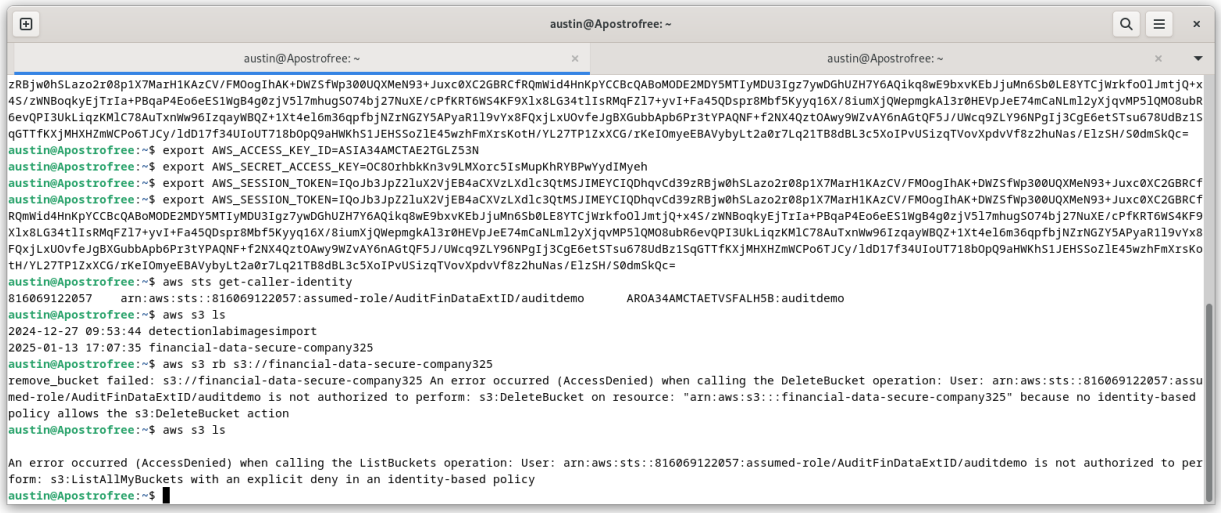
Next we revoke all active sessions under IAM > Access Management > Roles this immediately revokes the access tokens, useful for instance when credentials have been leaked or stolen.



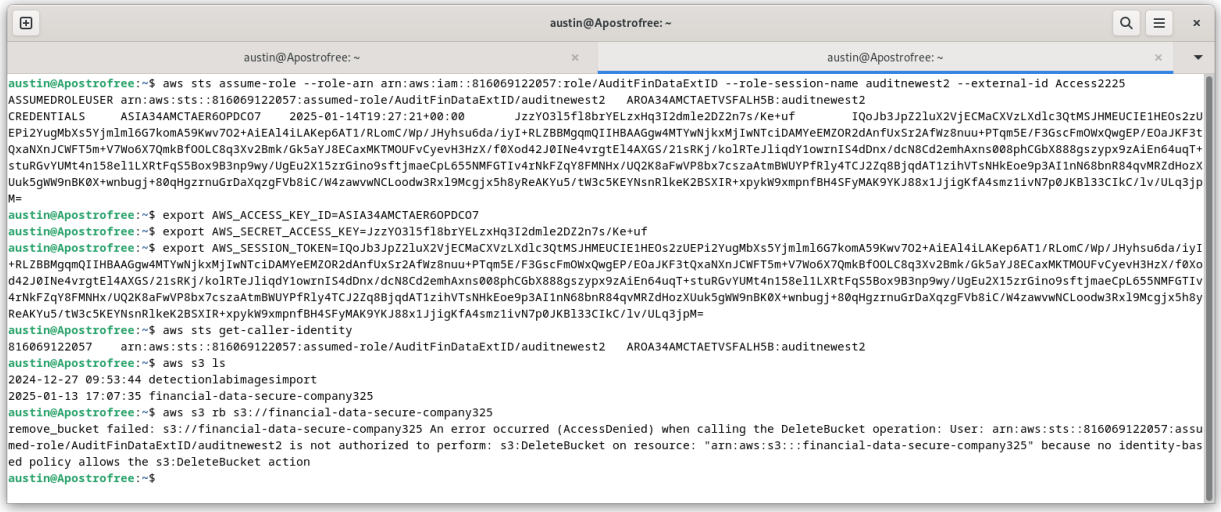


This creates the permission policy shown.

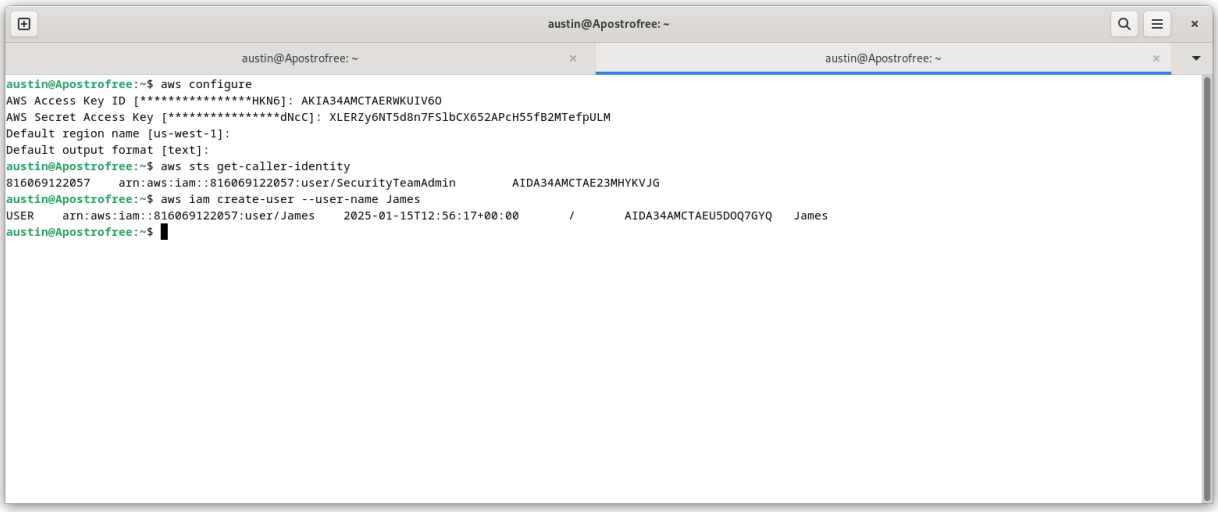
A



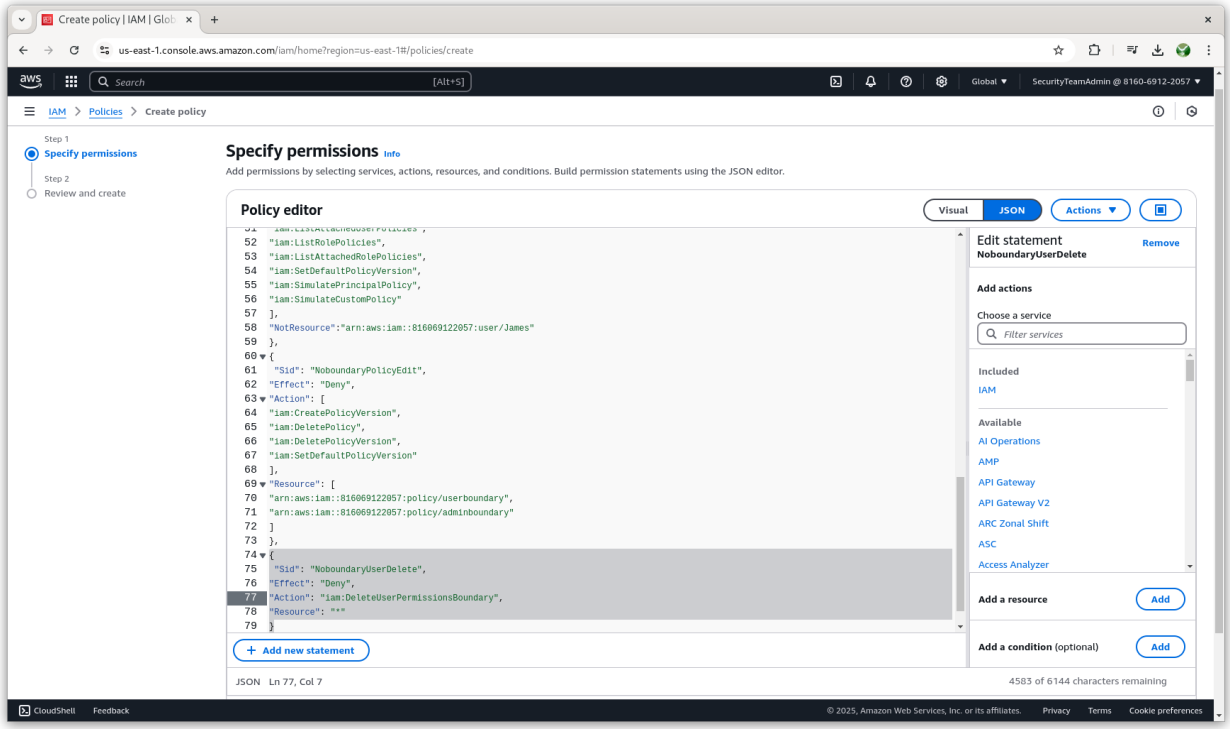
As seen above the Auditing Team using the credentials that were revoked is no longer able to list or read S3 resources!



