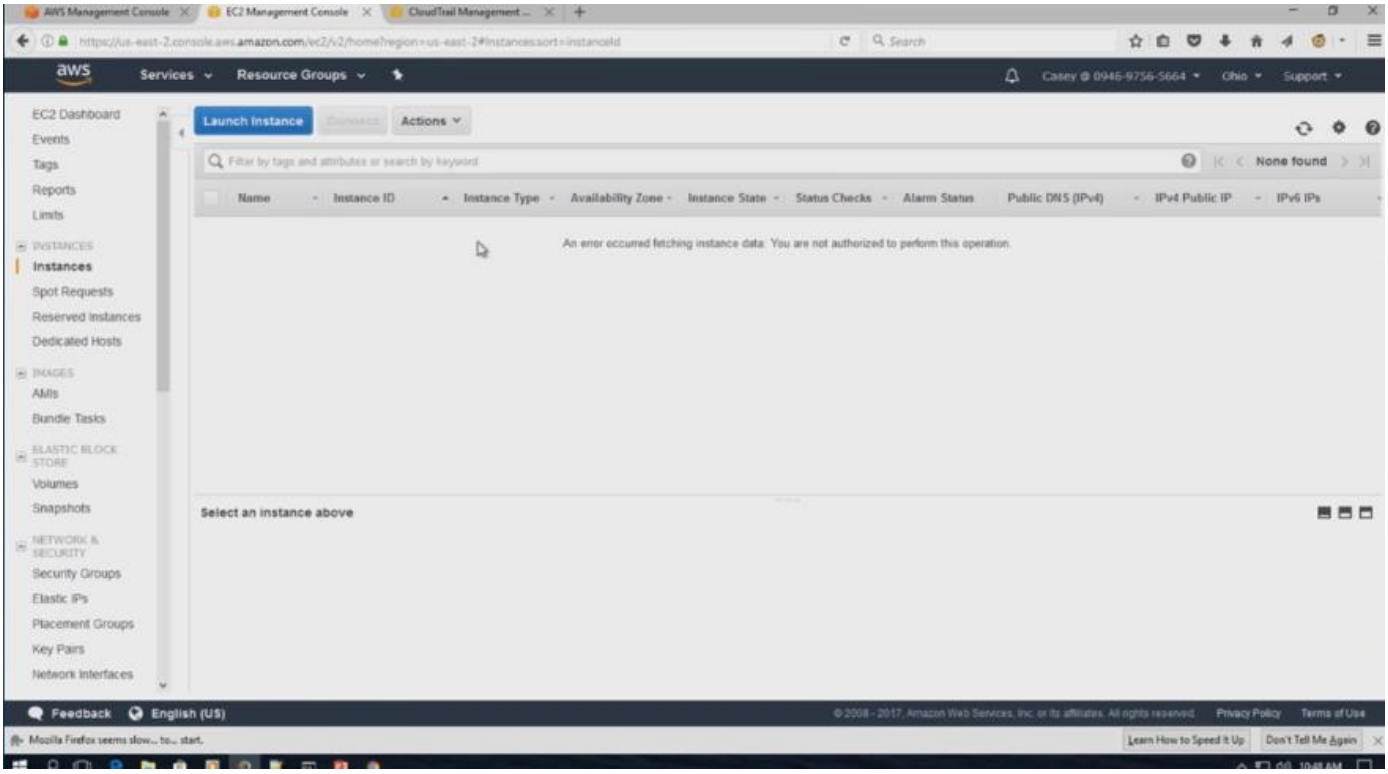
Are you interested in learning ***how to control access to your AWS resources***? Have you wondered how to best scope permissions to achieve least-privilege permissions access control? If your answer is "yes", this session is for you. We look at the ***Identity & Access Management policy language***, starting with the basics of the ***policy*** language and how to create and attach policies to IAM ***users***, ***groups***, and ***roles***. We explore policy variables, conditions, and tools to help you author least privilege policies. We cover common use cases, such as granting a user secure access to an Amazon ***S3 bucket*** or to launch an ***Amazon EC2 instance*** of a specific type.

We are going to build a policy to do the above

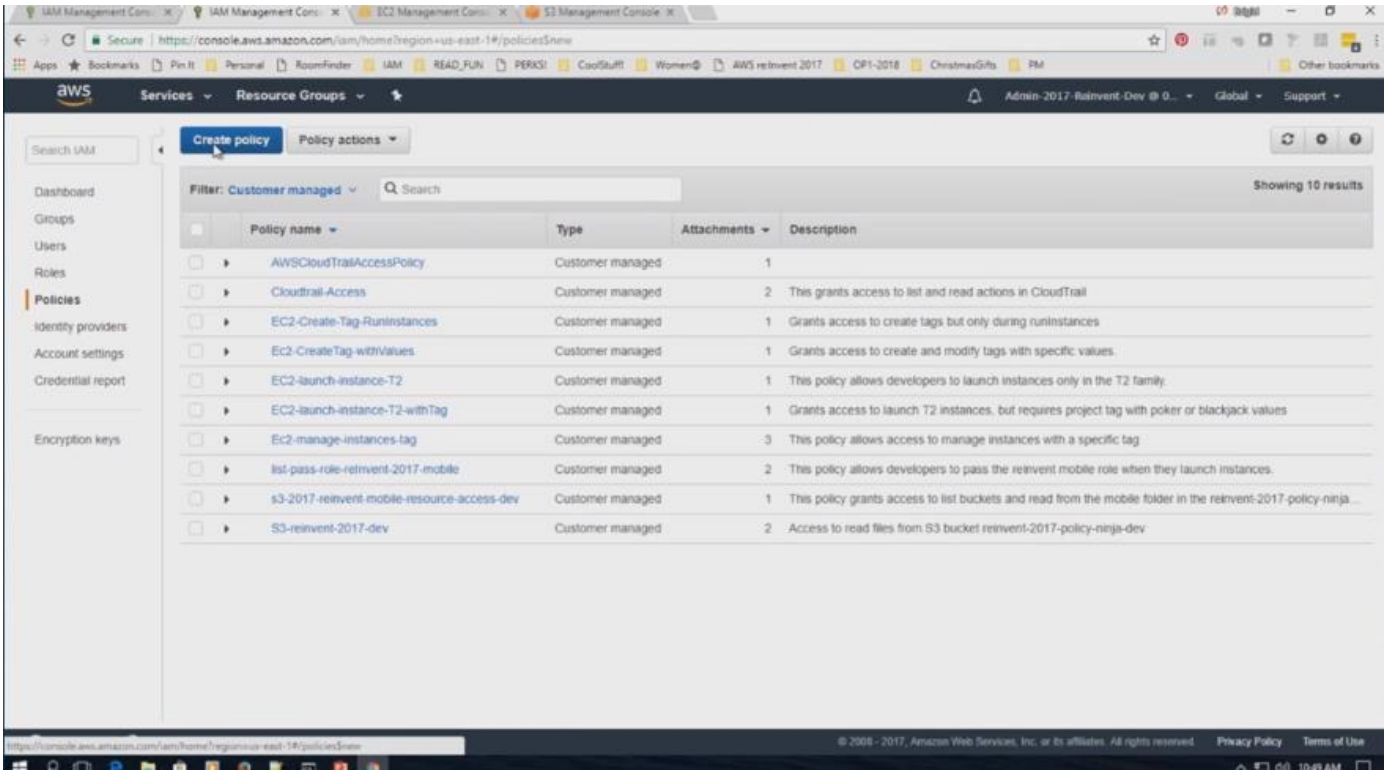
Limit Amazon EC2 instance types
Demo

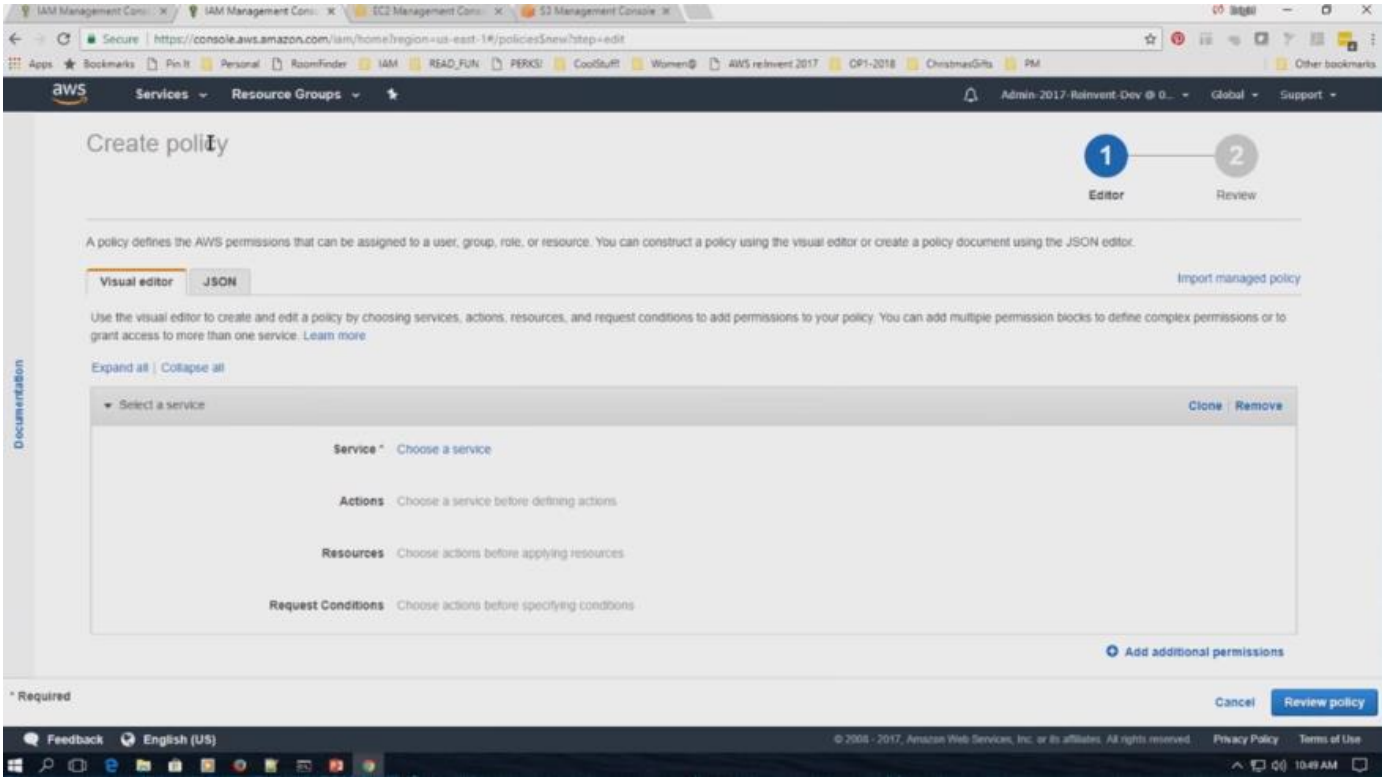
- Goal: Limit a user from starting an instance unless the instance is `t2.*`
- Let's try to:
  - Create a managed policy that attempts to limit starting an EC2 instance except for these instance types.
  - Attach that policy to an IAM user.

We will start with a demo in the console as below

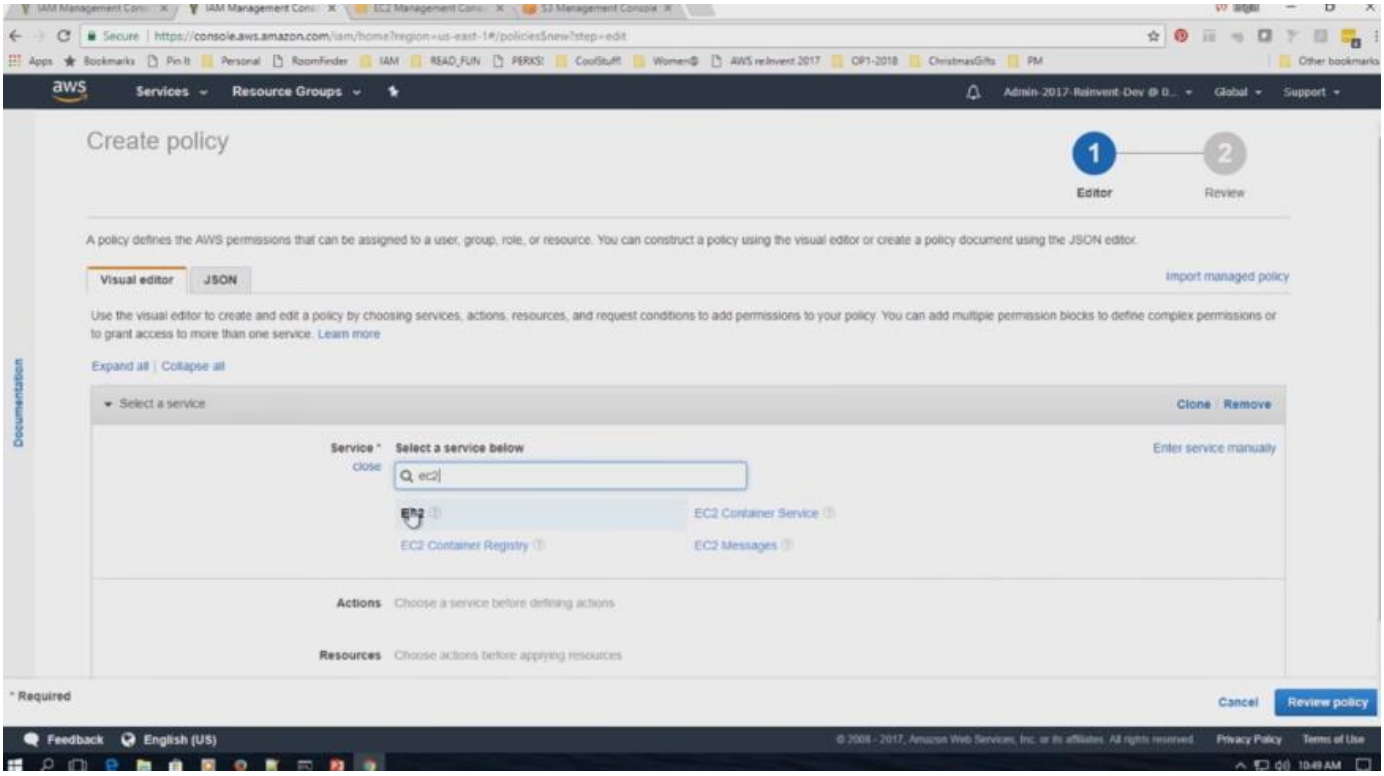


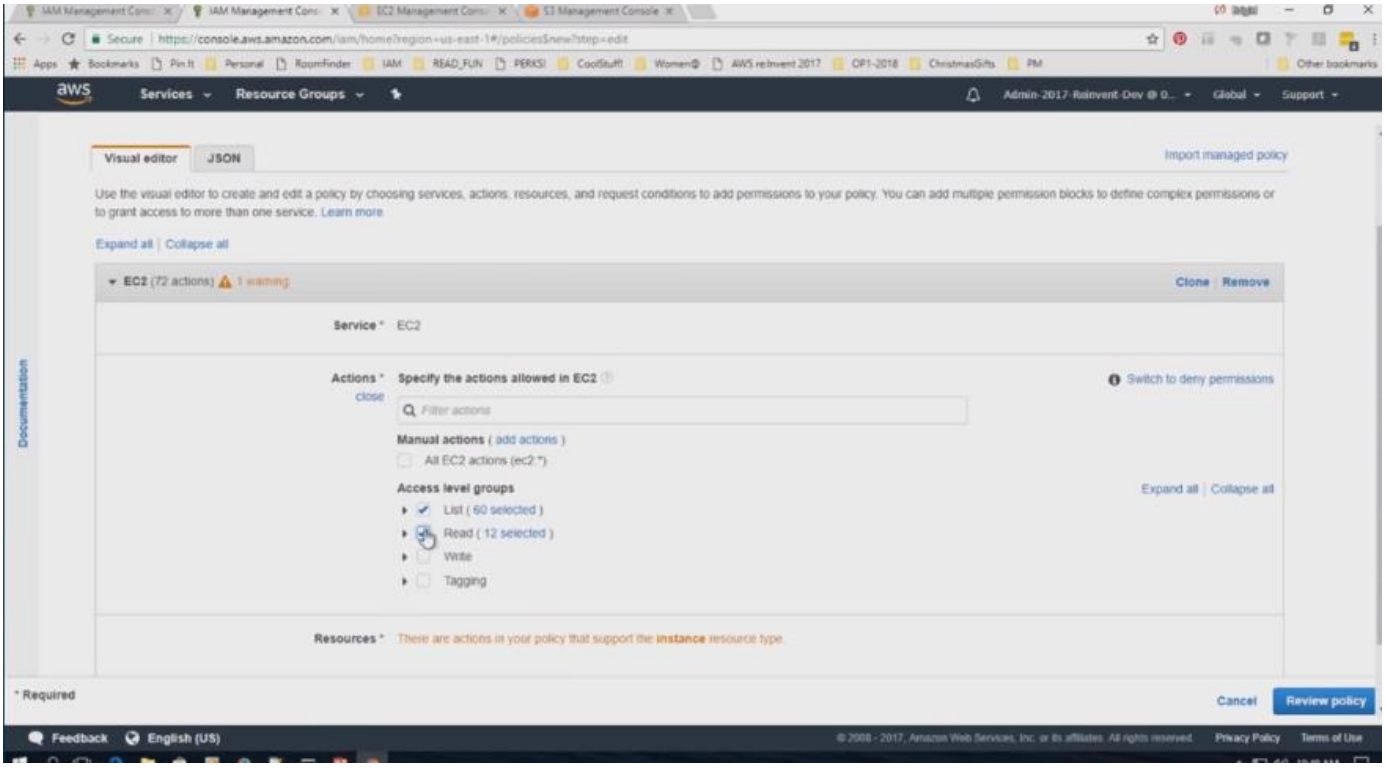
Casey has no permissions at the moment and cannot do anything. We will start by creating a policy to start giving him the exact access he needs to do his job