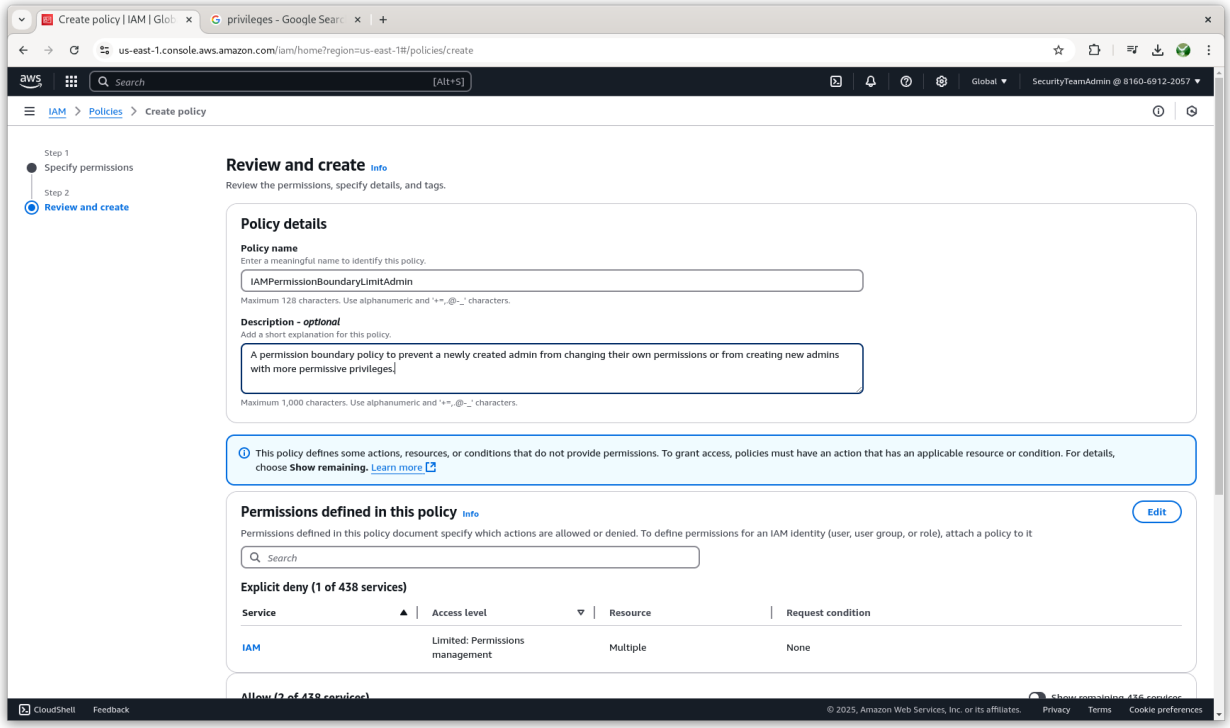
However the role can be assumed again now with the new access tokens and the same permissions provided to the role are in place once again, however the leaked or stolen credentials remain invalid.



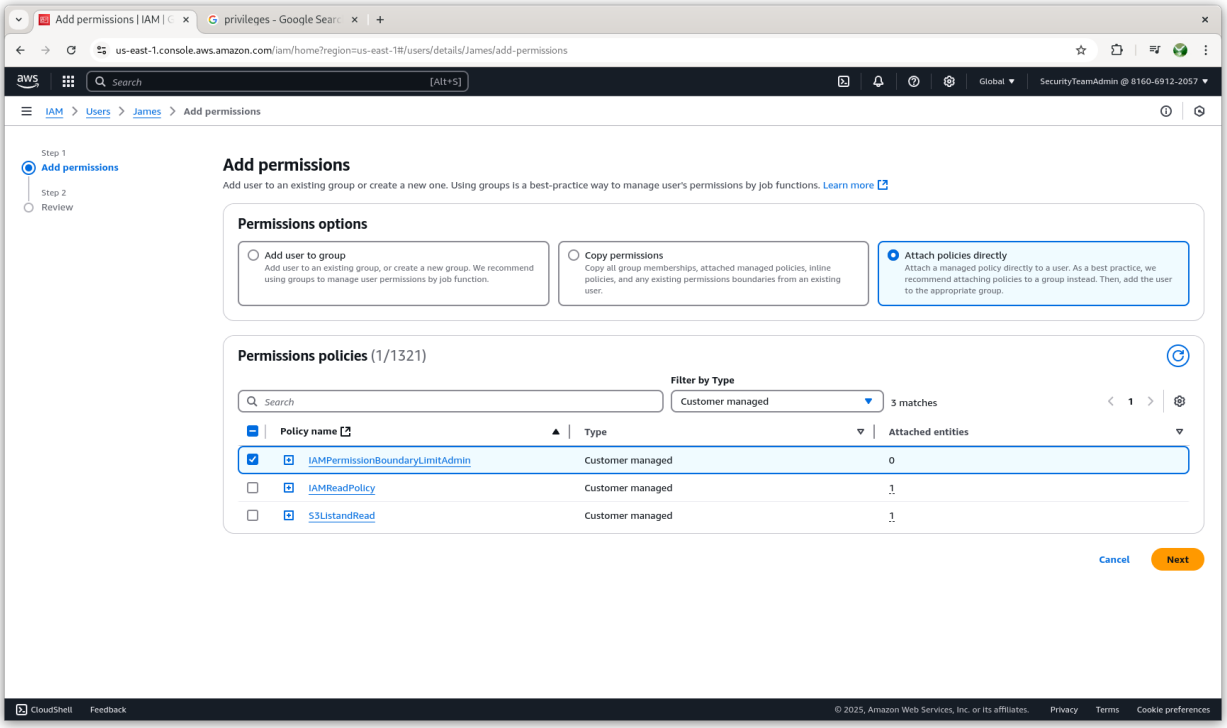
Next we switch back to the SecurityTeamAdmin IAM user and create a new user James with *aws iam create-user --user-name James* to show the permissions of SecurityTeamAdmin are assumed.

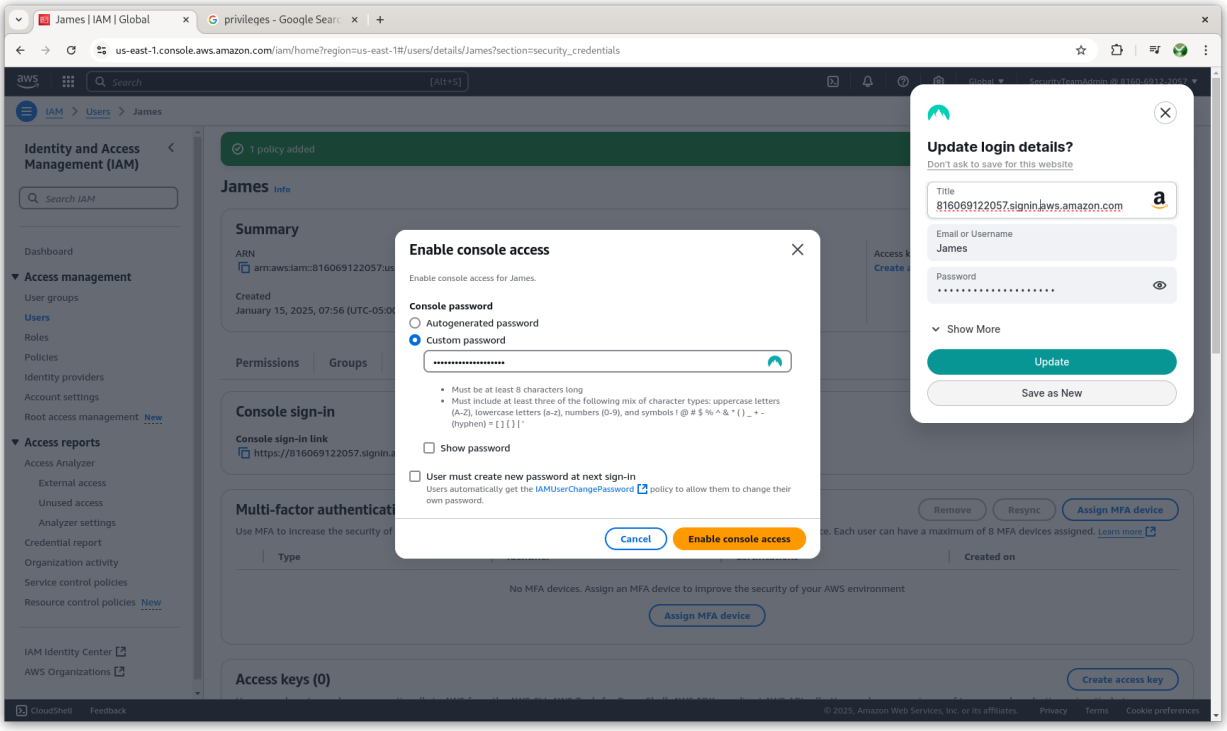


Next we create a new policy with a permission boundary, a permission boundary explicitly denies an action overriding other more priveleged permissions in a policy. This permission boundary is applied to our new administrator James and prevents an admin from changing their own permissions, or other admins permissions, or from creating new admins with more privileges.