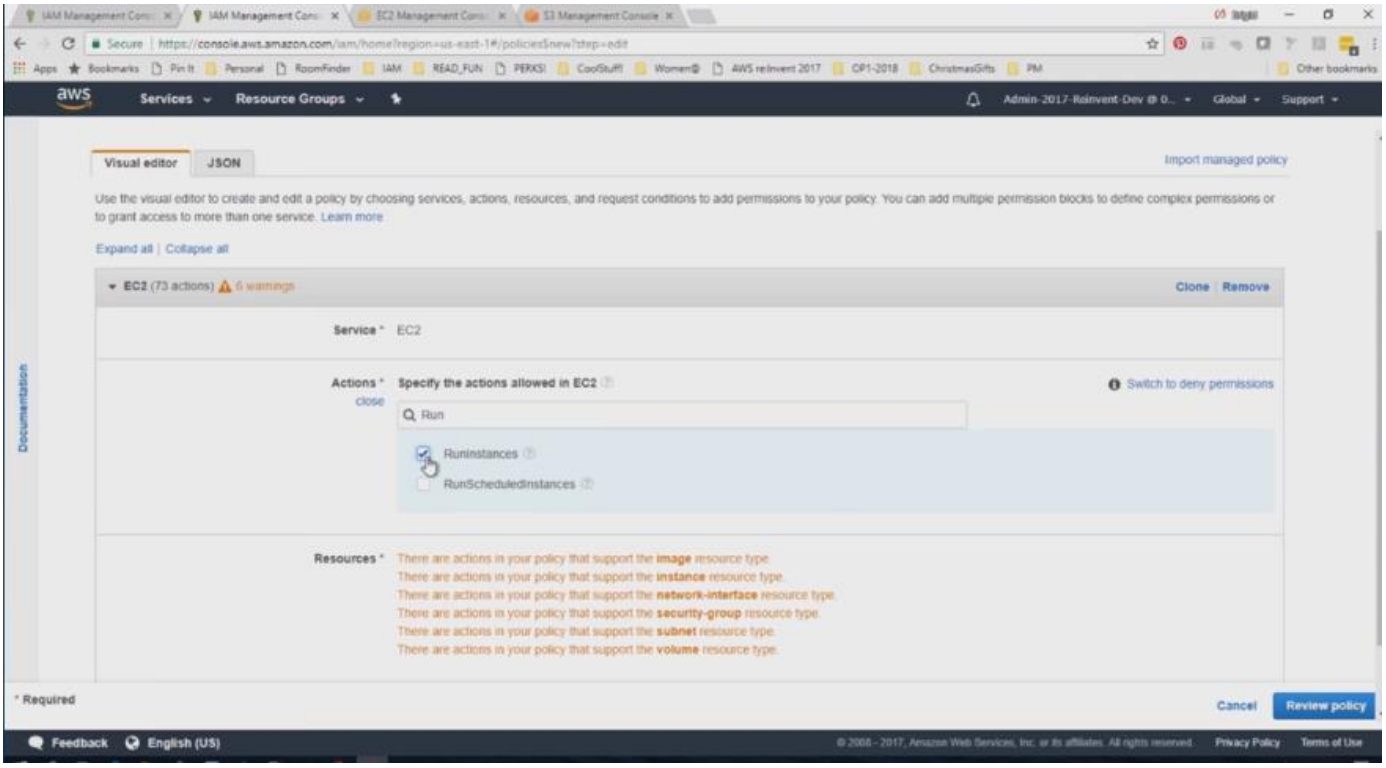




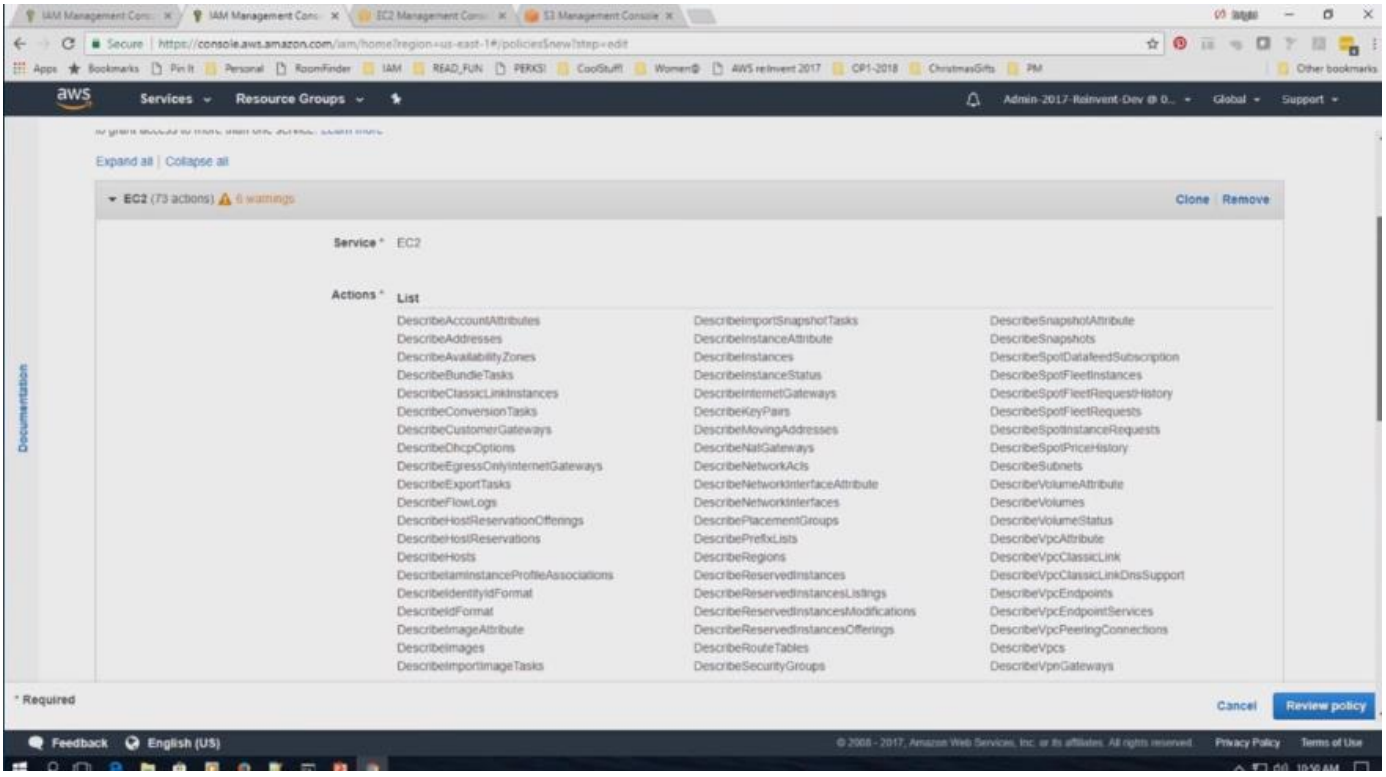
We want to create a policy that will allow Casey to create and use instances of t1 type only

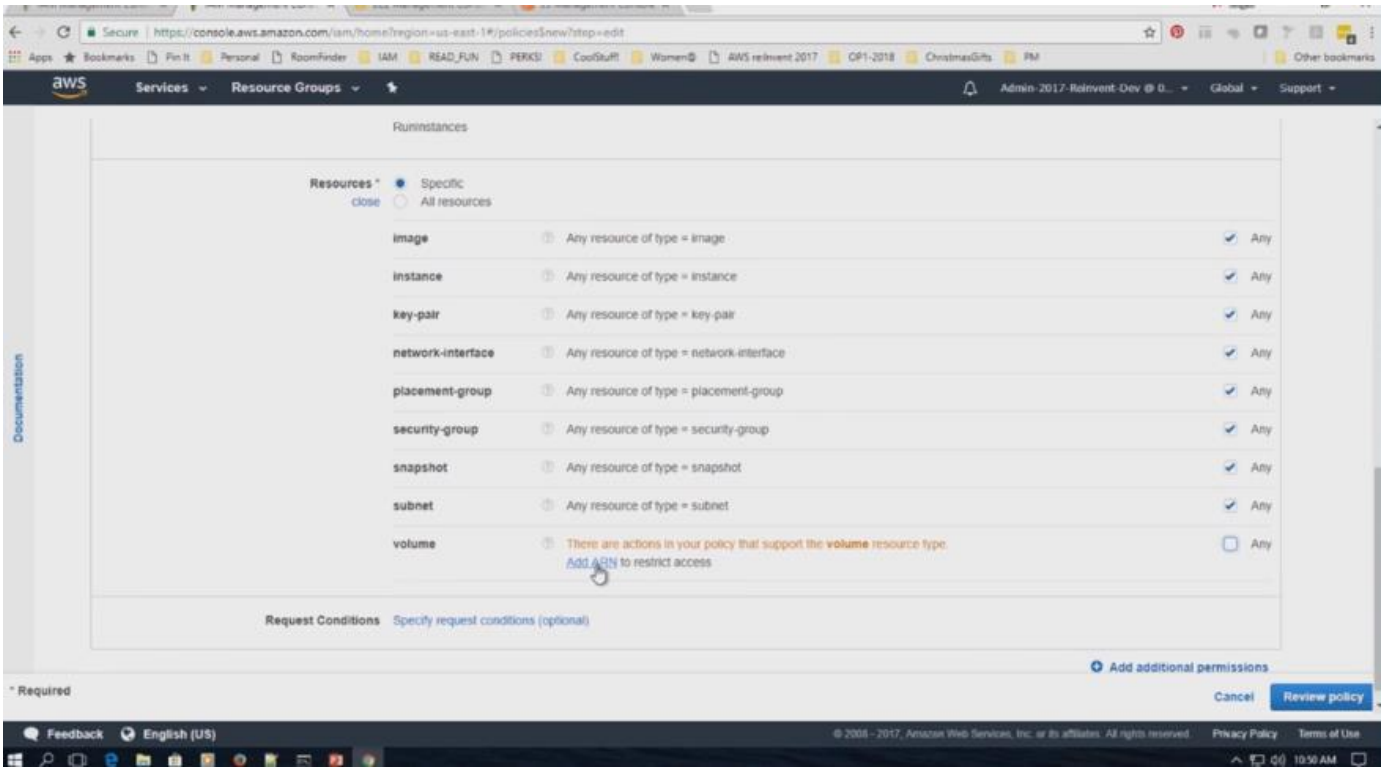

Create policy
1 Editor
2 Review
A policy defines the AWS permissions that can be assigned to a user, group, role, or resource. You can construct a policy using the visual editor or create a policy document using the JSON editor.
Visual editor
JSON
Import managed policy
Use the visual editor to create and edit a policy by choosing services, actions, resources, and request conditions to add permissions to your policy. You can add multiple permission blocks to define complex permissions or to grant access to more than one service. Learn more
Expand all | Collapse all
Select a service
Clone | Remove
Service *
close
Select a service below
Enter service manually
ec2
EC2 Container Service
EC2 Container Registry
EC2 Messages
Actions
Choose a service before defining actions
Resources
Choose actions before applying resources
* Required
Cancel
Review policy
Feedback
English (US)
Privacy Policy
Terms of Use

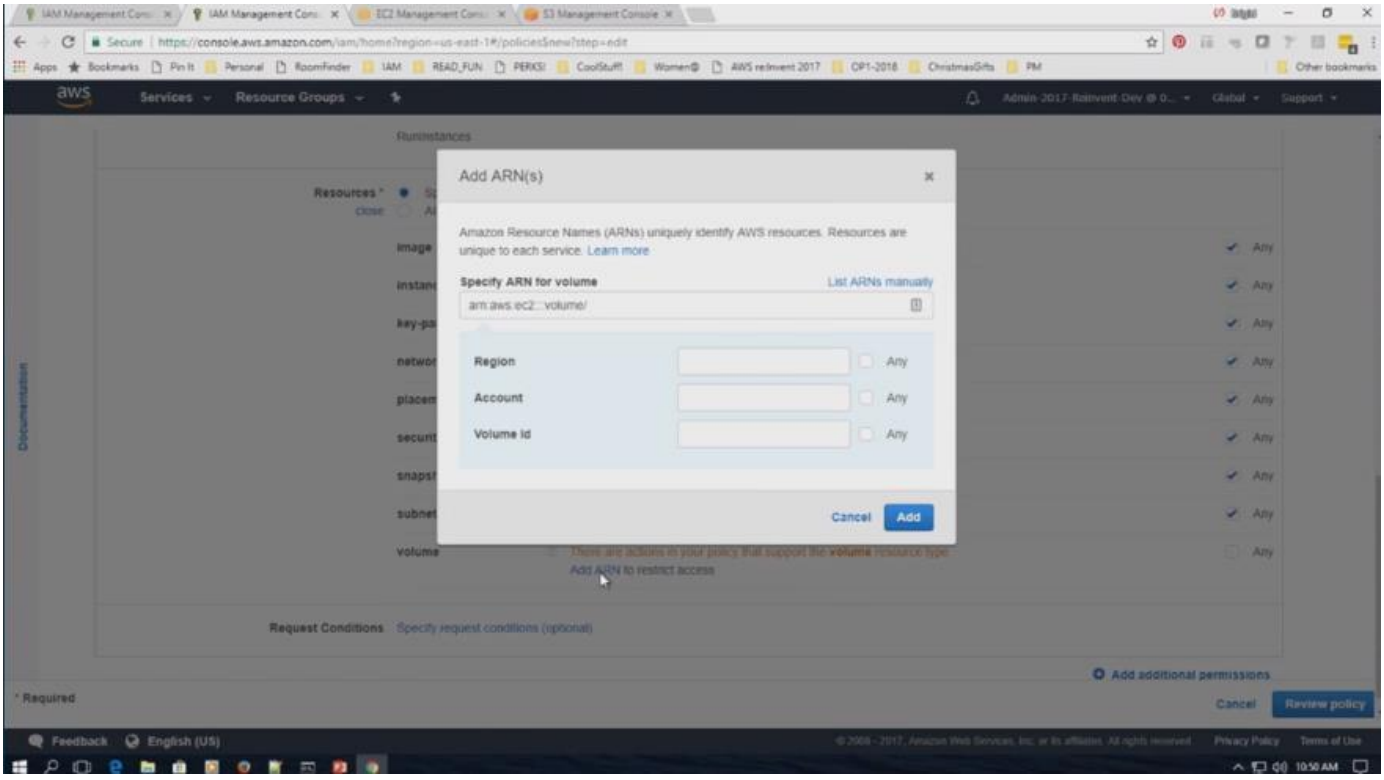

Visual editor
JSON
Import managed policy
Use the visual editor to create and edit a policy by choosing services, actions, resources, and request conditions to add permissions to your policy. You can add multiple permission blocks to define complex permissions or to grant access to more than one service. Learn more
Expand all | Collapse all
EC2 (72 actions) 1 warning
Clone | Remove
Service * EC2
Actions *
close
Specify the actions allowed in EC2
Switch to deny permissions
Filter actions
Manual actions (add actions)
All EC2 actions (ec2:*)
Access level groups
Expand all | Collapse all
List (60 selected)
Read (12 selected)
Write
Tagging
Resources *
There are actions in your policy that support the instance resource type.
* Required
Cancel
Review policy
Feedback
English (US)
Privacy Policy
Terms of Use

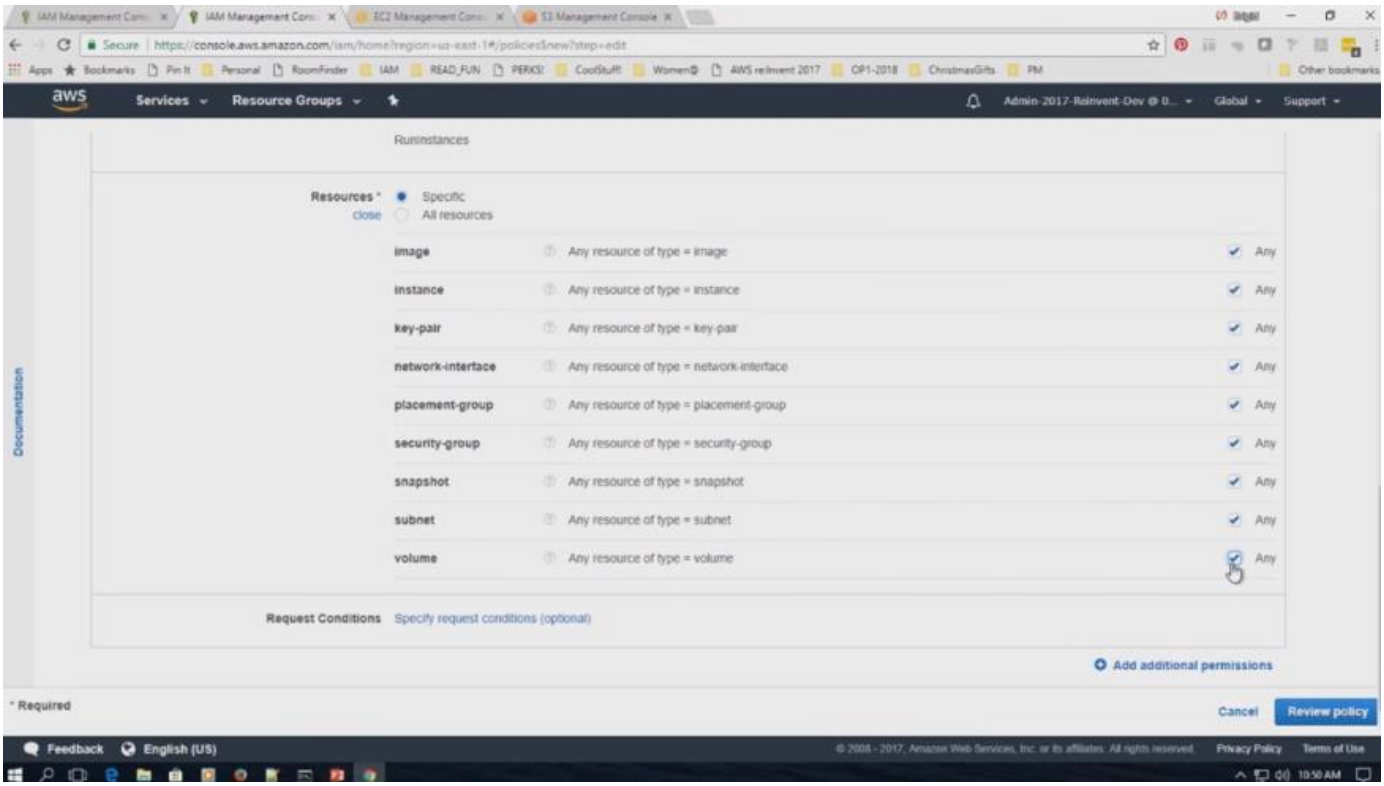


In addition to the List and Read permissions, we also need to allow Casey to run instances

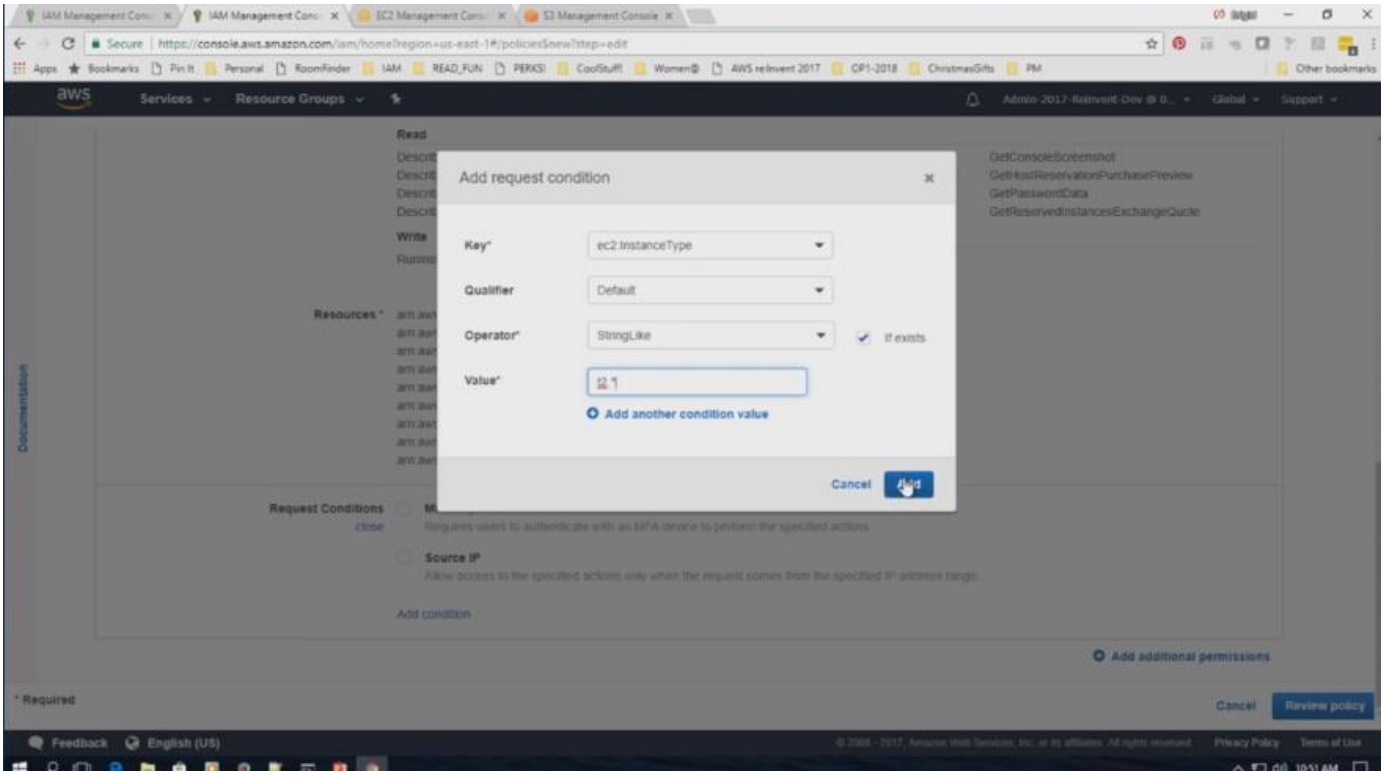


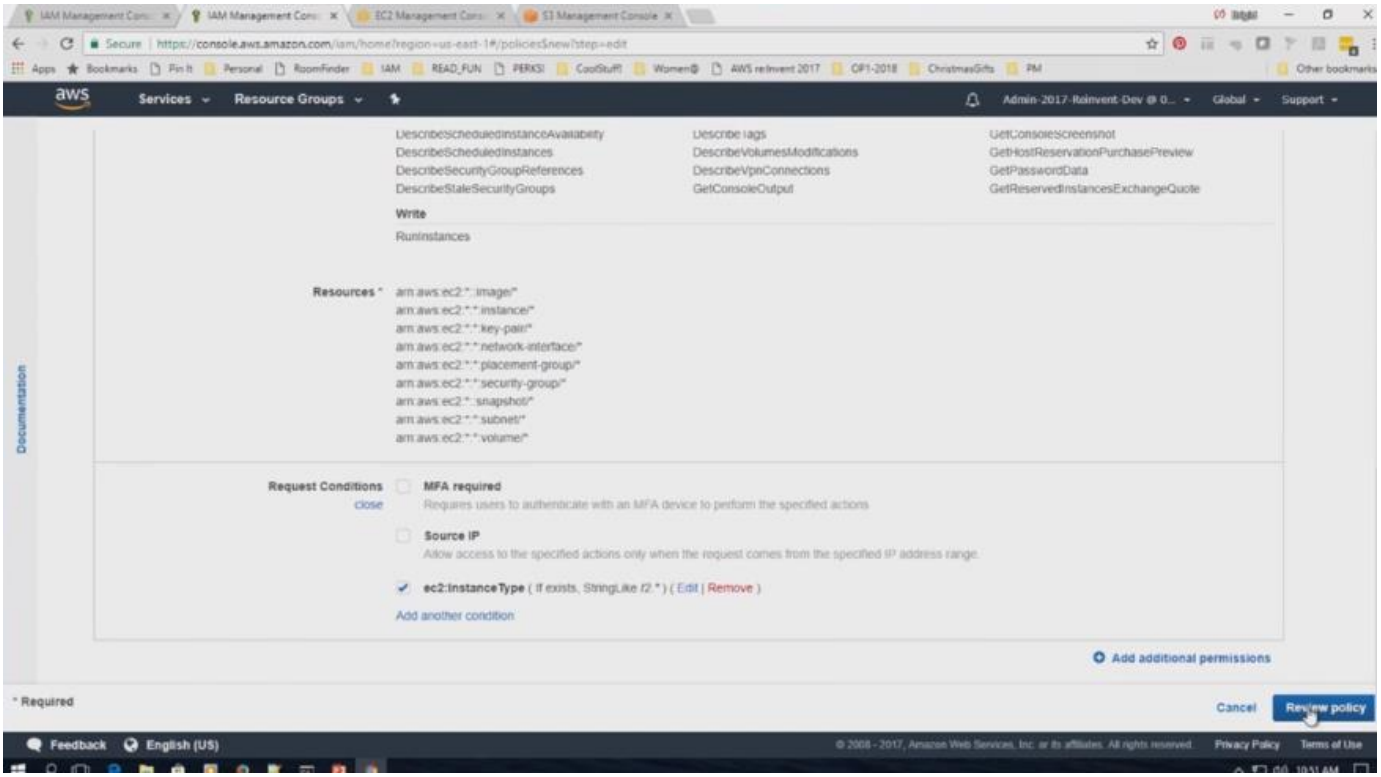

Services
Resource Groups
Runinstances
Resources *
Specific
All resources
close
image
Any resource of type = image
instance
Any resource of type = instance
key-pair
Any resource of type = key-pair
network-interface
Any resource of type = network-interface
placement-group
Any resource of type = placement-group
security-group
Any resource of type = security-group
snapshot
Any resource of type = snapshot
subnet
Any resource of type = subnet
volume
There are actions in your policy that support the volume resource type.
Add ARN to restrict access
Any
Request Conditions
Specify request conditions (optional)
Add additional permissions
* Required
Cancel
Review policy
Feedback
English (US)
Documentation


Add ARN(s)
Amazon Resource Names (ARNs) uniquely identify AWS resources. Resources are unique to each service. Learn more
Specify ARN for volume
List ARNs manually
arn:aws:ec2:::volume/
Region
Account
Volume id
Any
Cancel
Add

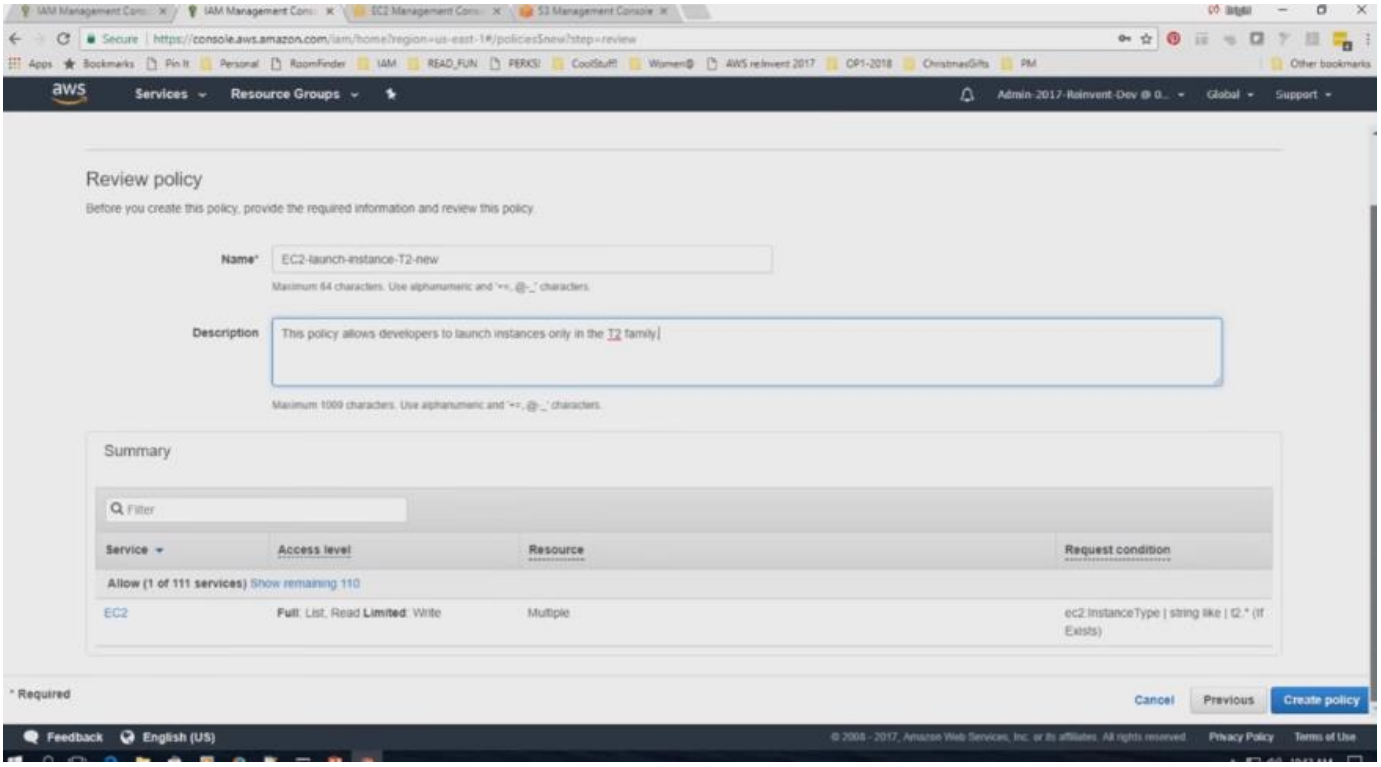


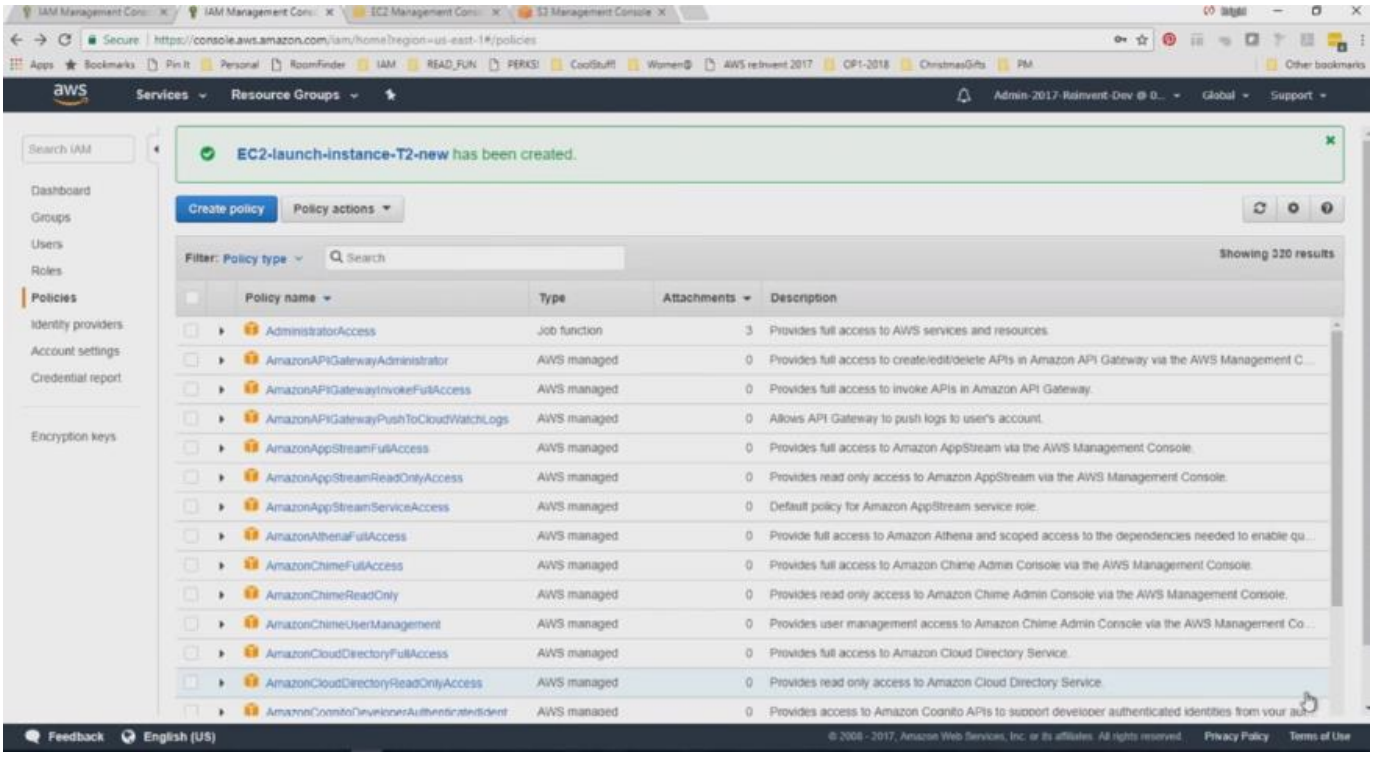
But where does the t1 instance type come into the policy? That is a condition that we need to specify as below