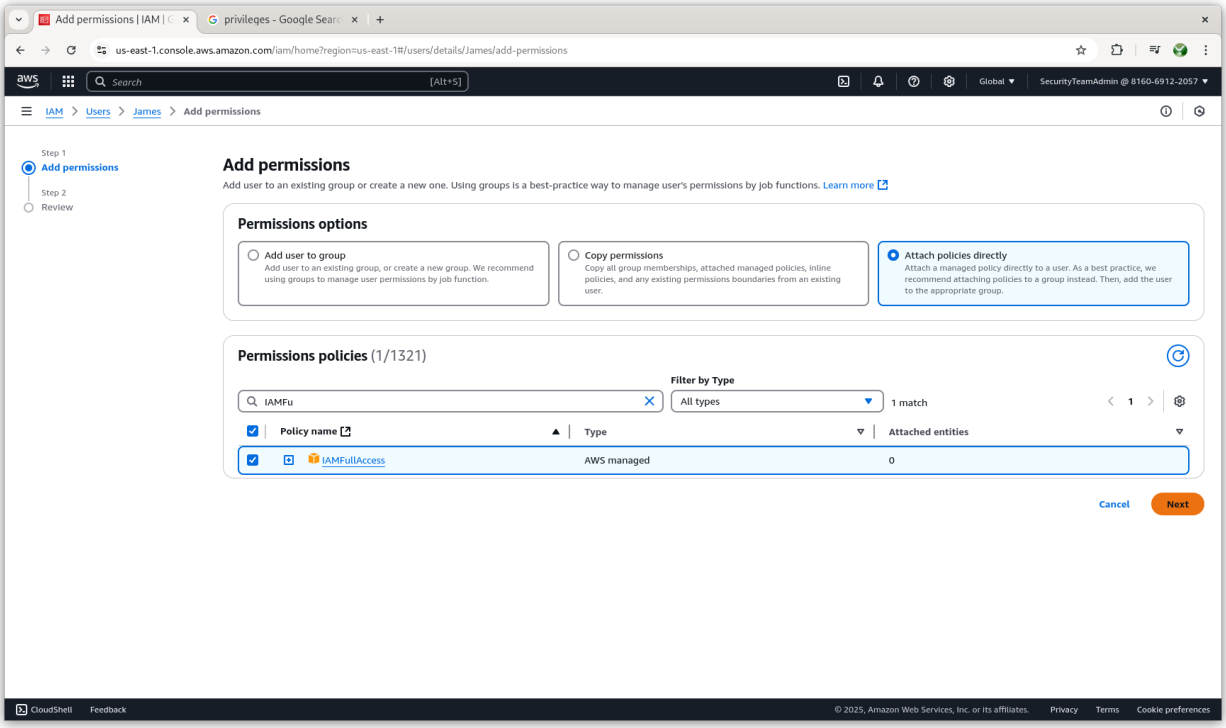


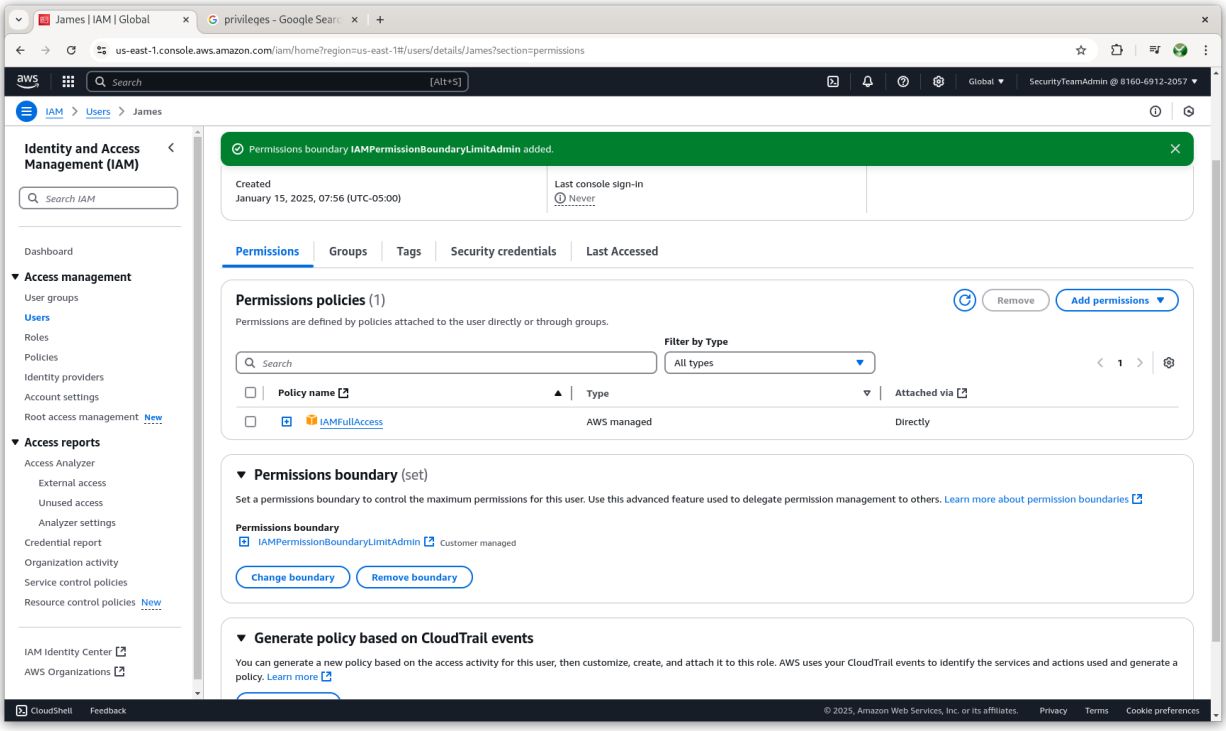
We name the permission boundary IAMPermissionBoundaryLimitAdmin and save the permission



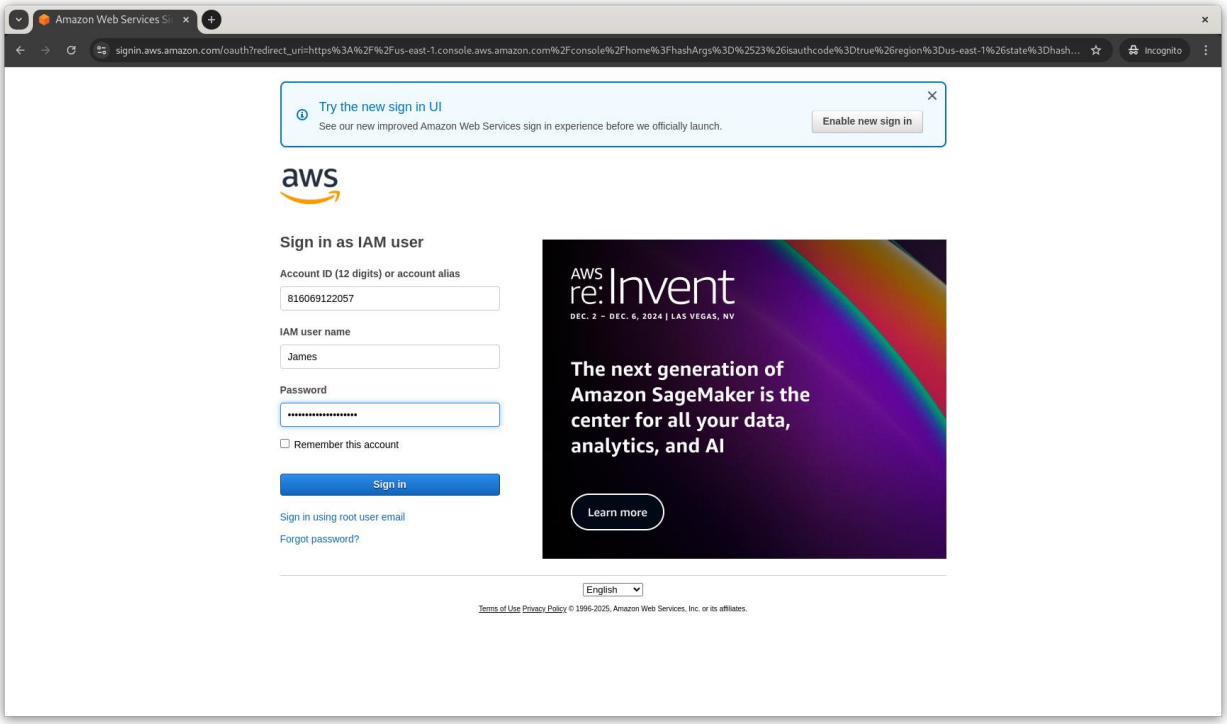


Next we attach the policy to James and enable console access for James our new admin.

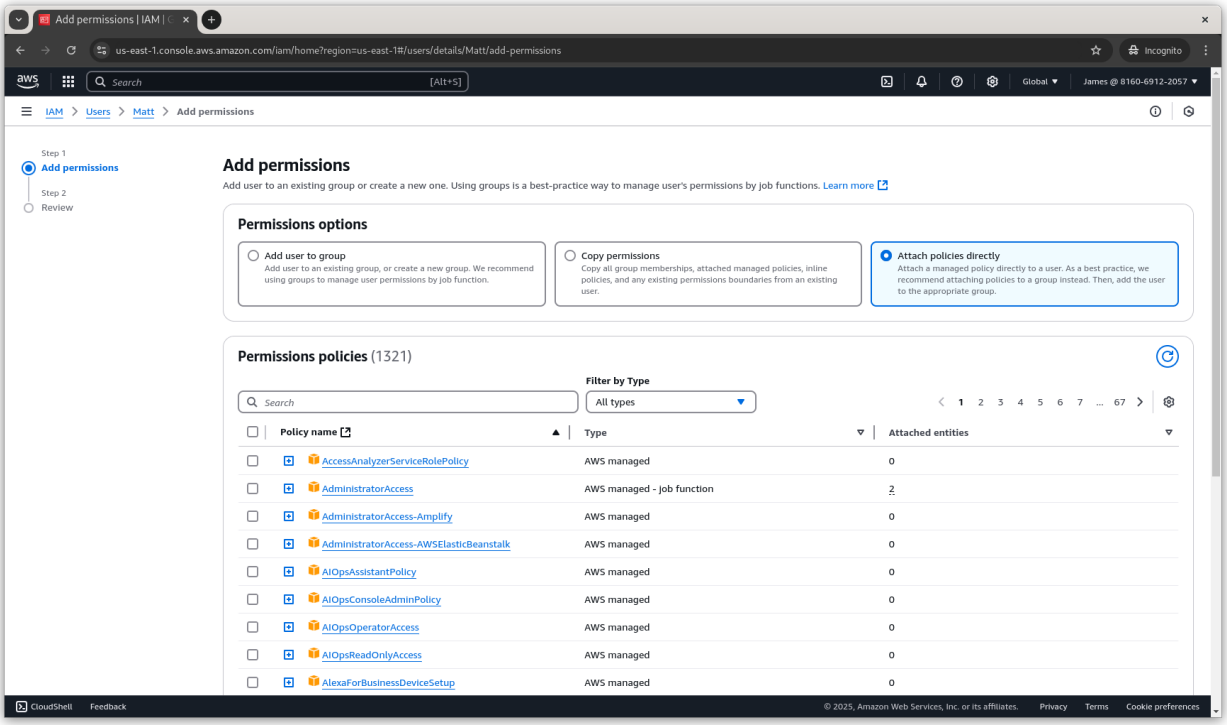




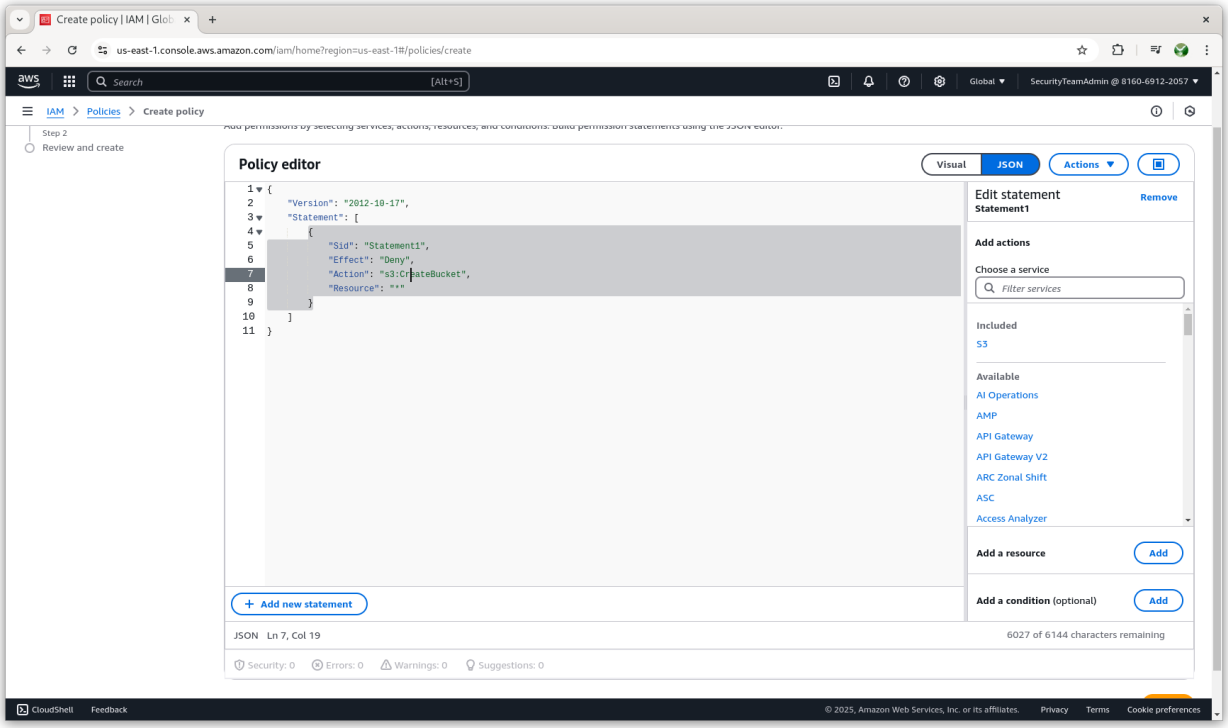
The admin James now has the permission policy IAMFullAccess as well as the permission boundary we created to limit admin permissions.