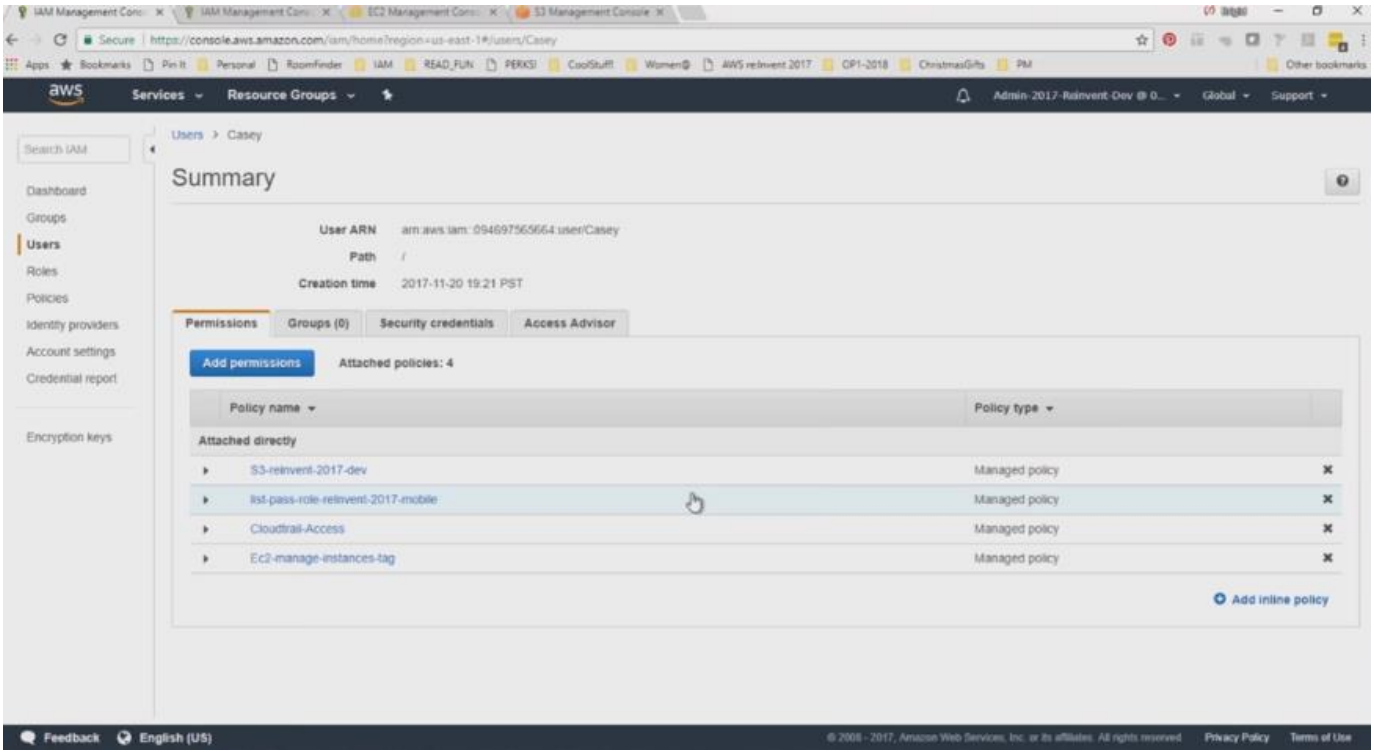


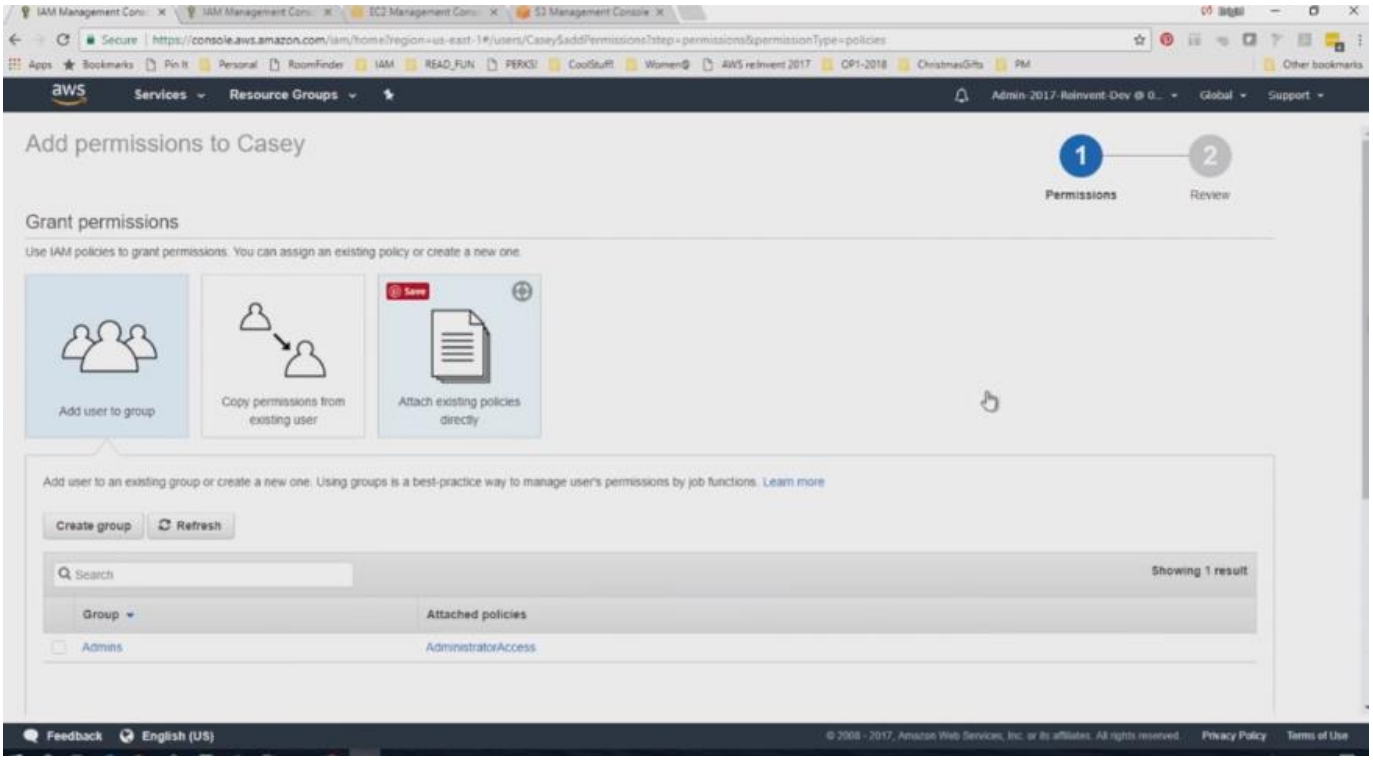
We now have the policy, we can review it and click the button

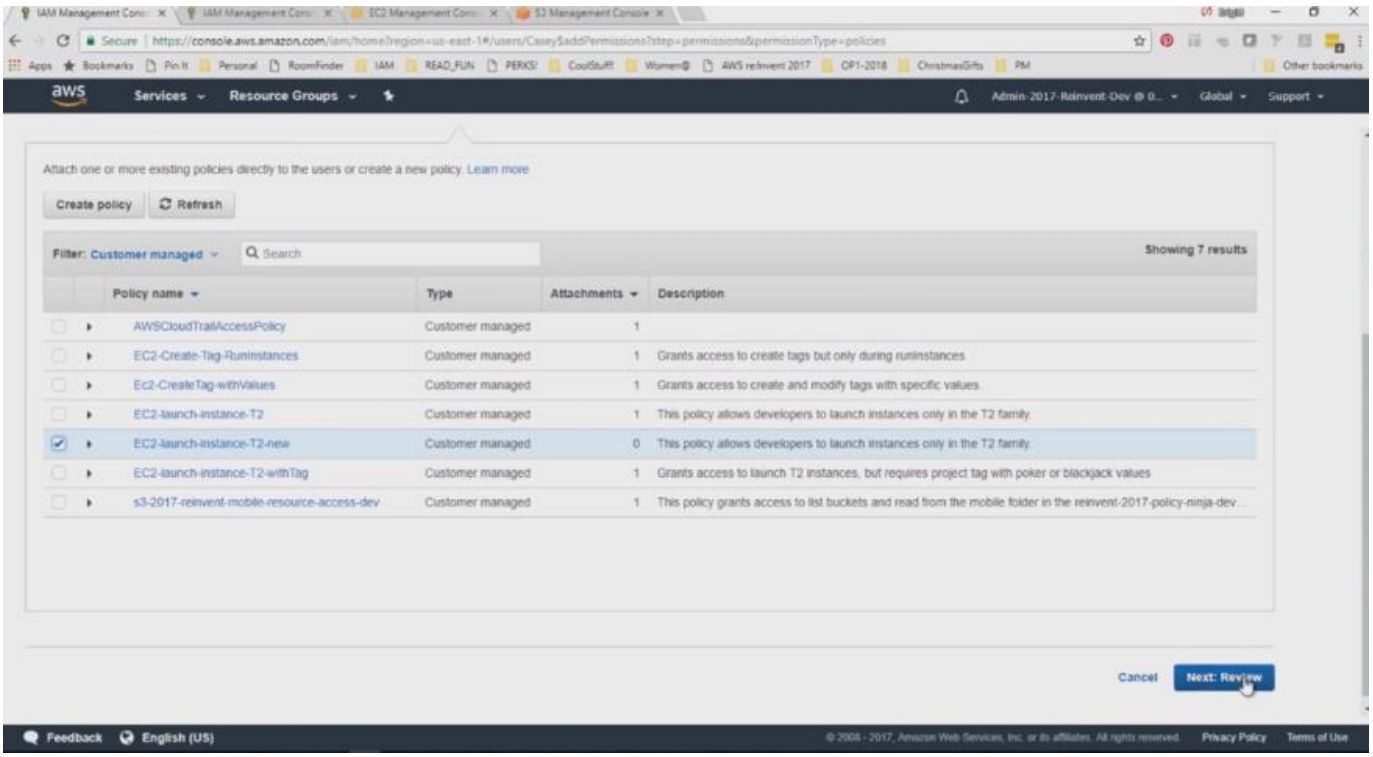


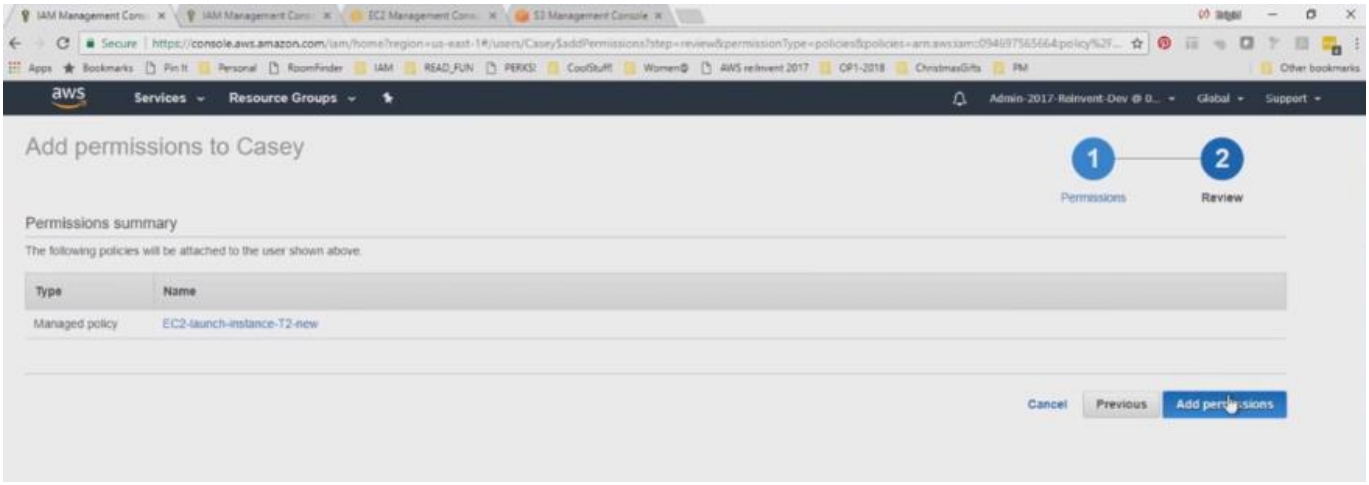


we then attach the policy for Casey

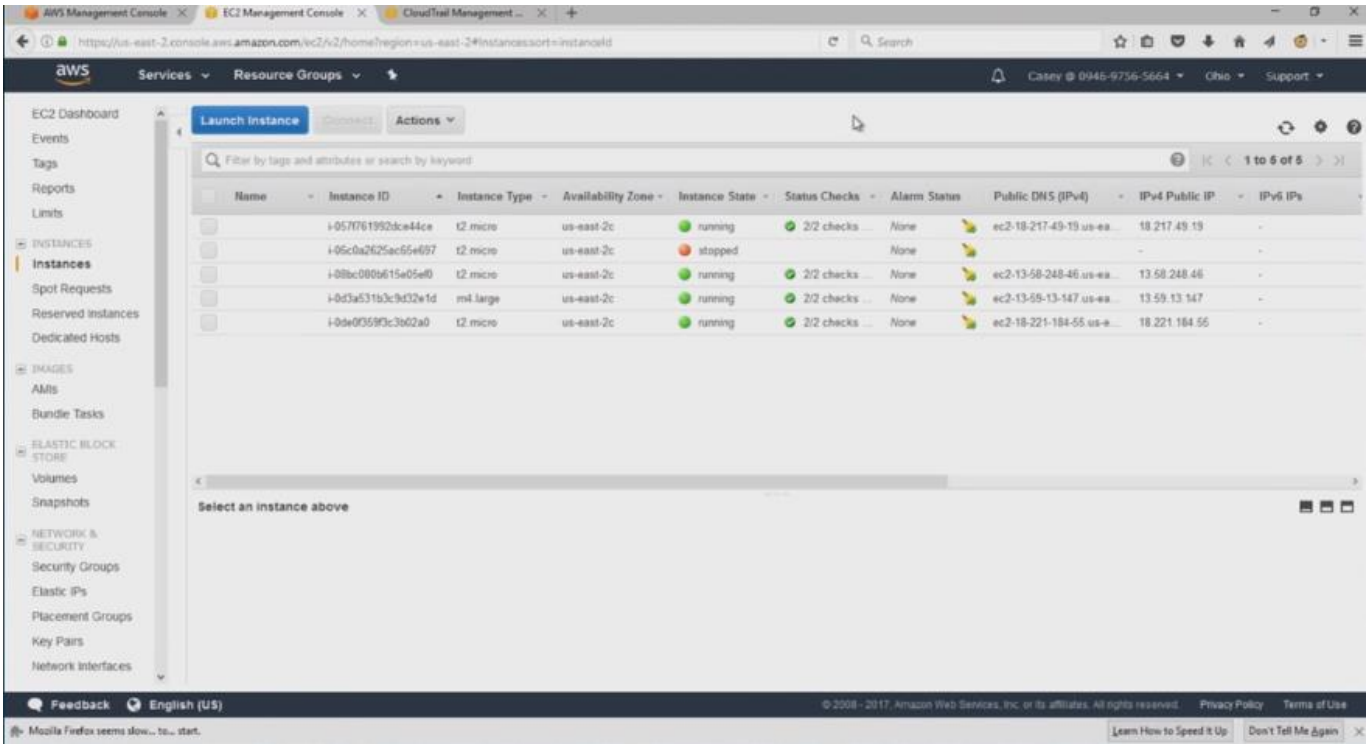



Add permissions to Casey
1 Permissions
2 Review
Grant permissions
Use IAM policies to grant permissions. You can assign an existing policy or create a new one.
Add user to group
Copy permissions from existing user
Attach existing policies directly
Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. Learn more
Create group
Refresh
Search
Showing 1 result
Group
Attached policies
Admins
AdministratorAccess

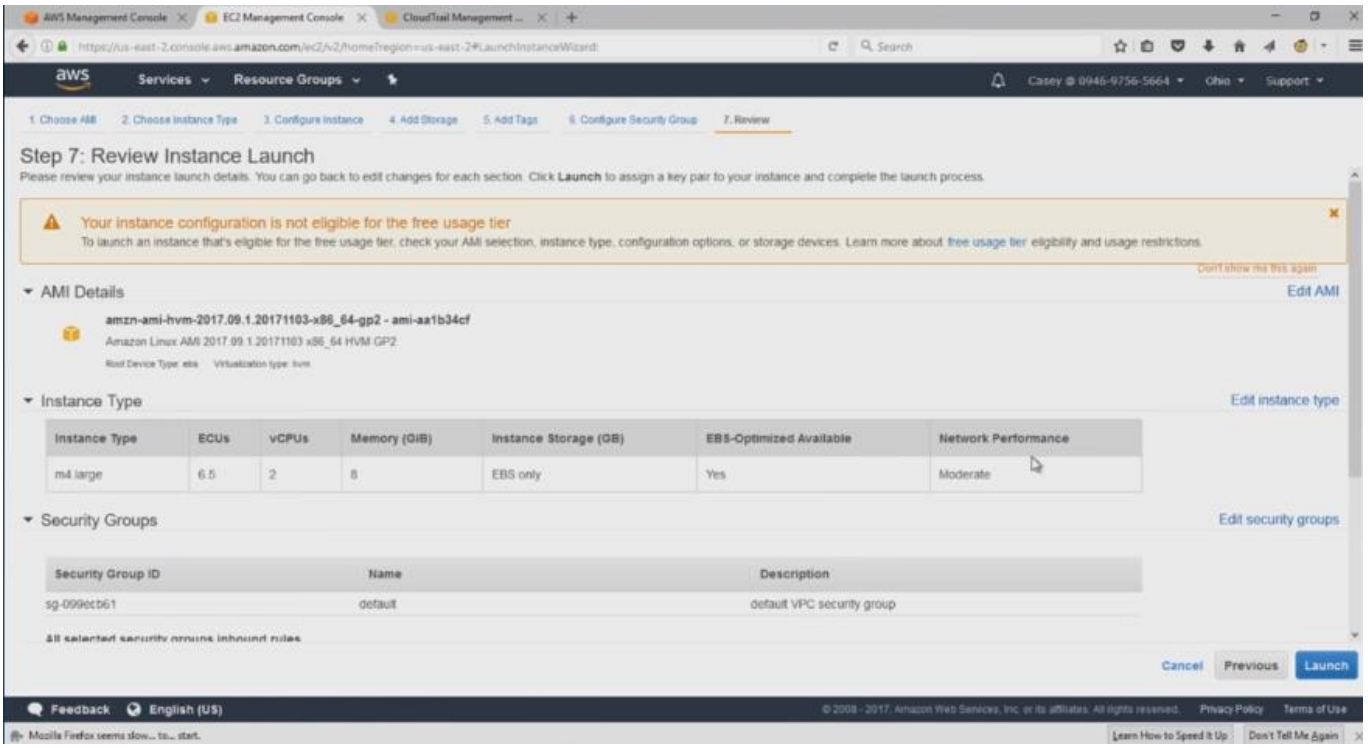

Attach one or more existing policies directly to the users or create a new policy. Learn more
Create policy
Refresh
Filter: Customer managed
Search
Showing 7 results
Policy name
Type
Attachments
Description
AWSCloudTrailAccessPolicy Customer managed 1
EC2-Create-Tag-RunInstances Customer managed 1 Grants access to create tags but only during runinstances
Ec2-CreateTag-withValues Customer managed 1 Grants access to create and modify tags with specific values.
EC2-launch-instance-T2 Customer managed 1 This policy allows developers to launch instances only in the T2 family.
EC2-launch-instance-T2-new Customer managed 0 This policy allows developers to launch instances only in the T2 family.
EC2-launch-instance-T2-withTag Customer managed 1 Grants access to launch T2 instances, but requires project tag with poker or blackjack values
s3-2017-reinvent-mobile-resource-access-dev Customer managed 1 This policy grants access to list buckets and read from the mobile folder in the reinvent-2017-policy-ninja-dev
Cancel
Next: Review

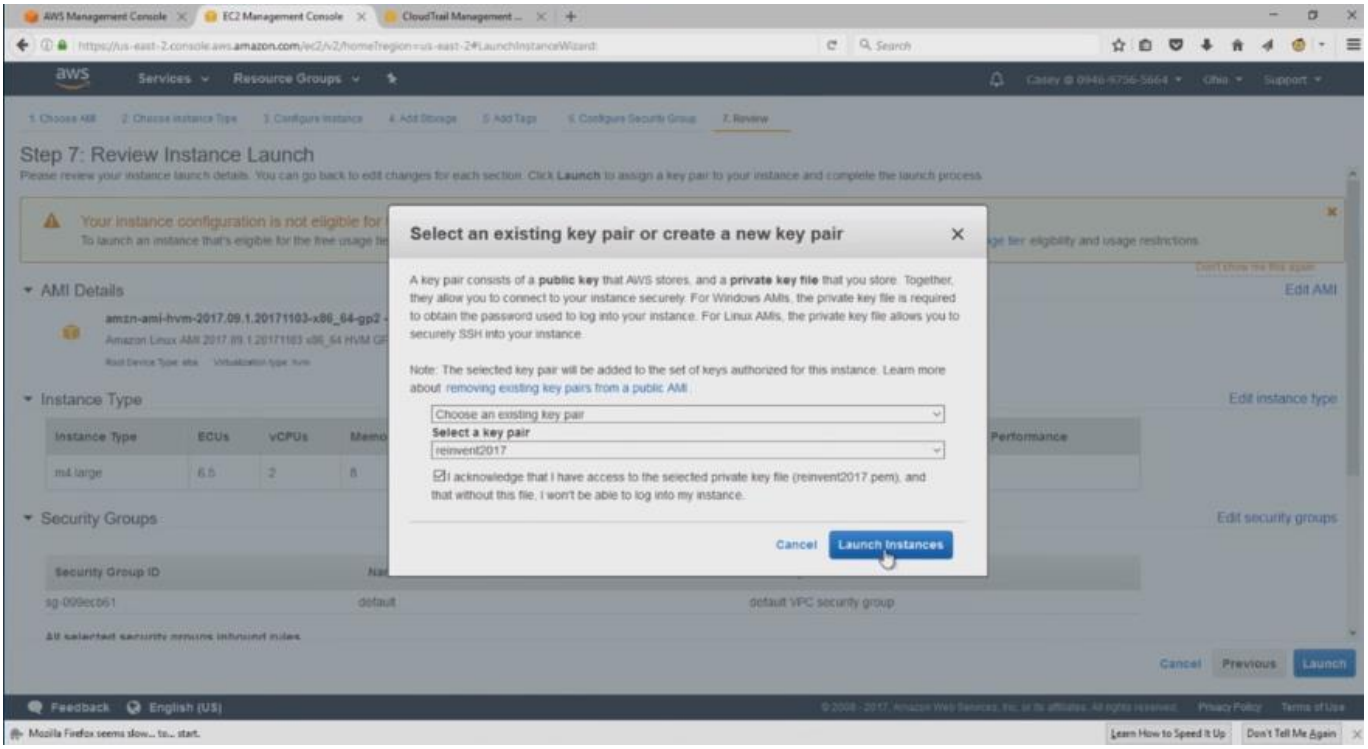


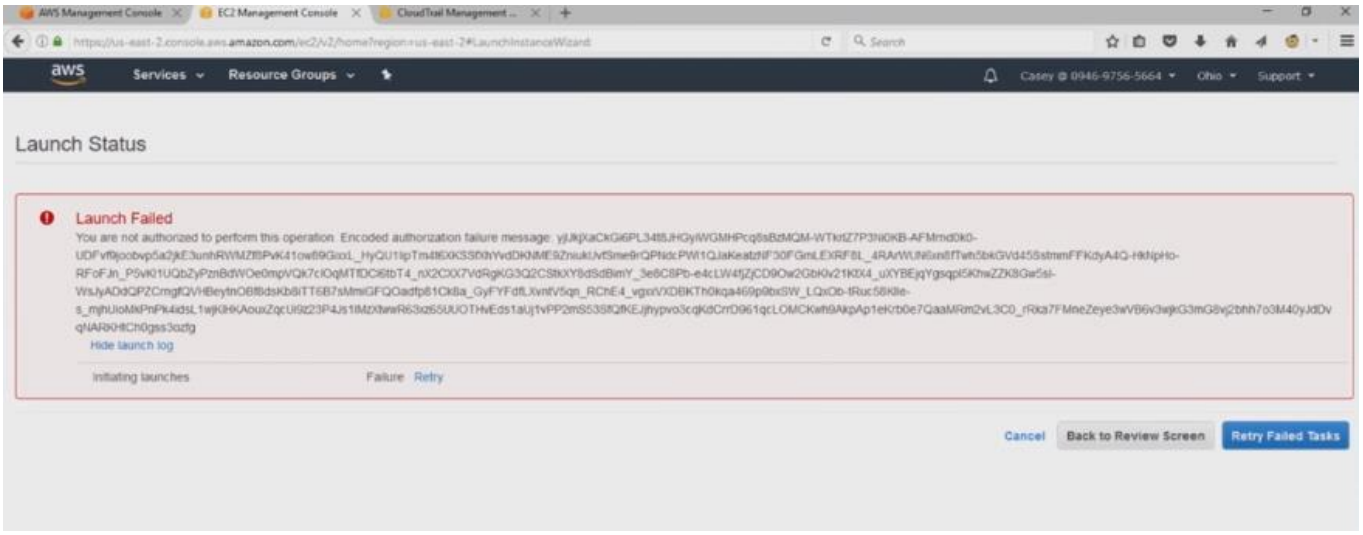
Next, we need to go and see what Casey can do below

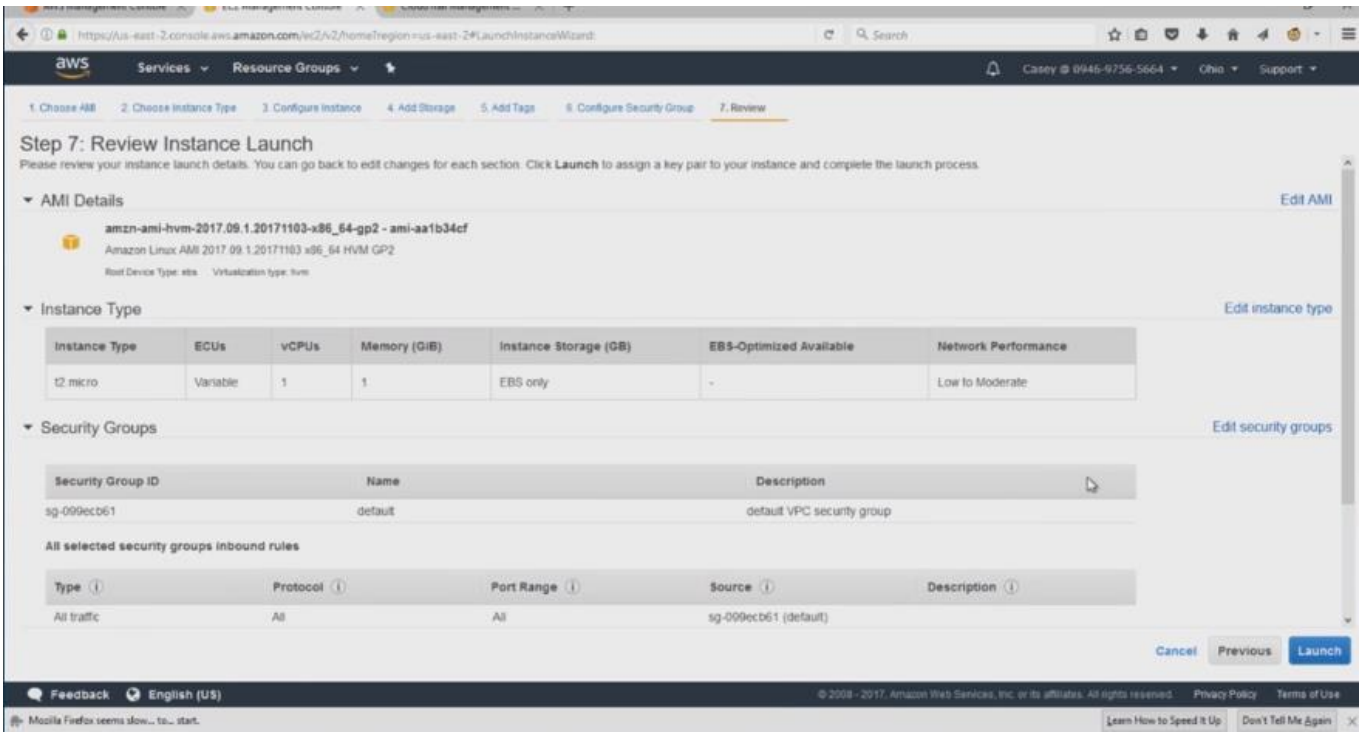


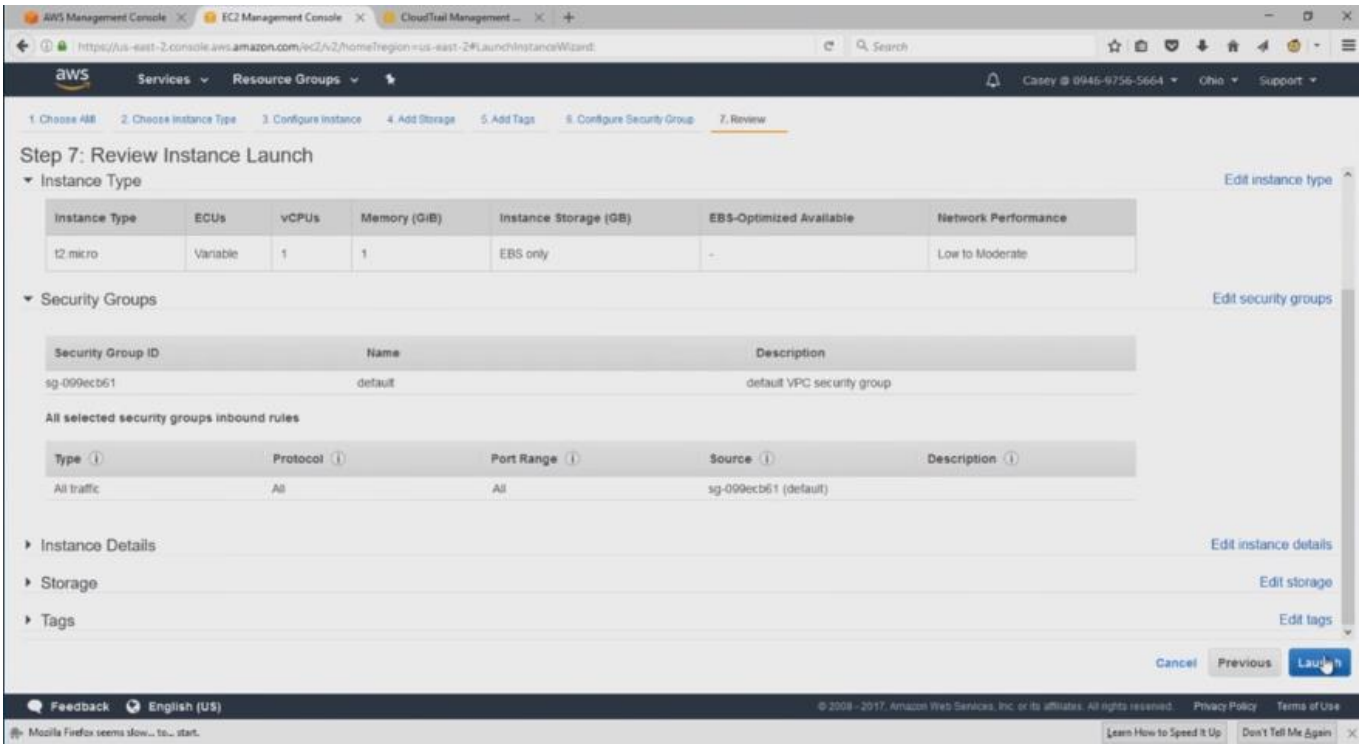
Casey can now see some instances

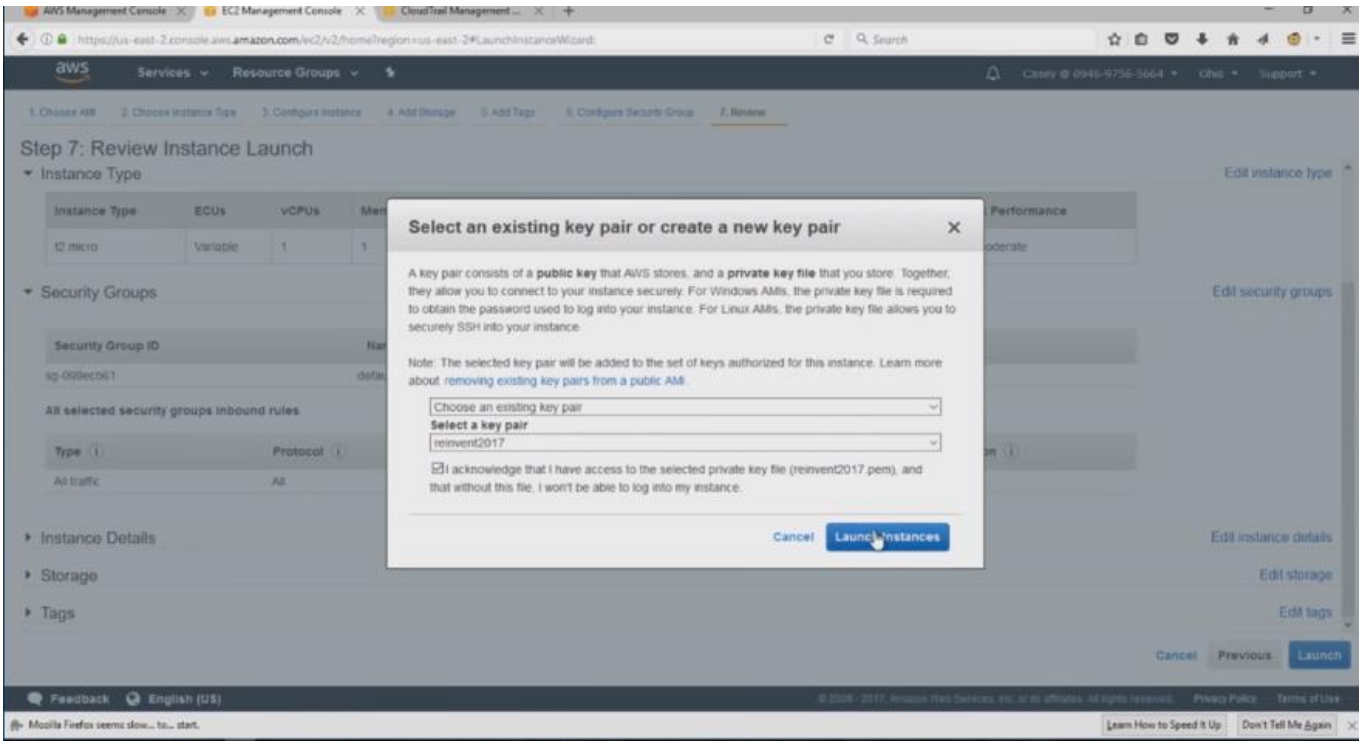

Services
Resource Groups
1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
Step 7: Review Instance Launch
Please review your instance launch details. You can go back to edit changes for each section. Click Launch to assign a key pair to your instance and complete the launch process.
Your instance configuration is not eligible for the free usage tier
To launch an instance that's eligible for the free usage tier, check your AMI selection, instance type, configuration options, or storage devices. Learn more about free usage tier eligibility and usage restrictions.
AMI Details
Edit AMI
amzn-ami-hvm-2017.09.1.20171103-x86_64-gp2 - ami-aa1b34cf
Amazon Linux AMI 2017.09.1.20171103 x86_64 HVM GP2
Instance Type
Edit instance type
Instance Type | ECUs | vCPUs | Memory (GiB) | Instance Storage (GB) | EBS-Optimized Available | Network Performance
m4.large | 6.5 | 2 | 8 | EBS only | Yes | Moderate
Security Groups
Edit security groups
Security Group ID | Name | Description
sg-099ecb61 | default | default VPC security group
Cancel
Previous
Launch
Feedback
English (US)