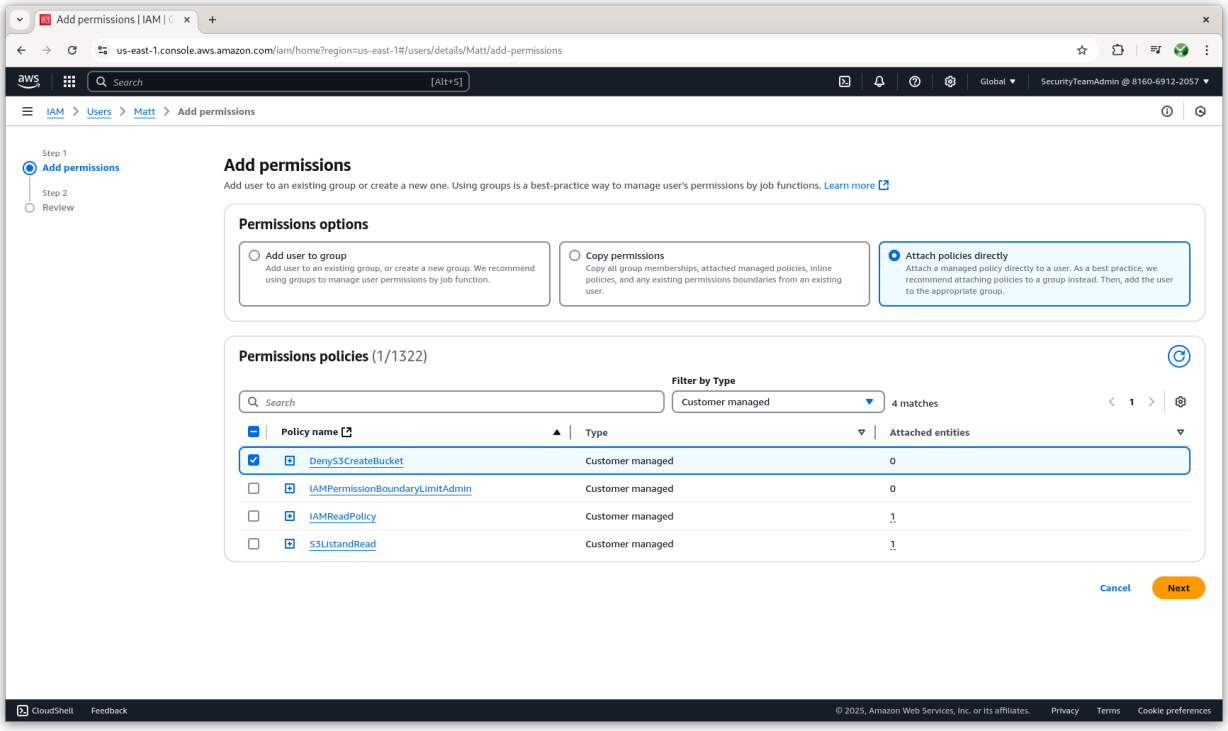


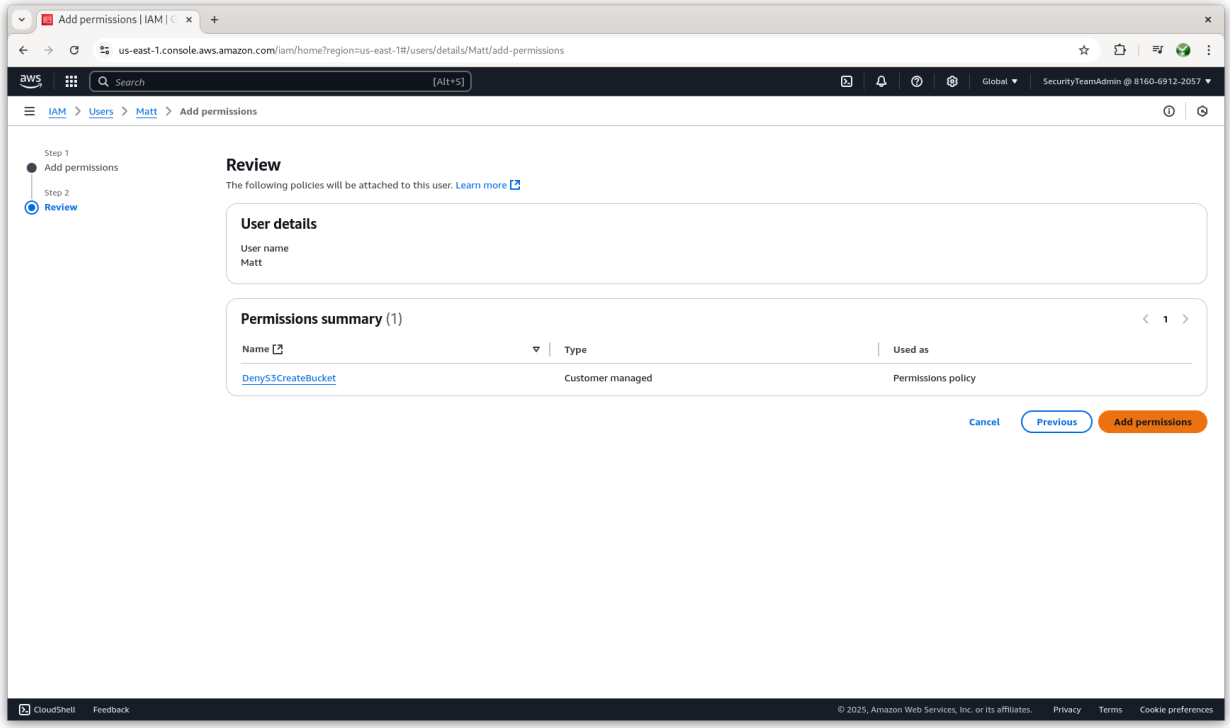
We log into the console as James.



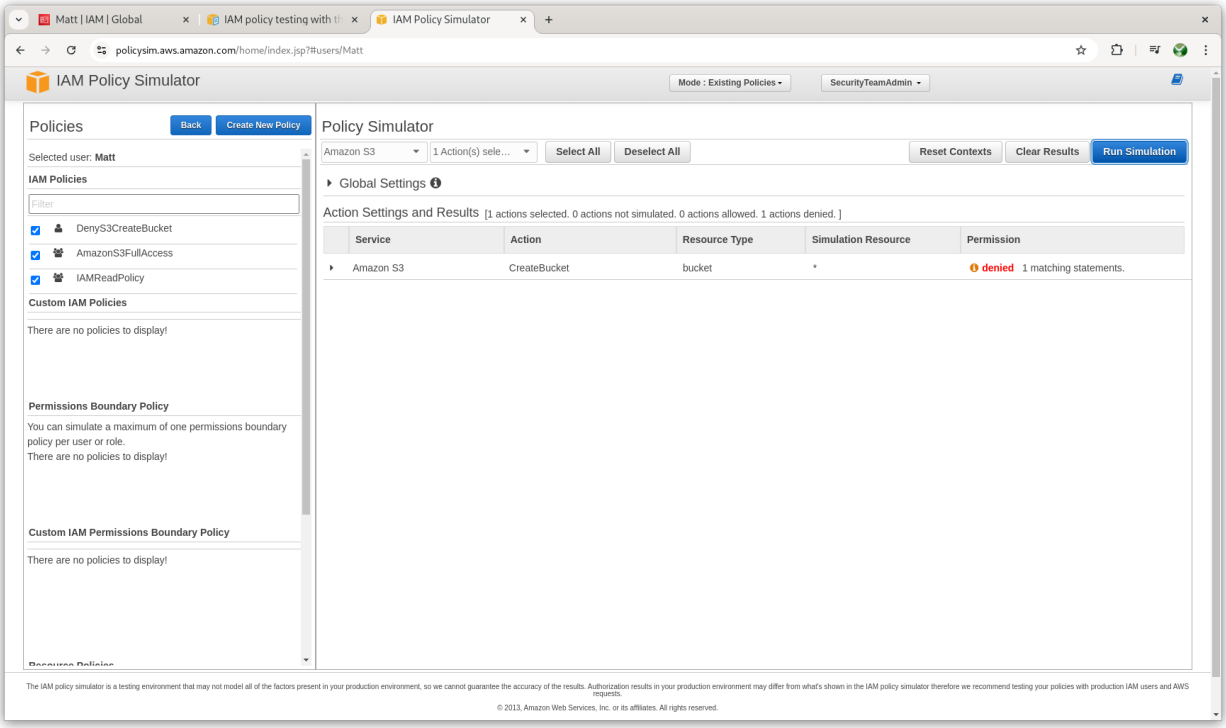
James is unable to modify his own permissions or those of other administrators.