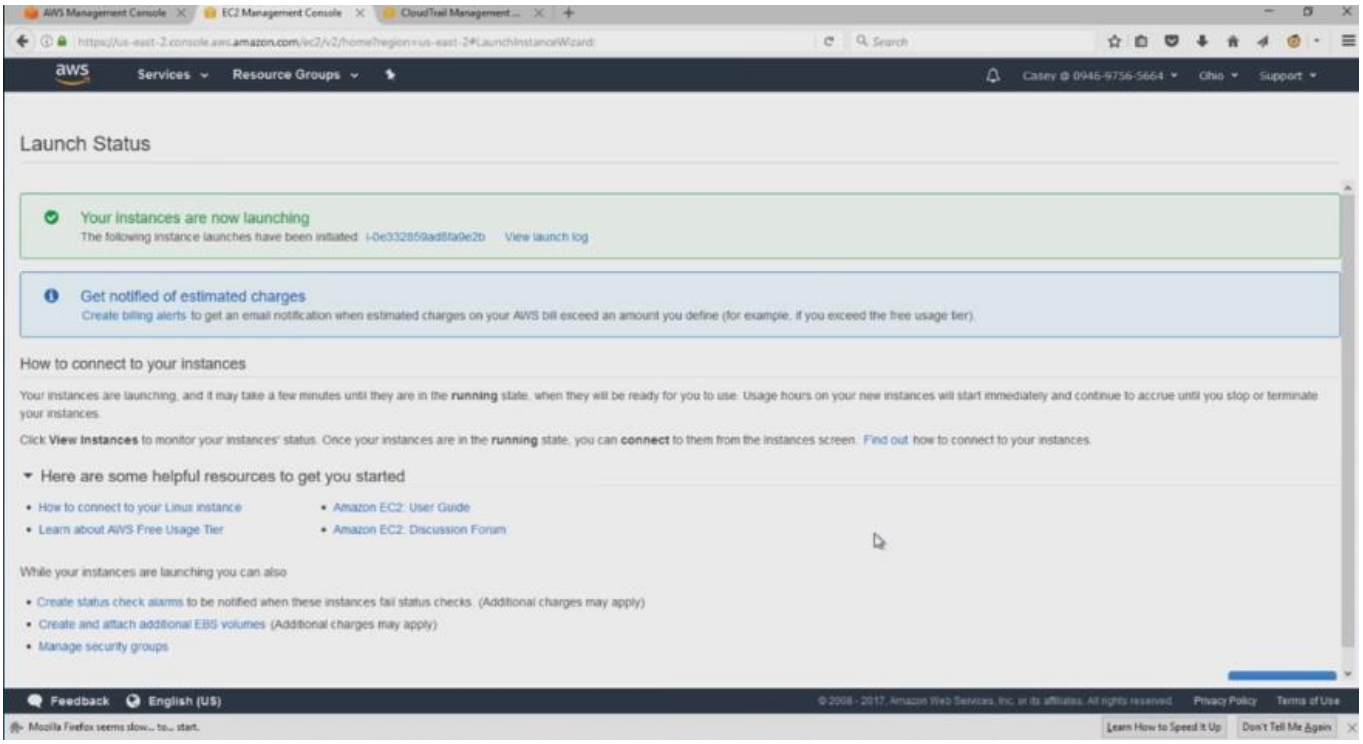

Select an existing key pair or create a new key pair
A key pair consists of a public key that AWS stores, and a private key file that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.
Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.
Choose an existing key pair
Select a key pair
reinvent2017
I acknowledge that I have access to the selected private key file (reinvent2017.pem), and that without this file, I won't be able to log into my instance.
Cancel
Launch Instances


Launch Status
Launch Failed
You are not authorized to perform this operation. Encoded authorization failure message: yjJkjXaCkGi6PL348SJHGyiWGMHPcq6sBzMQM-WTktSZ7P3NiOKB-AFMmd0k0-UDFvf9joobvp5a2jkE3unhRWMZf8PvK41ow69Gcol_HyQU1lpTm4tE6XKSSfXhYWdDkMES2niukUvtSme9rQPfidcPWt1QJaKeatzfiF30FGmLEXRF8L_4RArWUNi6xn8fTwh5bkGVd455stmmFFKdyA4Q-HkNpHo-RFoFJn_P5vKl1UQbZyPznBdWOe0mpVQk7cIOqMTDCi6tbT4_nX2CiX7VdRgKG3Q2CStkXY8dSdBimY_3e8C8Pb-e4cLW4fjZjCD9Ow2GbKv21KtX4_uXYBEjqYgsqpl5KhwZZK8Gw5sl-WsJyADdQPZCmgfQVHBeytnOBfbdsKb8iTT6B7sMmiGFQOadfp81CkBa_GyFYFdfLXvnfV5qn_RCnE4_vgoVXDBKTh0kqa469p9bxSW_LQxOb-tRuc58Klle-s_mjhUioMkPnPk4idsL1wjKiHkAouxZqcUi9z23P4Js1IMzXtwaR63qz65UUOTHvEds1aUjtvPP2mS53SfQfKEJjhypvo3cqKdCrrD961qcLOMCKwh9AkpAp1eKrb0e7QaaMRm2vL3C0_rRka7FMneZeye3wVB6v3wjkG3mG8vj2bhh7o3M40yJdDvqNAR9HCh0gss3ozfg
Hide launch log
Initiating launches
Failure
Retry
Cancel
Back to Review Screen
Retry Failed Tasks

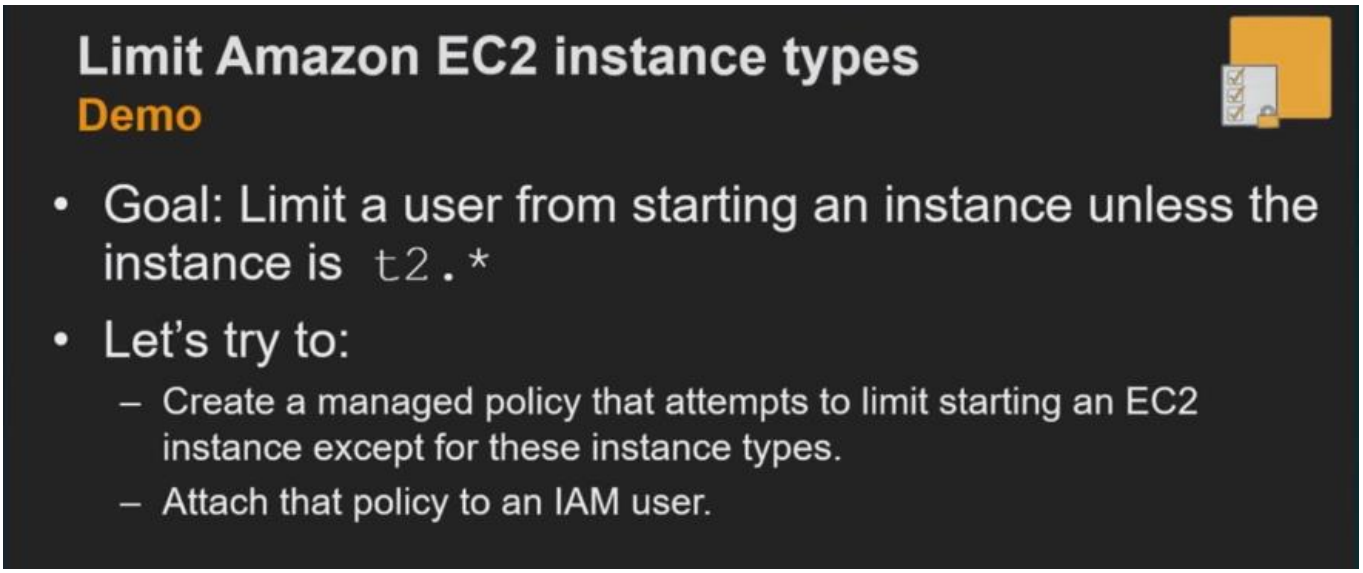

Step 7: Review Instance Launch
Please review your instance launch details. You can go back to edit changes for each section. Click Launch to assign a key pair to your instance and complete the launch process.
AMI Details
Edit AMI
amzn-ami-hvm-2017.09.1.20171103-x86_64-gp2 - ami-aa1b34cf
Amazon Linux AMI 2017.09.1.20171103 x86_64 HVM GP2
Instance Type
Edit instance type
Instance Type | ECUs | vCPUs | Memory (GiB) | Instance Storage (GB) | EBS-Optimized Available | Network Performance
t2.micro | Variable | 1 | 1 | EBS only | - | Low to Moderate
Security Groups
Edit security groups
Security Group ID | Name | Description
sg-099ecb61 | default | default VPC security group
All selected security groups inbound rules
Type | Protocol | Port Range | Source | Description
All traffic | All | All | sg-099ecb61 (default)
Cancel
Previous
Launch
Feedback
English (US)

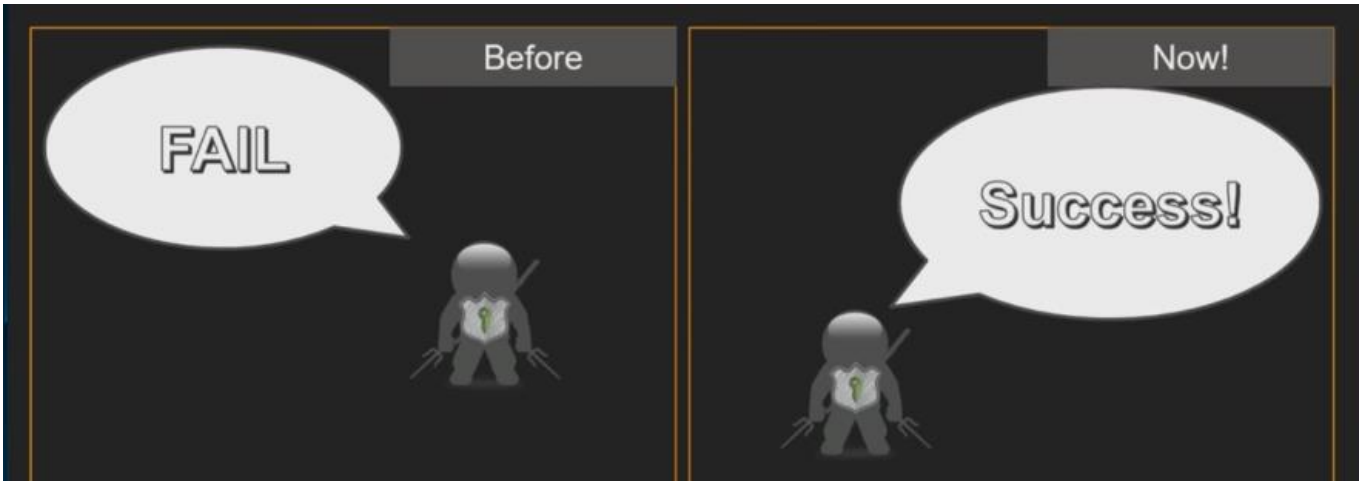

Services
Resource Groups
Step 7: Review Instance Launch
Instance Type
Edit instance type
Instance Type | ECUs | vCPUs | Memory (GiB) | Instance Storage (GB) | EBS-Optimized Available | Network Performance
t2.micro | Variable | 1 | 1 | EBS only | - | Low to Moderate
Security Groups
Edit security groups
Security Group ID | Name | Description
sg-099ecb61 | default | default VPC security group
All selected security groups inbound rules
Type | Protocol | Port Range | Source | Description
All traffic | All | All | sg-099ecb61 (default)
Instance Details
Edit instance details
Storage
Edit storage
Tags
Edit tags
Cancel
Previous
Launch
Feedback
English (US)


Select an existing key pair or create a new key pair
A key pair consists of a public key that AWS stores, and a private key file that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.
Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.
Choose an existing key pair
Select a key pair
reinvent2017
I acknowledge that I have access to the selected private key file (reinvent2017.pem), and that without this file, I won't be able to log into my instance.
Cancel
Launch Instances



Casey can only use the t2 instance types





What to expect from this session

Learn the policy language to control access to your AWS resources

Understand the different types of policies, why to use them, and how they work together

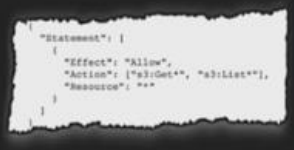
Get to know the tools available to help you with policies

Become a policy ninja!

The policy language

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["s3:Get*", "s3:List*"],
      "Resource": "*"
    }
  ]
}
```

The policy language



- Provides authorization
- Two parts:
  - **Specification**: *Defining* access policies
  - **Enforcement**: *Evaluating* policies

Your job is to define and attach the policies to Users or Roles, make sure that you have granted the right permissions that are correctly scoped down. AWS will evaluate the policies by giving Yes or No to access requests.

Policy specification basics

- ☑ JSON-formatted documents
- ☑ Contain a statement (permissions) that specifies:
  - Which actions a principal can perform
  - Which resources can be accessed

```
{
 "Statement":[{
  "Effect":"effect",
  "Principal":"principal",
  "Action":"action",
  "Resource":"arn",
  "Condition":{
   "condition":{
    "key":"value" }
   }
  }
  ]
 }
```

You can have multiple statements and each statement is comprised of PARC

Principal
Action
Resource
Condition

Principal: Examples

Principal
Action
Resource
Condition

- An entity that is allowed or denied access to a resource
- Indicated by an Amazon Resource Name (ARN)
- With IAM policies, the principal element is implicit (i.e., the user, group, or role attached)

```
<!-- Everyone (anonymous users) -->
"Principal":"AWS":"*.*"

<!-- Specific account or accounts -->
"Principal":{"AWS":"arn:aws:iam::123456789012:root" }
"Principal":{"AWS":"123456789012"}

<!-- Individual IAM user -->
"Principal":"AWS":"arn:aws:iam::123456789012:user/username"

<!-- Federated user (using web identity federation) -->
"Principal":{"Federated":"accounts.google.com"}

<!-- Specific role -->
"Principal":{"AWS":"arn:aws:iam::123456789012:role/rolename"}

<!-- Specific service -->
"Principal":{"Service":"ec2.amazonaws.com"}
```

Replace with your account number

Action: Examples

Principal
Action
Resource
Condition

- Describes the type of access that should be allowed or denied
- You can find actions in the docs or use the policy editor to get a drop-down list
- Statements must include either an `Action` or `NotAction` element

```
<!-- EC2 action -->
"Action":"ec2:StartInstances"

<!-- IAM action -->
"Action":"iam:ChangePassword"

<!- Amazon S3 action -->
"Action":"s3:GetObject"

<!-- Specify multiple values for the Action element-->
"Action":["sqs:SendMessage","sqs:ReceiveMessage"]