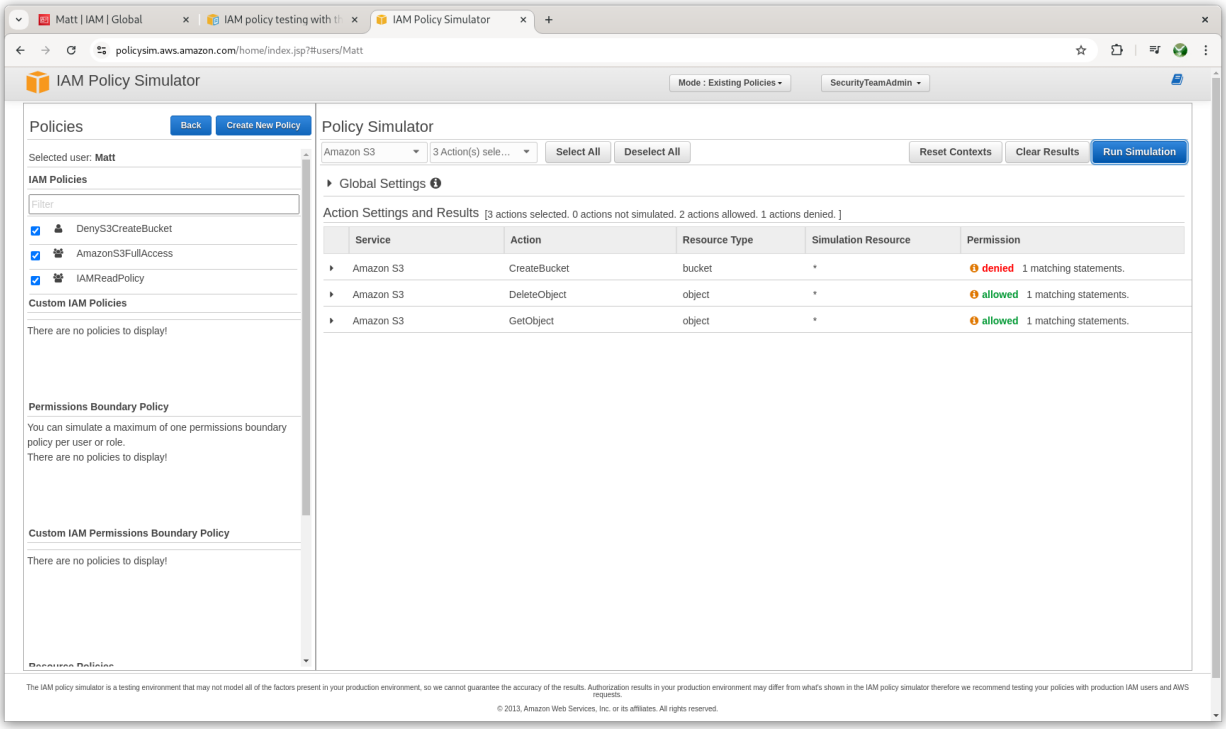

Create policy | IAM | Glob
us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/policies/create
Search [Alt+S]
Global
SecurityTeamAdmin @ 8160-6912-2057
IAM > Policies > Create policy
Step 2
Review and create
Policy editor
Visual
JSON
Actions
{
"Version": "2012-10-17",
"Statement": [
{
"Sid": "Statement1",
"Effect": "Deny",
"Action": "s3:CreateBucket",
"Resource": "*"
}
]
}
Edit statement
Statement1
Remove
Add actions
Choose a service
Filter services
Included
S3
Available
AI Operations
AMP
API Gateway
API Gateway V2
ARC Zonal Shift
ASC
Access Analyzer
Add a resource
Add
Add a condition (optional)
Add
Add new statement
JSON Ln 7, Col 19
6027 of 6144 characters remaining
Security: 0
Errors: 0
Warnings: 0
Suggestions: 0
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

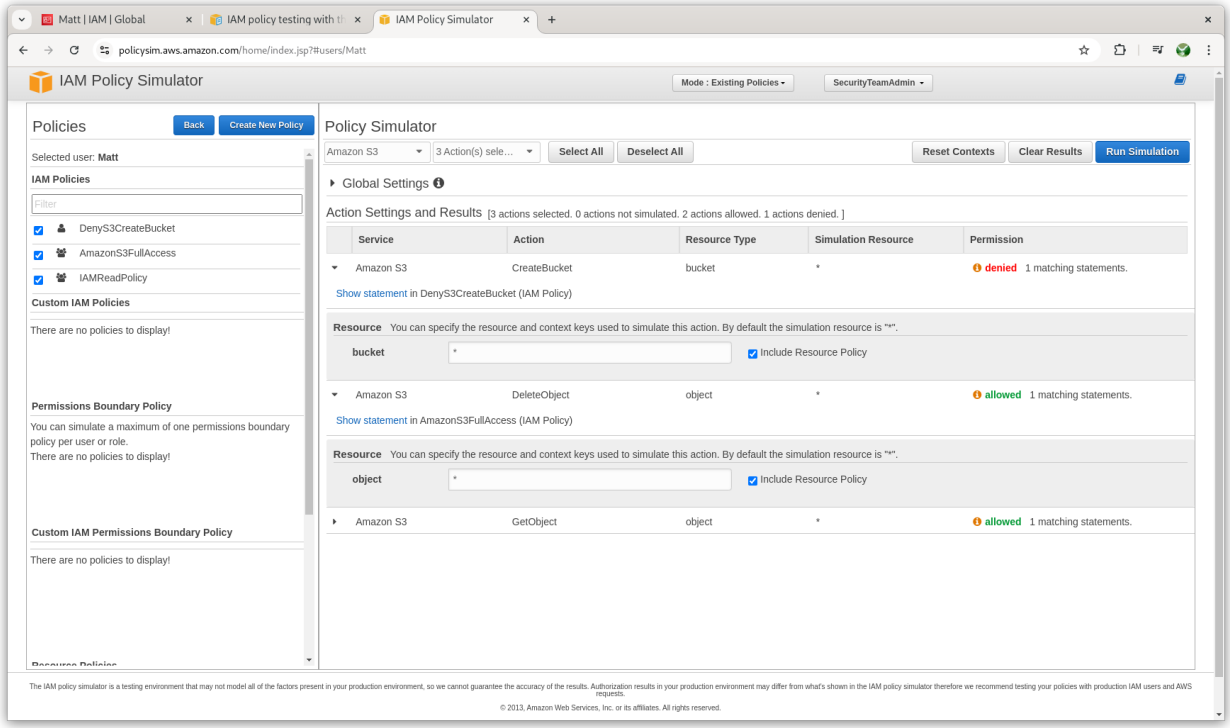

Add permissions | IAM |
us-east-1.console.aws.amazon.com/iam/home?region=us-east-1#/users/details/Matt/add-permissions
Search [Alt+S]
Global
SecurityTeamAdmin @ 8160-6912-2057
IAM > Users > Matt > Add permissions
Step 1
Add permissions
Step 2
Review
Add permissions
Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. Learn more
Permissions options
Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.
Copy permissions
Copy all group memberships, attached managed policies, inline policies, and any existing permissions boundaries from an existing user.
Attach policies directly
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.
Permissions policies (1/1322)
Filter by Type
Search
Customer managed
4 matches
1
Policy name
Type
Attached entities
DenyS3CreateBucket
Customer managed
0
IAMPermissionBoundaryLimitAdmin
Customer managed
0
IAMReadPolicy
Customer managed
1
S3ListandRead
Customer managed
1
Cancel
Next
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

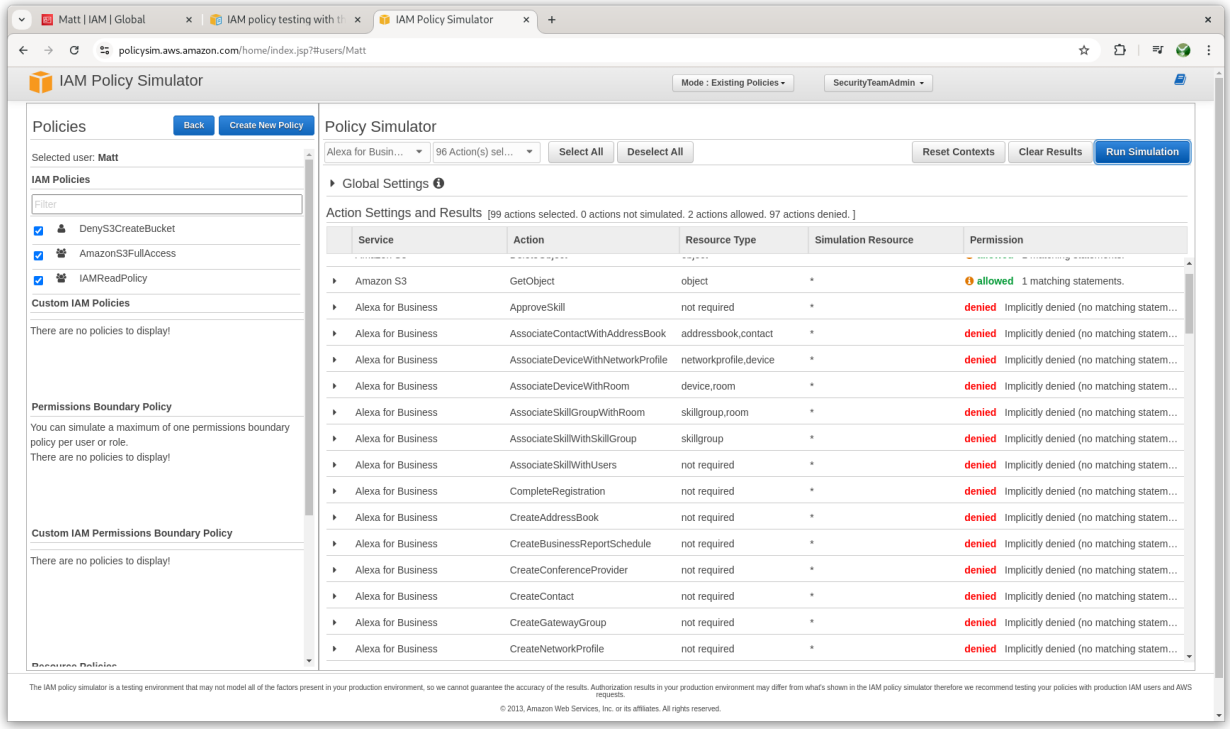


Now we attach a permission policy denying the creation of S3 buckets and attach it to the user Matt.


Matt | IAM | Global
IAM policy testing with th
IAM Policy Simulator
policysim.aws.amazon.com/home/index.jsp?#users/Matt
IAM Policy Simulator
Mode : Existing Policies
SecurityTeamAdmin
Policies
Back
Create New Policy
Selected user: Matt
IAM Policies
Filter
DenyS3CreateBucket
AmazonS3FullAccess
IAMReadPolicy
Custom IAM Policies
There are no policies to display!
Permissions Boundary Policy
You can simulate a maximum of one permissions boundary policy per user or role.
There are no policies to display!
Custom IAM Permissions Boundary Policy
There are no policies to display!
Policy Simulator
Amazon S3
1 Action(s) sele…
Select All
Deselect All
Reset Contexts
Clear Results
Run Simulation
Global Settings
Action Settings and Results [1 actions selected. 0 actions not simulated. 0 actions allowed. 1 actions denied.]
Service
Action
Resource Type
Simulation Resource
Permission
Amazon S3
CreateBucket
bucket
*
denied 1 matching statements.
The IAM policy simulator is a testing environment that may not model all of the factors present in your production environment, so we cannot guarantee the accuracy of the results. Authorization results in your production environment may differ from what's shown in the IAM policy simulator therefore we recommend testing your policies with production IAM users and AWS requests.
© 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved.