<-- Wildcards (* or ?) in the action name. Below covers create/delete/list/update-->
"Action":"iam:*AccessKey*"
```

Every policy statement must have an Action or a NotAction.

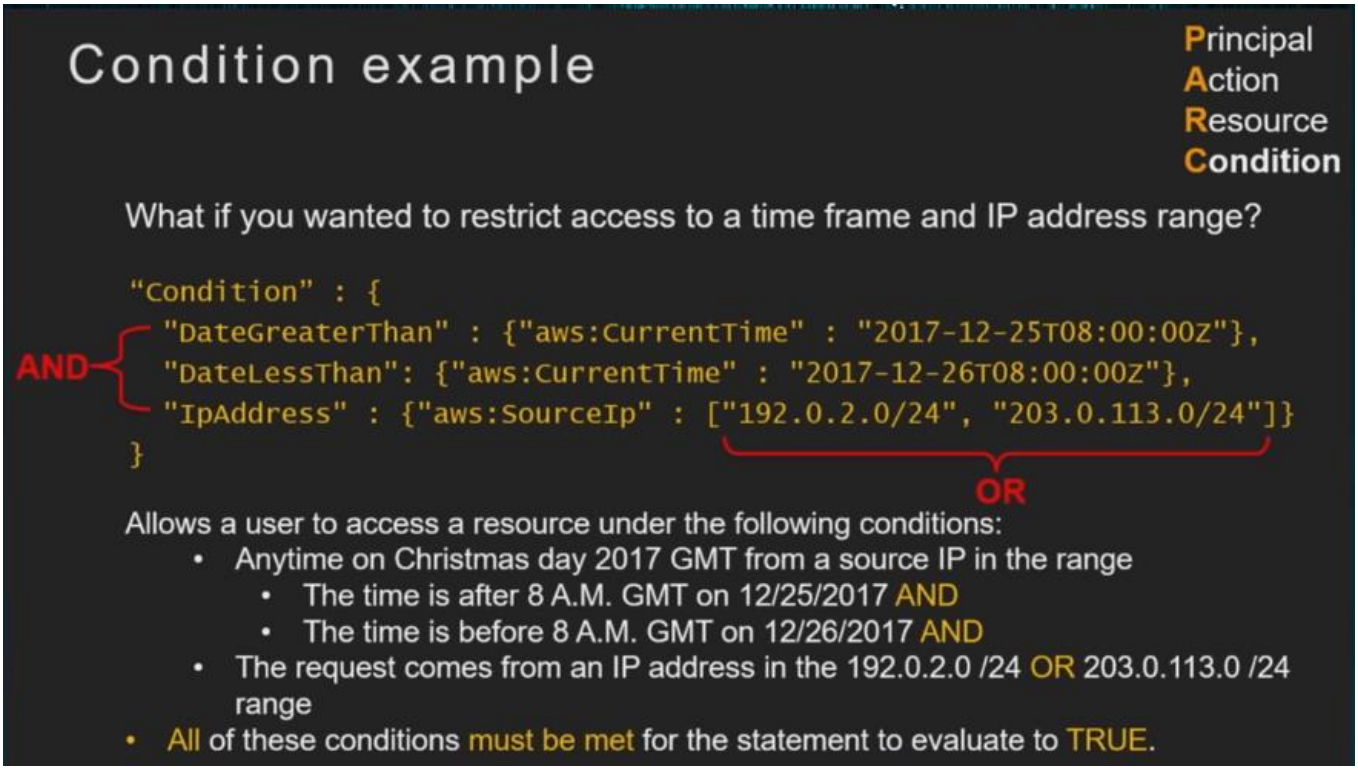
Resource: Examples

Principal
Action
Resource
Condition

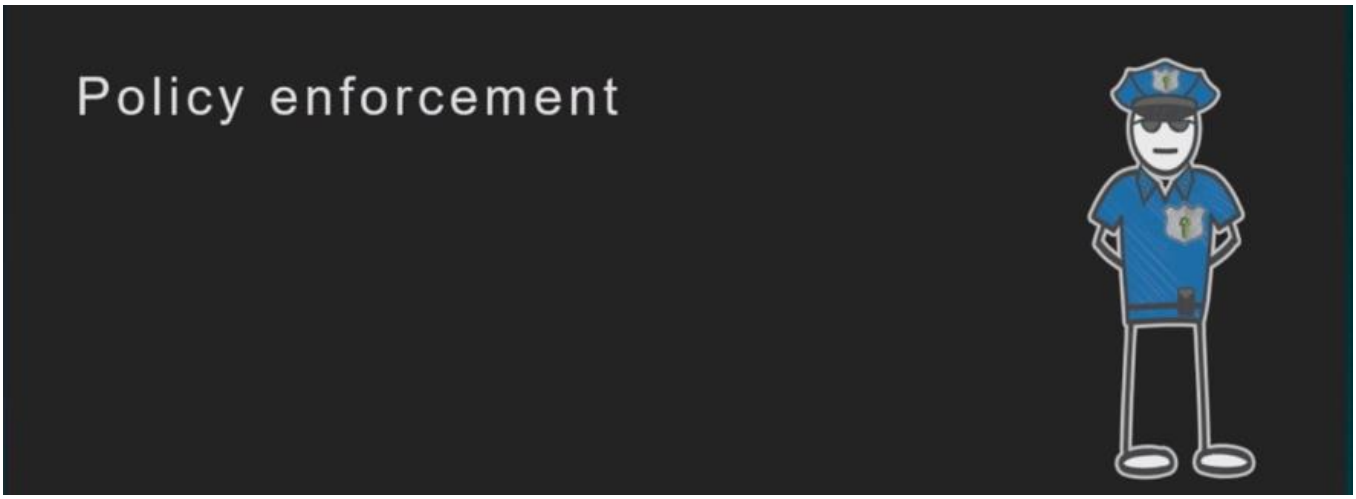
- The object or objects being requested
- Statements must include either a `Resource` or a `NotResource` element

```
<-- S3 bucket -->
"Resource":"arn:aws:s3:::my_corporate_bucket/*"

<-- All S3 buckets, except this one -->
"NotResource":"arn:aws:s3:::security_logging_bucket/*"

<-- Amazon SQS queue-->
"Resource":"arn:aws:sqs:us-west-2:123456789012:queue1"

<-- Multiple Amazon DynamoDB tables -->
"Resource":["arn:aws:dynamodb:us-west-2:123456789012:table/books_table",
            "arn:aws:dynamodb:us-west-2:123456789012:table/magazines_table"]

<-- All EC2 instances for an account in a region -->
"Resource": "arn:aws:ec2:us-east-1:123456789012:instance/*"
```

Replace with your account number



If you have a complex condition, you should break it out into multiple conditions that must all evaluate to be true.



This is what AWS has responsibility to enforce

Policy enforcement

1. Decision starts at Deny
2. Evaluate all applicable policies
3. Is there an explicit Deny?
4. Is there an Allow?
5. Final decision ="Deny" (default Deny)

- AWS retrieves all policies associated with the user and resource
- Only policies that match the action and conditions are evaluated

No

No

Yes

Yes

Final decision ="Deny" (explicit Deny)

- If a policy statement has a Deny, it trumps all other policy statements

Final decision ="Allow"

- Access is granted if there is an explicit Allow and no Deny

- By default, an implicit (default) Deny is returned

It's not just IAM policies

AWS Organizations
Service control policies (SCPs)

AWS Identity and Access Management
Inline policies
Managed policies

AWS Security Token Service (STS)
Scoped-down policies

Specific AWS services
Resource-based policies

Example: Amazon S3 bucket policies

All use the same policy language

What is AWS Organizations?

When to use each type of permission policy

AWS Organizations
Service control policies (SCPs)

Guardrails on the account to disable access to services

AWS IAM
Inline policies
Managed policies

Set granular permissions based on functions that users or applications need to perform

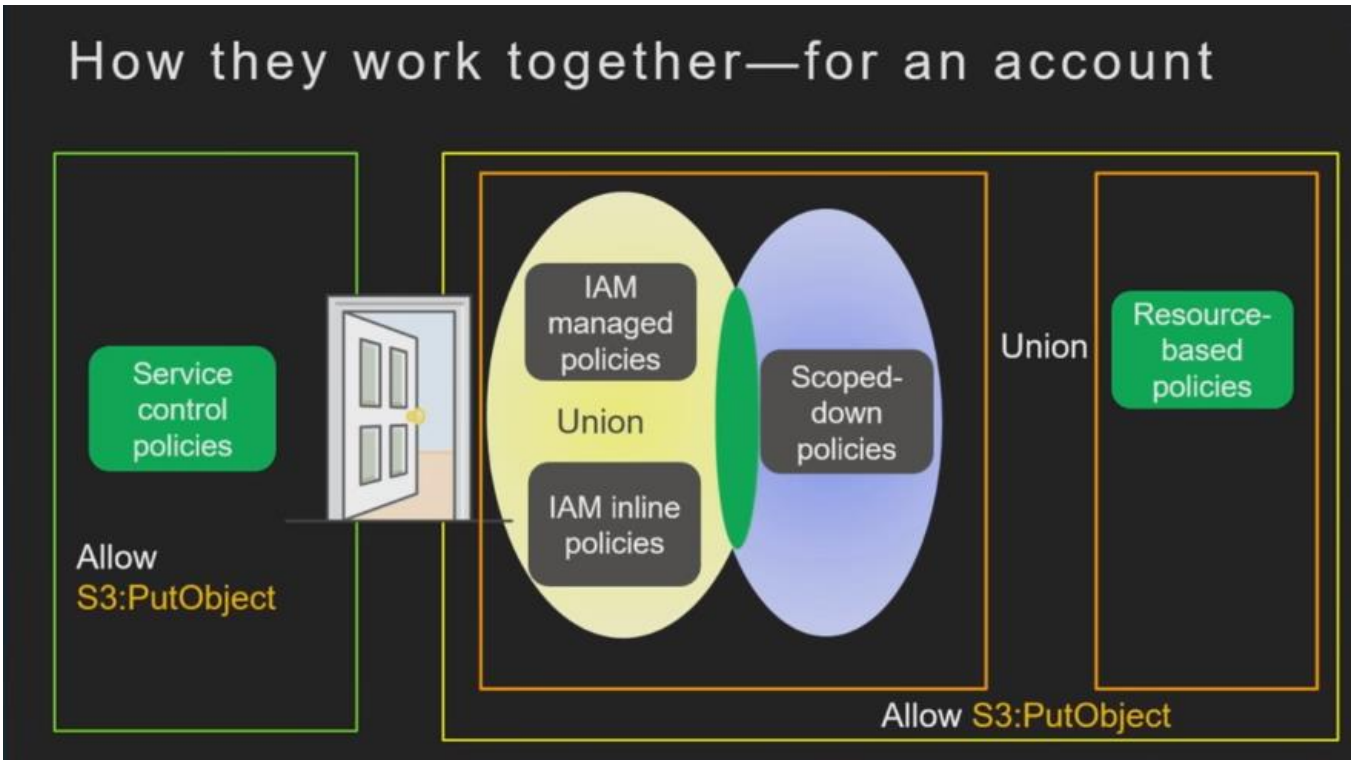
AWS STS
Scoped-down policies

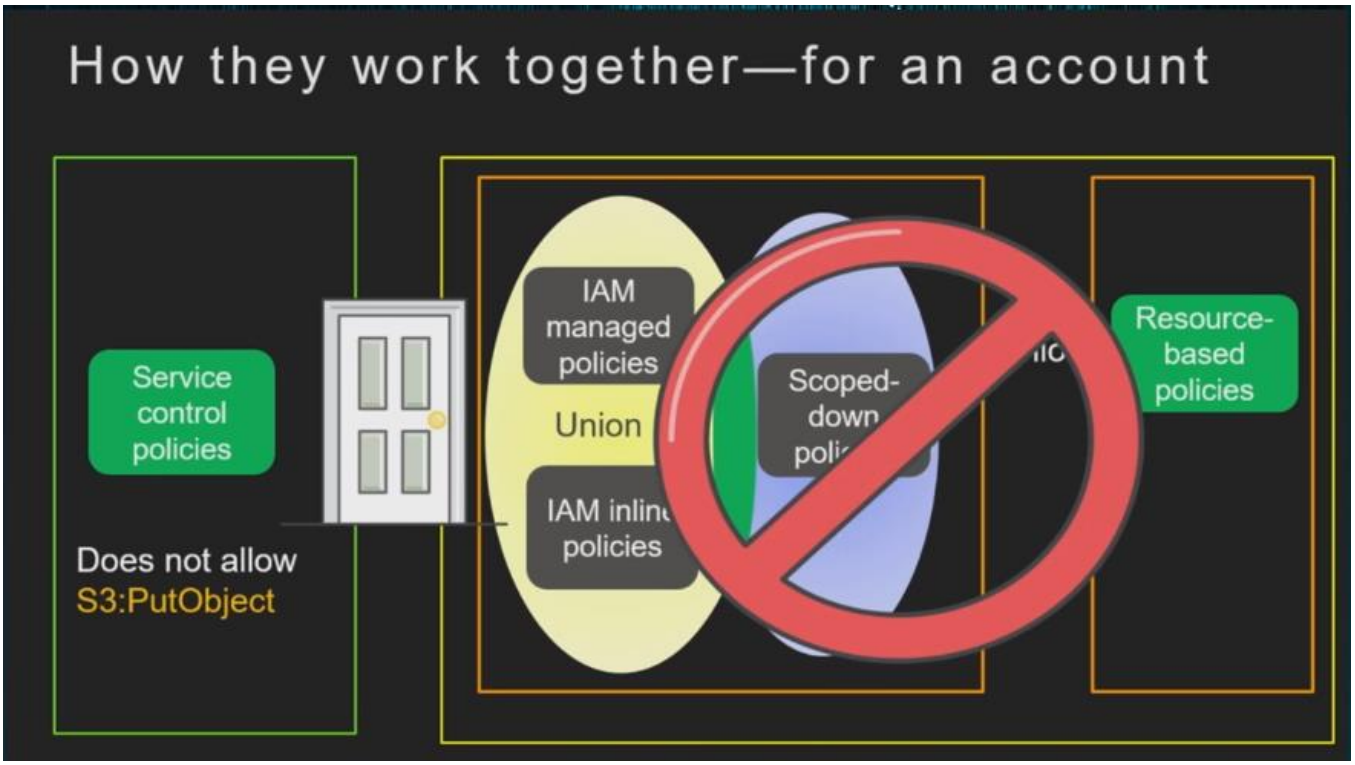
Reduce general shared permissions further

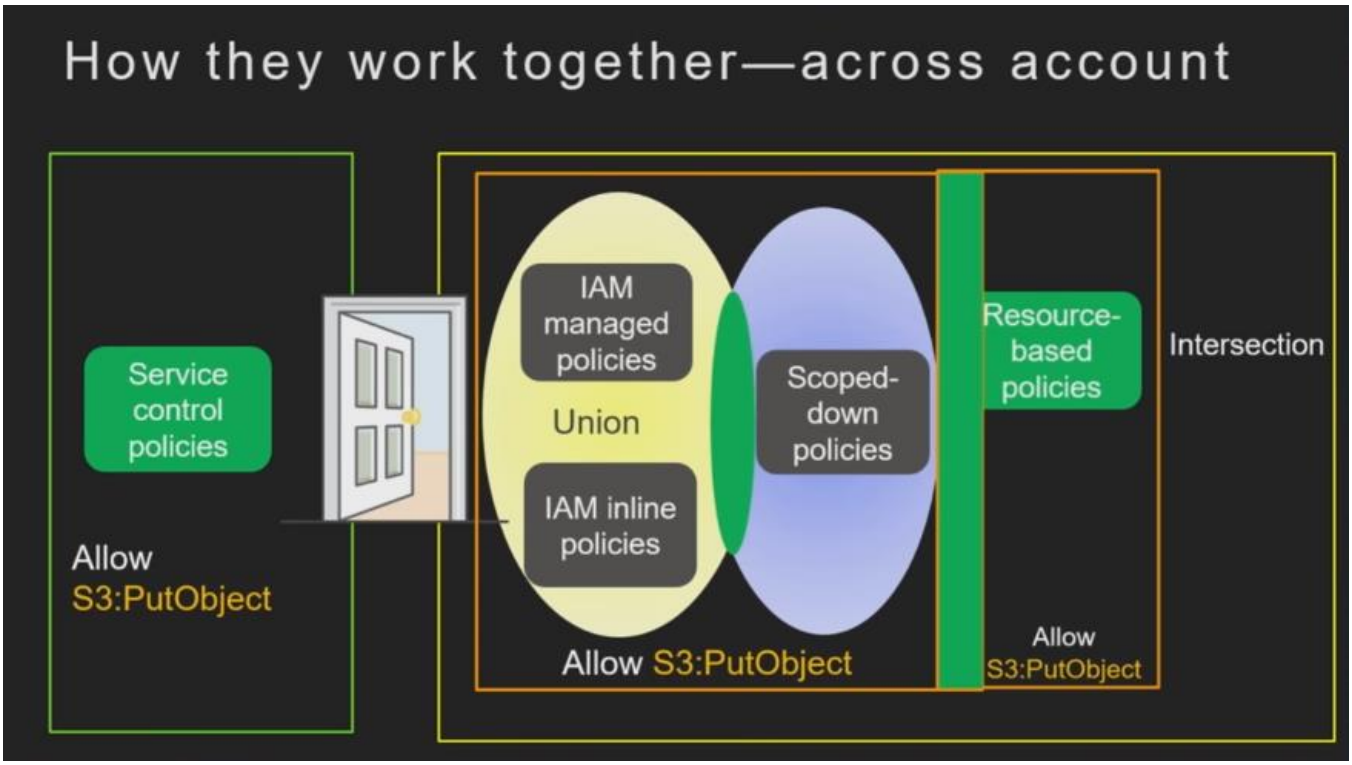
Specific AWS services
Resource-based policies

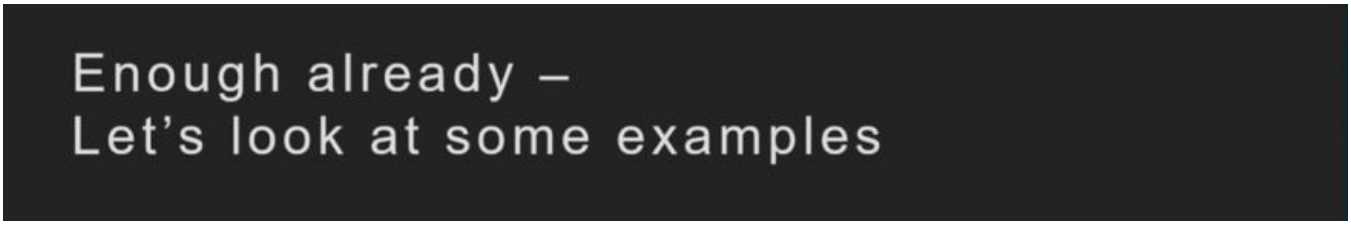
Example: S3 bucket policies

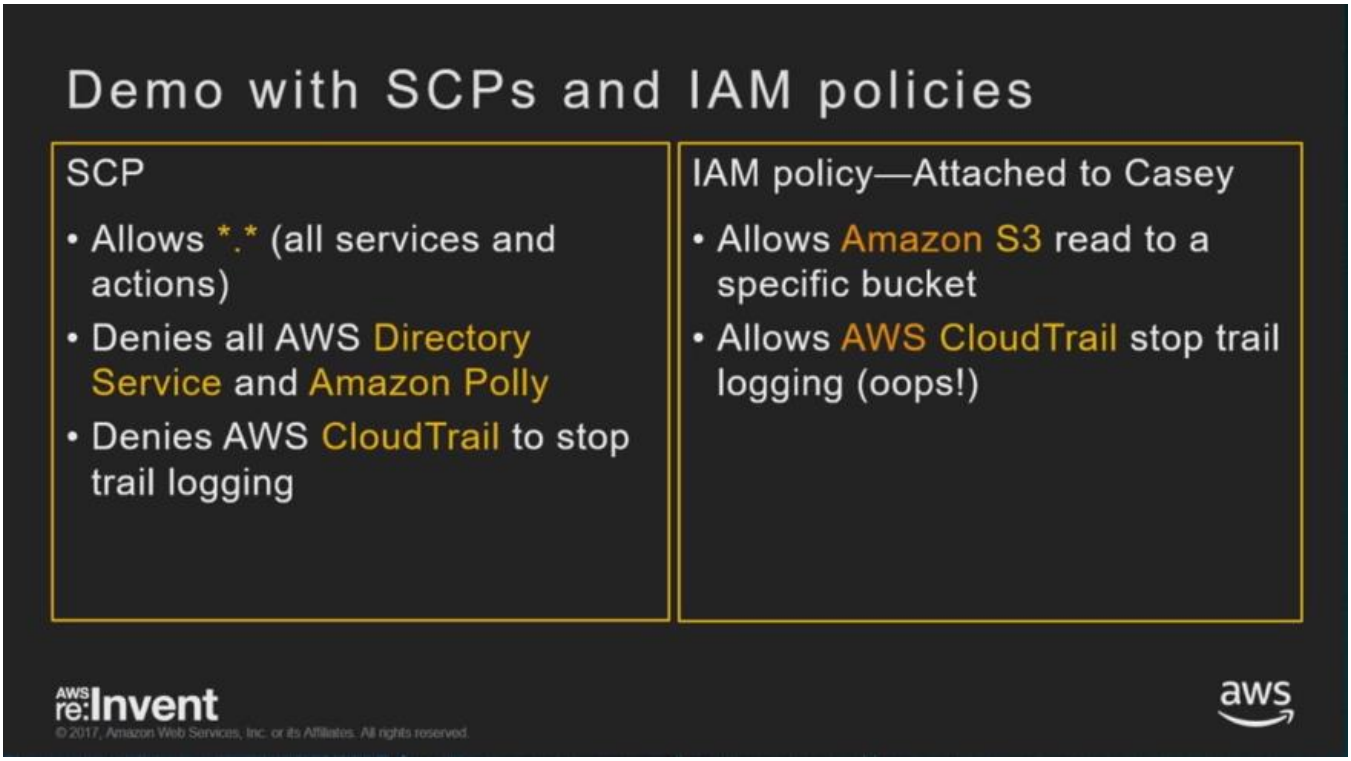
Cross-account access and to control access from the resource


How they work together—for an account
Service control policies
Allow S3:PutObject
IAM managed policies
Union
IAM inline policies
Scoped-down policies
Union
Resource-based policies
Allow S3:PutObject


How they work together—for an account
Service control policies
Does not allow S3:PutObject
IAM managed policies
Union
Resource-based policies







We have a ***Master account*** and we have ***Dev***, ***Stage*** and ***Prod*** accounts. We are mainly going to be working in the Dev account. We have put all the 4 accounts in an Organization and put the SCP policy on it.

SCP

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "DenyUnapprovedActions",
        "Effect": "Deny",
        "Action": ["polly:*",
            "ds:*",
            "cloudtrail:StopLogging"
        ],
        "Resource": ["*"]
    }]
}
```