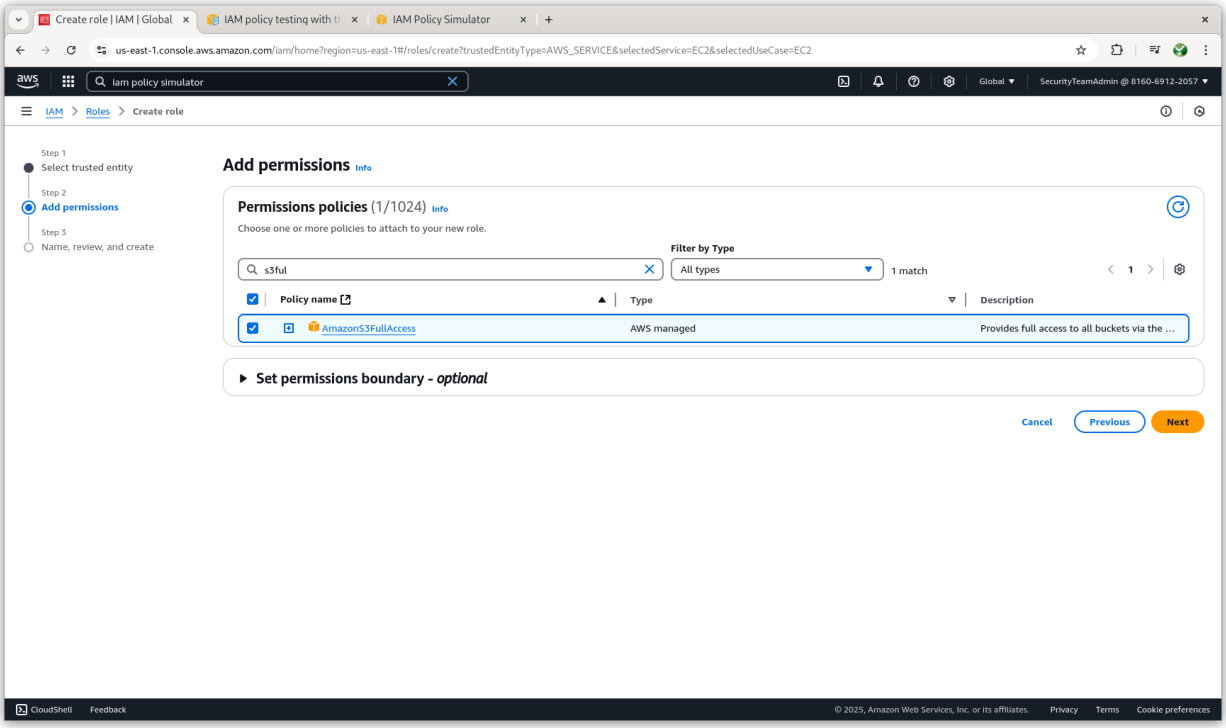

Matt | IAM | Global
IAM policy testing with th
IAM Policy Simulator
policysim.aws.amazon.com/home/index.jsp?#users/Matt
IAM Policy Simulator
Mode : Existing Policies
SecurityTeamAdmin
Policies
Back
Create New Policy
Selected user: Matt
IAM Policies
Filter
DenyS3CreateBucket
AmazonS3FullAccess
IAMReadPolicy
Custom IAM Policies
There are no policies to display!
Permissions Boundary Policy
You can simulate a maximum of one permissions boundary policy per user or role.
There are no policies to display!
Custom IAM Permissions Boundary Policy
There are no policies to display!
Policy Simulator
Amazon S3
3 Action(s) sele…
Select All
Deselect All
Reset Contexts
Clear Results
Run Simulation
Global Settings
Action Settings and Results [3 actions selected. 0 actions not simulated. 2 actions allowed. 1 actions denied.]
Service
Action
Resource Type
Simulation Resource
Permission
Amazon S3
CreateBucket
bucket
*
denied 1 matching statements.
Amazon S3
DeleteObject
object
*
allowed 1 matching statements.
Amazon S3
GetObject
object
*
allowed 1 matching statements.
The IAM policy simulator is a testing environment that may not model all of the factors present in your production environment, so we cannot guarantee the accuracy of the results. Authorization results in your production environment may differ from what's shown in the IAM policy simulator therefore we recommend testing your policies with production IAM users and AWS requests.
© 2013, Amazon Web Services, Inc. or its affiliates. All rights reserved.

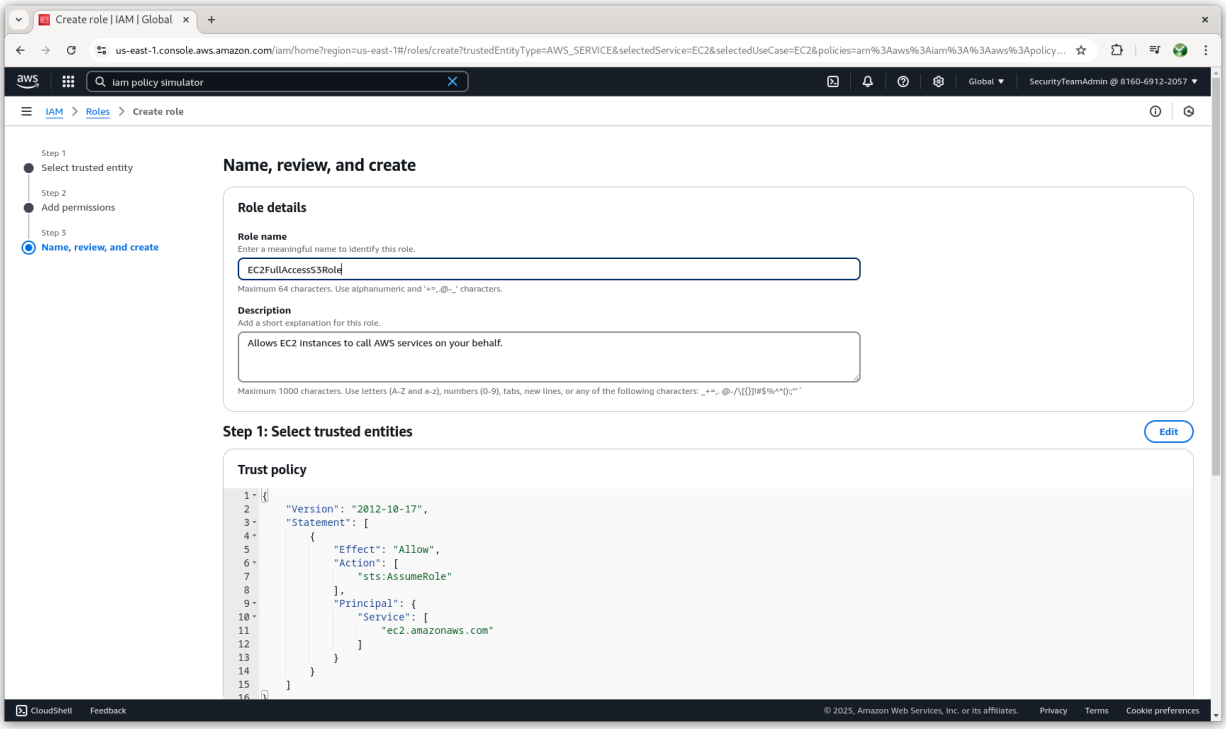




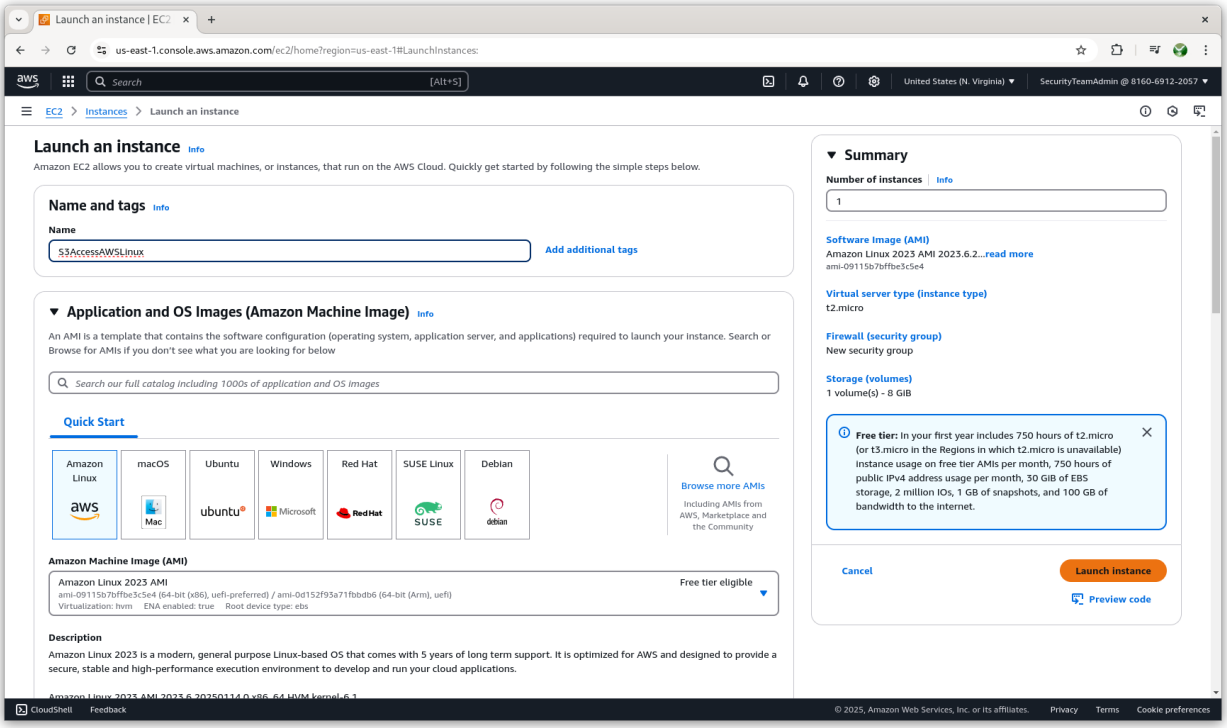
Using the policy simulator we can troubleshoot IAM policies and see that:

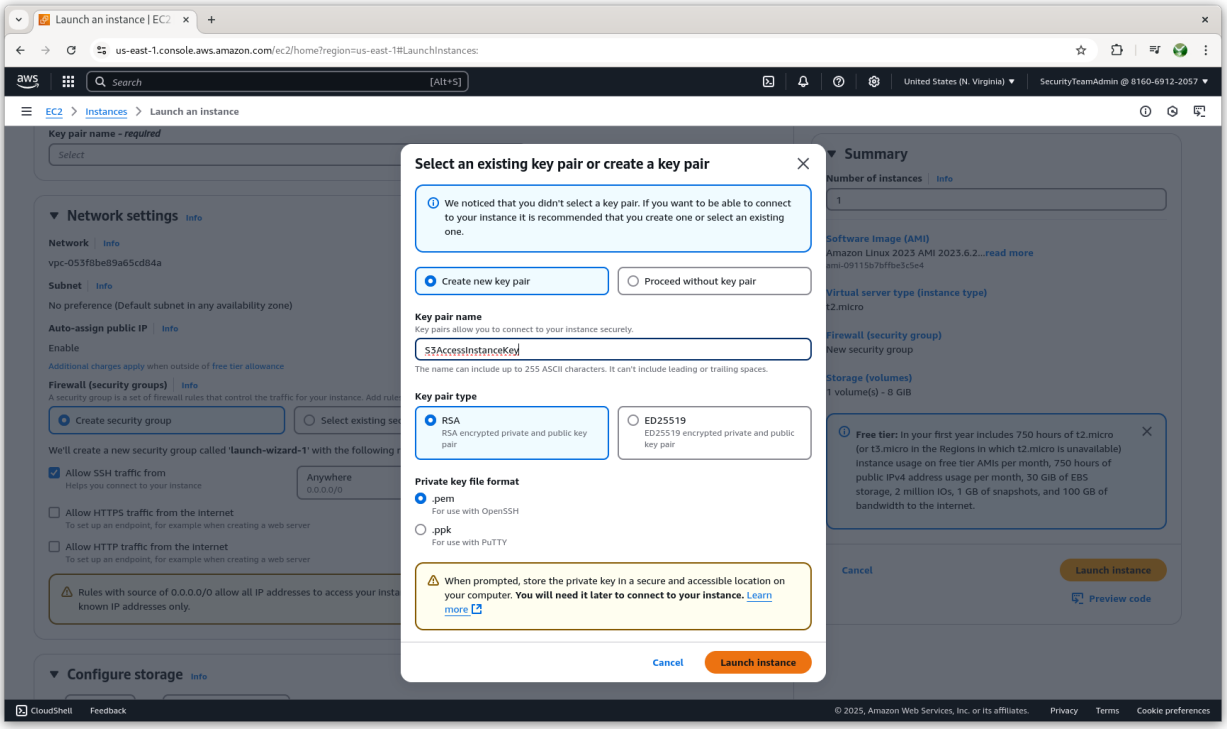
In IAM the Deny action overrides the Allow action and if you do not have explicit permission to access AWS resources or perform an action, you are automatically denied permission to do so.

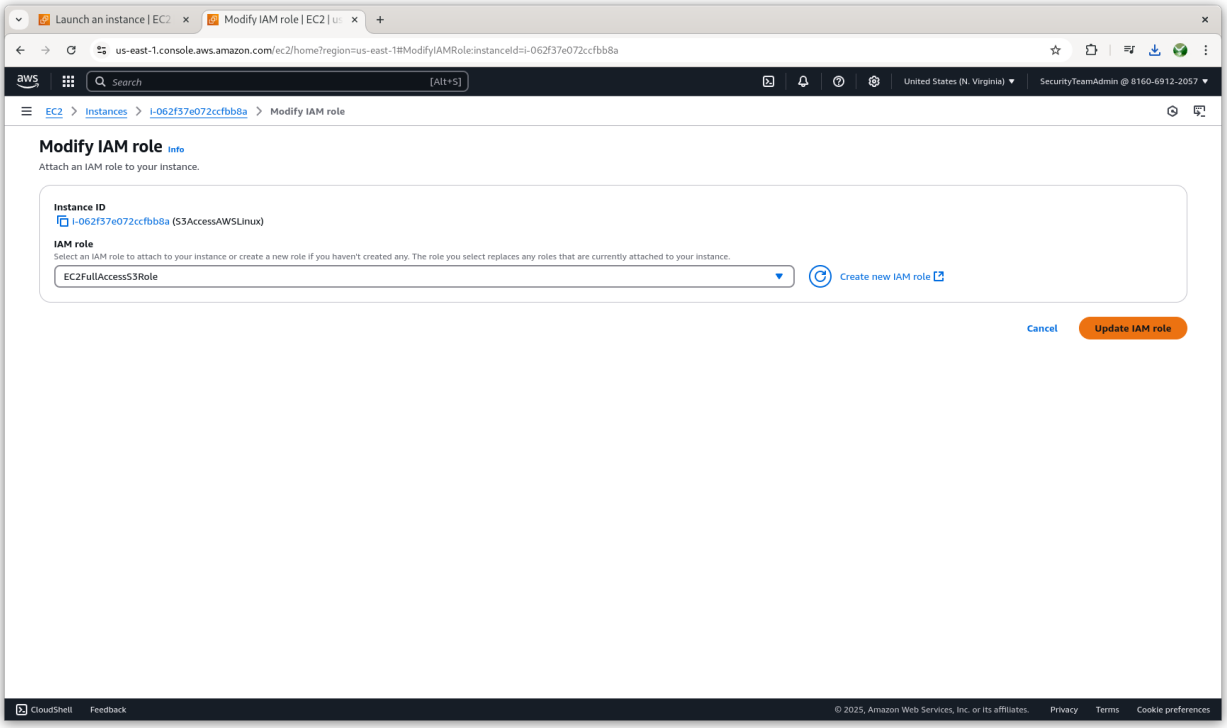




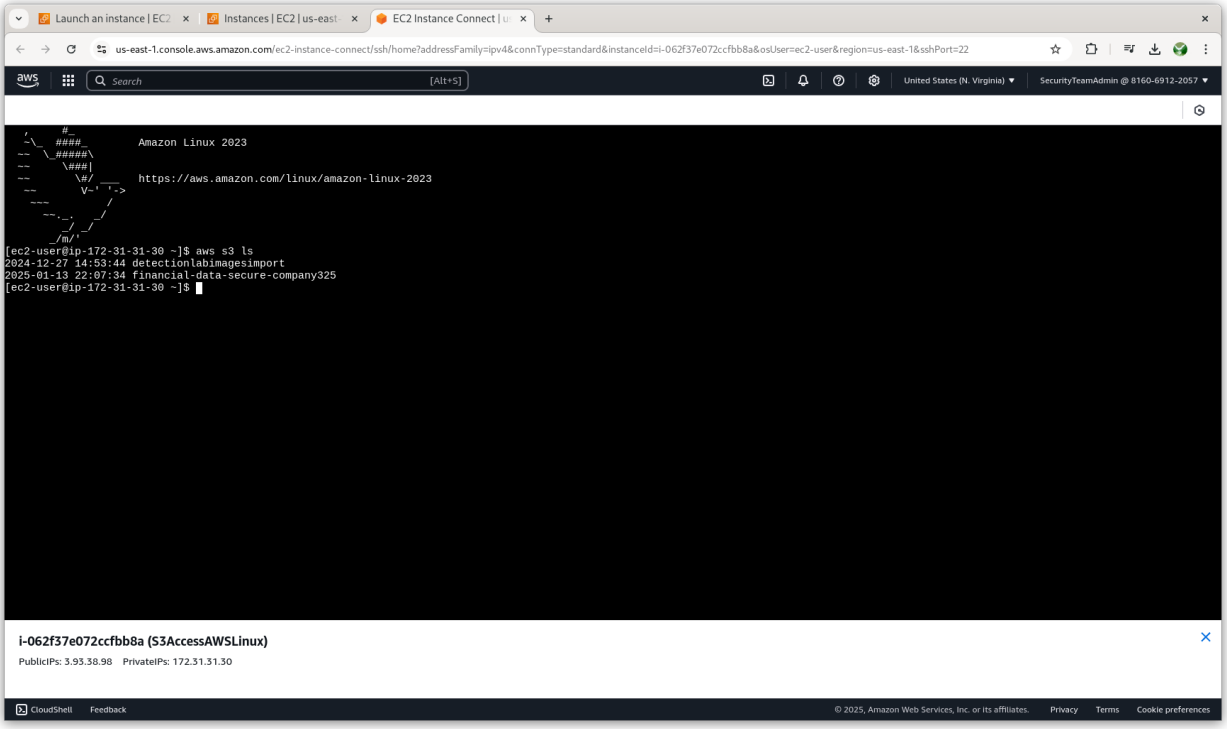
Next, we create an IAM role to grant full S3 access to an EC2 virtual machine.


Launch an instance | EC2
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:
Search [Alt+S]
United States (N. Virginia)
SecurityTeamAdmin @ 8160-6912-2057
EC2 > Instances > Launch an instance
Launch an instance Info
Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.
Name and tags Info
Name
S3AccessAWSLinux
Add additional tags
Application and OS Images (Amazon Machine Image) Info
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below
Search our full catalog including 1000s of application and OS images
Quick Start
Amazon Linux
macOS
Ubuntu
Windows
Red Hat
SUSE Linux
Debian
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
Amazon Machine Image (AMI)
Amazon Linux 2023 AMI
ami-09115b7bffbe3c5e4 (64-bit (x86), uefi-preferred) / ami-0d152f93a71fbbdb6 (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs
Free tier eligible
Description
Amazon Linux 2023 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.
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.6.2...read more
ami-09115b7bffbe3c5e4
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance
Preview code
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

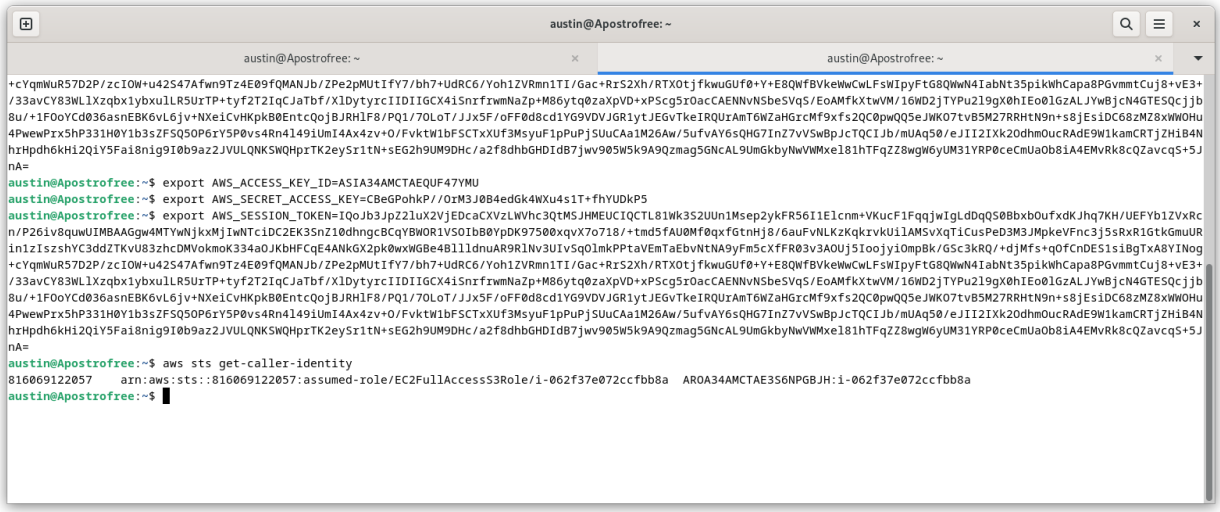

Launch an instance | EC2
us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:
Search [Alt+S]
United States (N. Virginia)
SecurityTeamAdmin @ 8160-6912-2057
EC2 > Instances > Launch an instance
Key pair name - required
Select
Network settings Info
Network Info
vpc-053f8be89a65cd84a
Subnet Info
No preference (Default subnet in any availability zone)
Auto-assign public IP Info
Enable
Additional charges apply when outside of free tier allowance
Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules
Create security group
Select existing se
We'll create a new security group called 'launch-wizard-1' with the following
Allow SSH traffic from
Helps you connect to your instance
Anywhere
0.0.0.0/0
Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server
Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server
Rules with source of 0.0.0.0/0 allow all IP addresses to access your insta
known IP addresses only.
Configure storage Info
Select an existing key pair or create a key pair
We noticed that you didn't select a key pair. If you want to be able to connect to your instance it is recommended that you create one or select an existing one.
Create new key pair
Proceed without key pair
Key pair name
Key pairs allow you to connect to your instance securely.
S3AccessInstanceKey
The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.
Key pair type
RSA
RSA encrypted private and public key pair
ED25519
ED25519 encrypted private and public key pair
Private key file format
.pem
For use with OpenSSH
.ppk
For use with PuTTY
When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. Learn more
Cancel
Launch instance
Summary
Number of instances Info
1
Software Image (AMI)
Amazon Linux 2023 AMI 2023.6.2...read more
ami-09115b7bffbe3c5e4
Virtual server type (instance type)
t2.micro
Firewall (security group)
New security group
Storage (volumes)
1 volume(s) - 8 GiB
Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GB of bandwidth to the internet.
Cancel
Launch instance
Preview code
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences



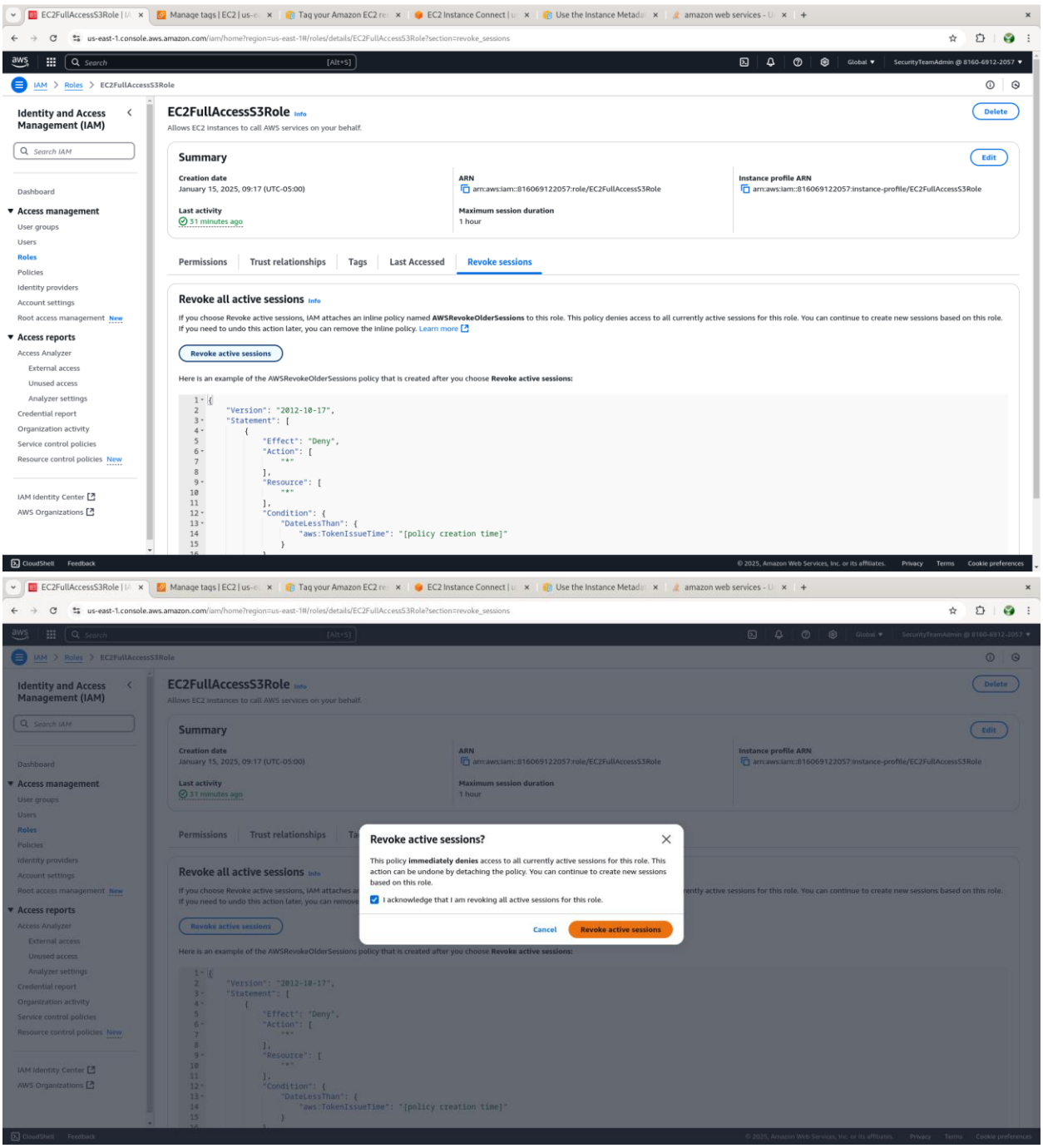
We launch the instance and update its role to the newly created EC2FullAccessS3Role.



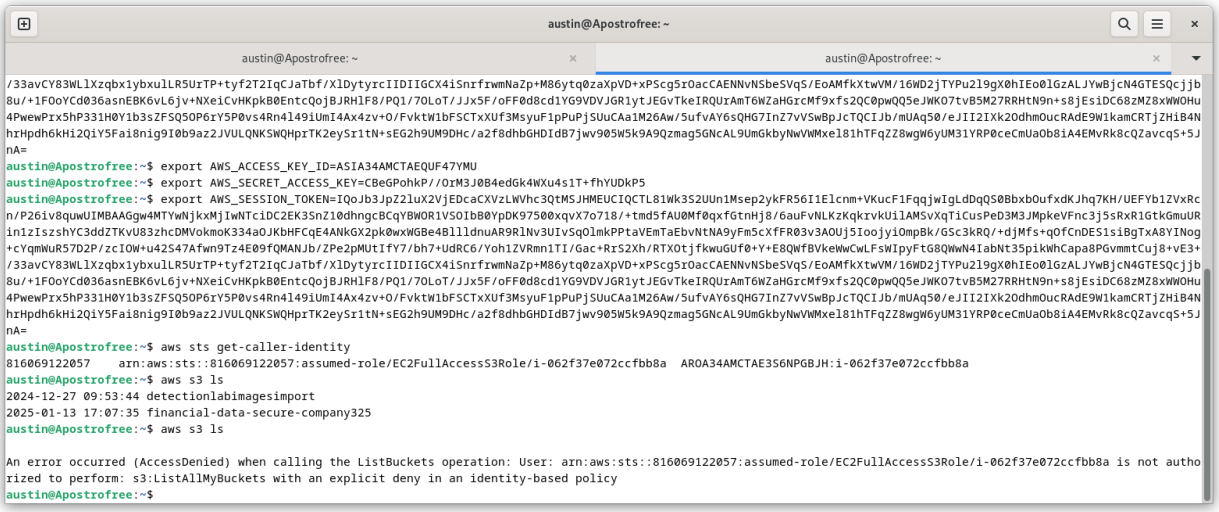
Here we see from the shell of the instance we can access S3 resources.



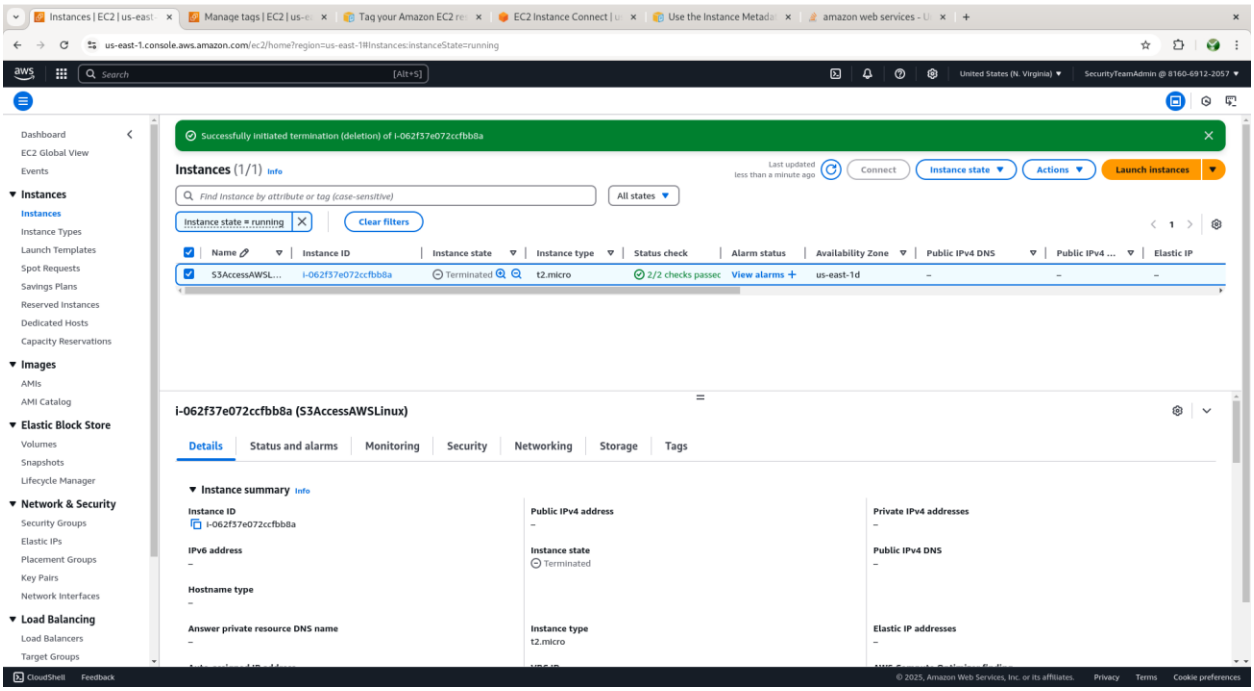
We then assume the role on our AWS CLI using the credentials from the EC2 instance including the role's session token. And show that we have access to S3 resources. Simulating the theft of the credentials from the instance.



In response we revoke the active session.



Now our stolen credentials are no longer active, and the session is given an access denied error when trying to list buckets.



Finally we terminate the instance and clean up the resources.