Don't Forget!

*We also have an Allow *.* policy attached to this OU*

This is what the policy looks like.

IAM policy – Attached to Casey – S3

```
{
     "Version": "2012-10-17",
     "Statement": [{
          "Effect": "Allow",
          "Action": ["s3:GetObject", "s3:ListBucket"],
          "Resource": [
                     "arn:aws:s3:::reinvent-2017-policy-ninja-dev/*",
                     "arn:aws:s3:::reinvent-2017-policy-ninja-dev"]
     }, {
          "Effect": "Allow",
          "Action": ["s3:ListAllMyBuckets", "s3:HeadBucket", "s3:ListObjects"],
          "Resource": "*"
     }]
}
```

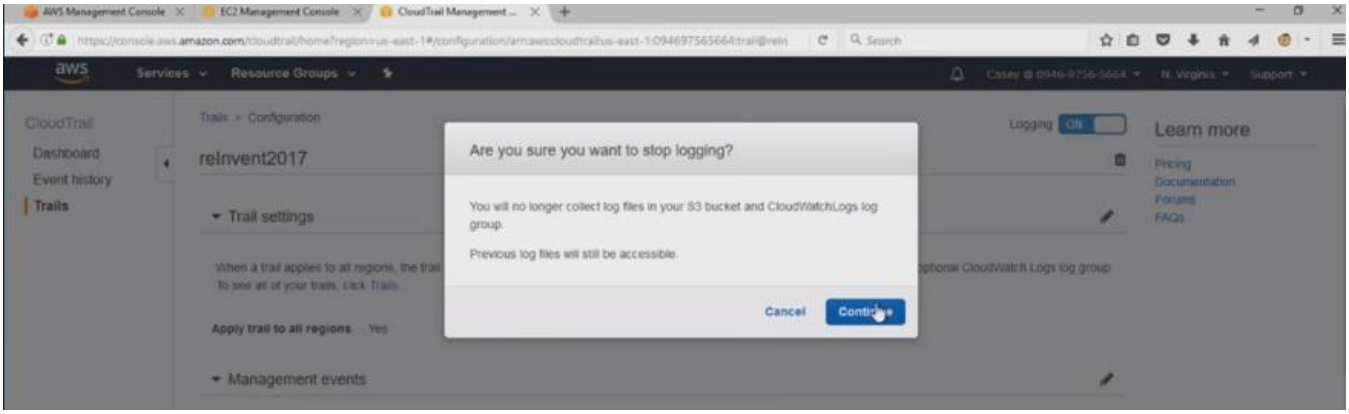
IAM policy – Attached to Casey – CloudTrail

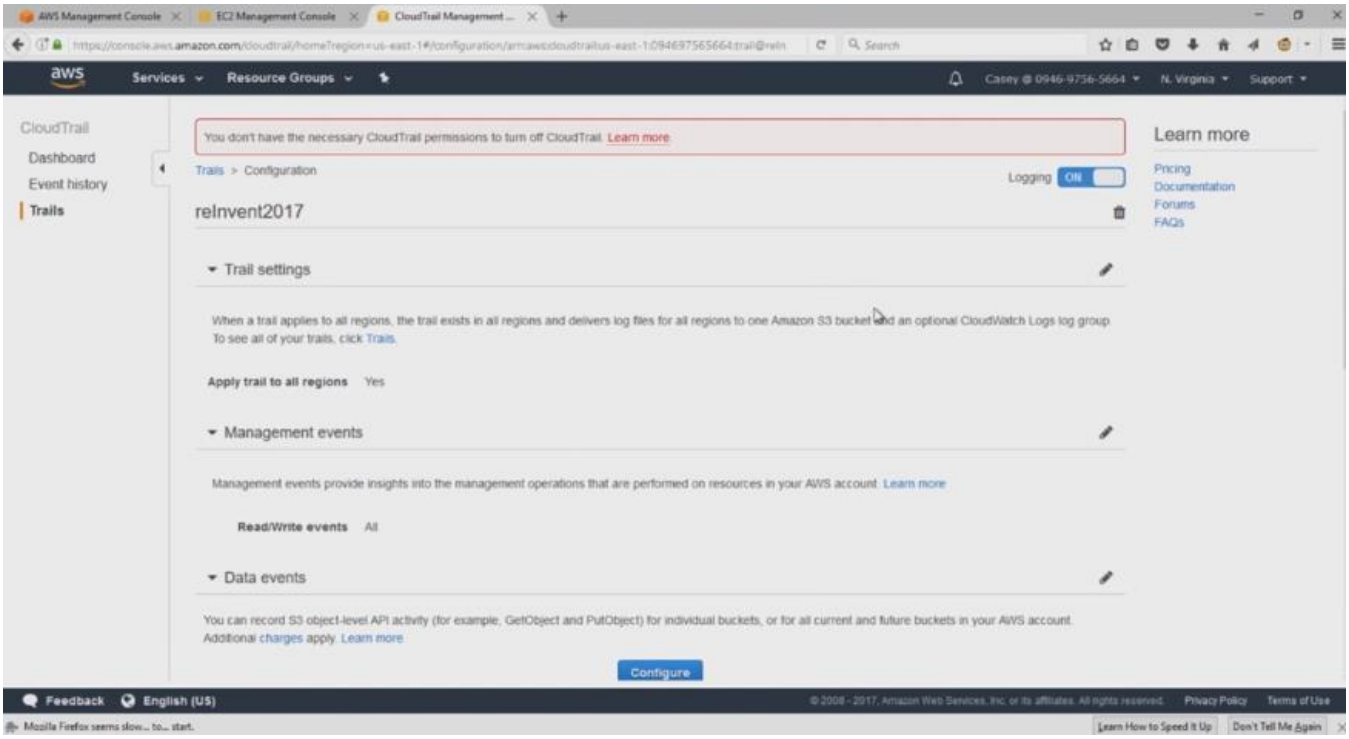
```
"Version": "2012-10-17",
"Statement": [{
    "Sid": "VisualEditor0",
    "Effect": "Allow",
    "Action": [
        "cloudtrail:LookupEvents",
        "cloudtrail:StopLogging",
        "cloudtrail:ListPublicKeys",
        "cloudtrail:ListTags",
        "cloudtrail:GetTrailStatus",
        "cloudtrail:GetEventSelectors",
        "cloudtrail:DescribeTrails"
    ],
    "Resource": "*"
}] }
```

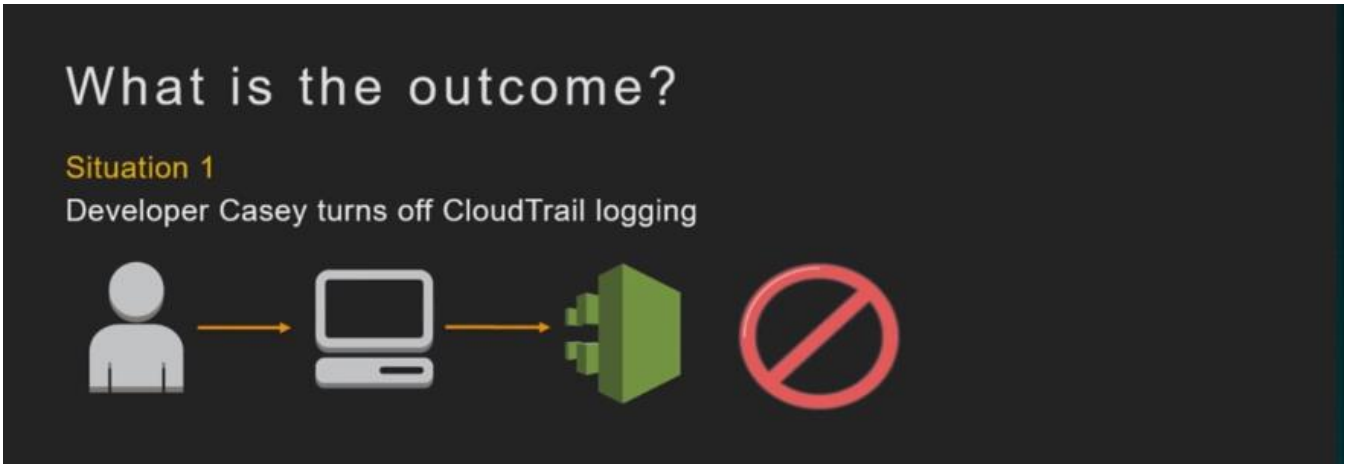
What is the outcome?

Situation 1

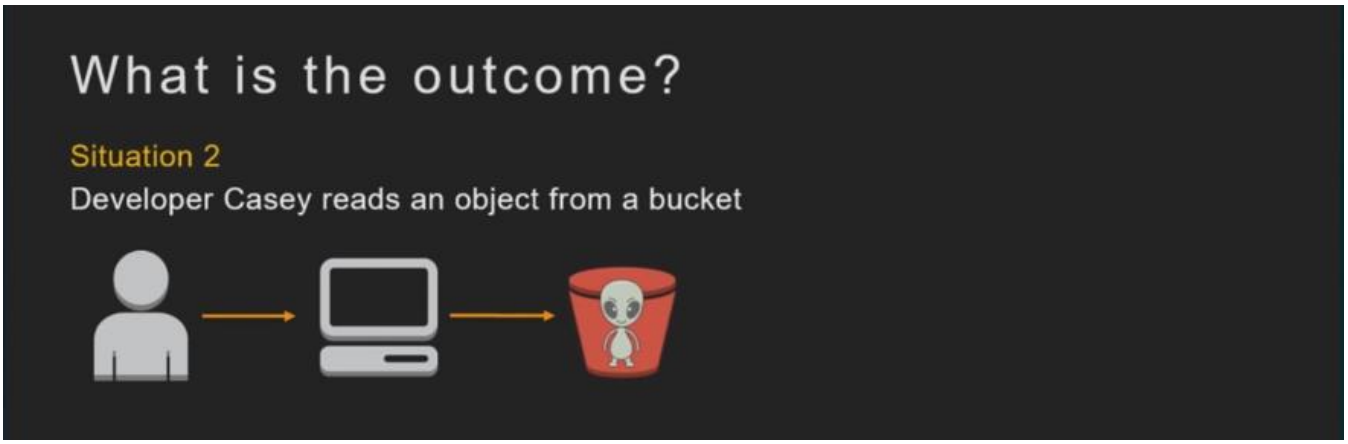
Developer Casey turns off CloudTrail logging



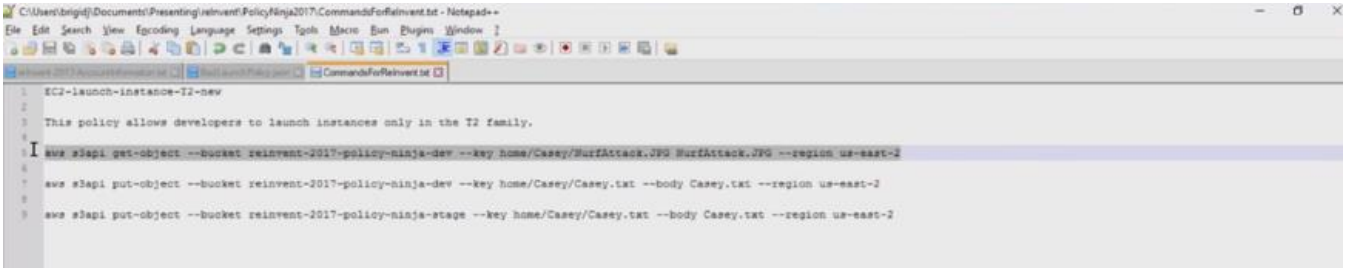


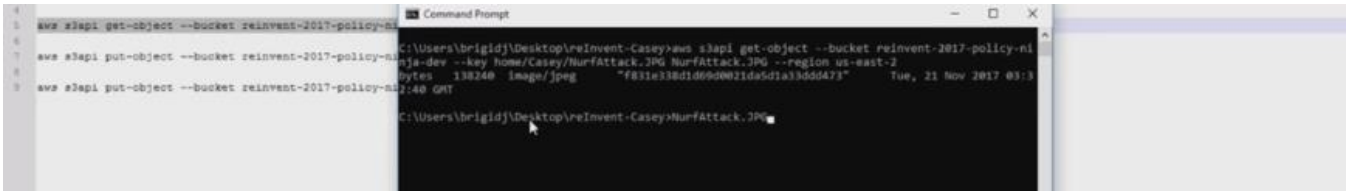


Casey can't do anything with CloudTrail

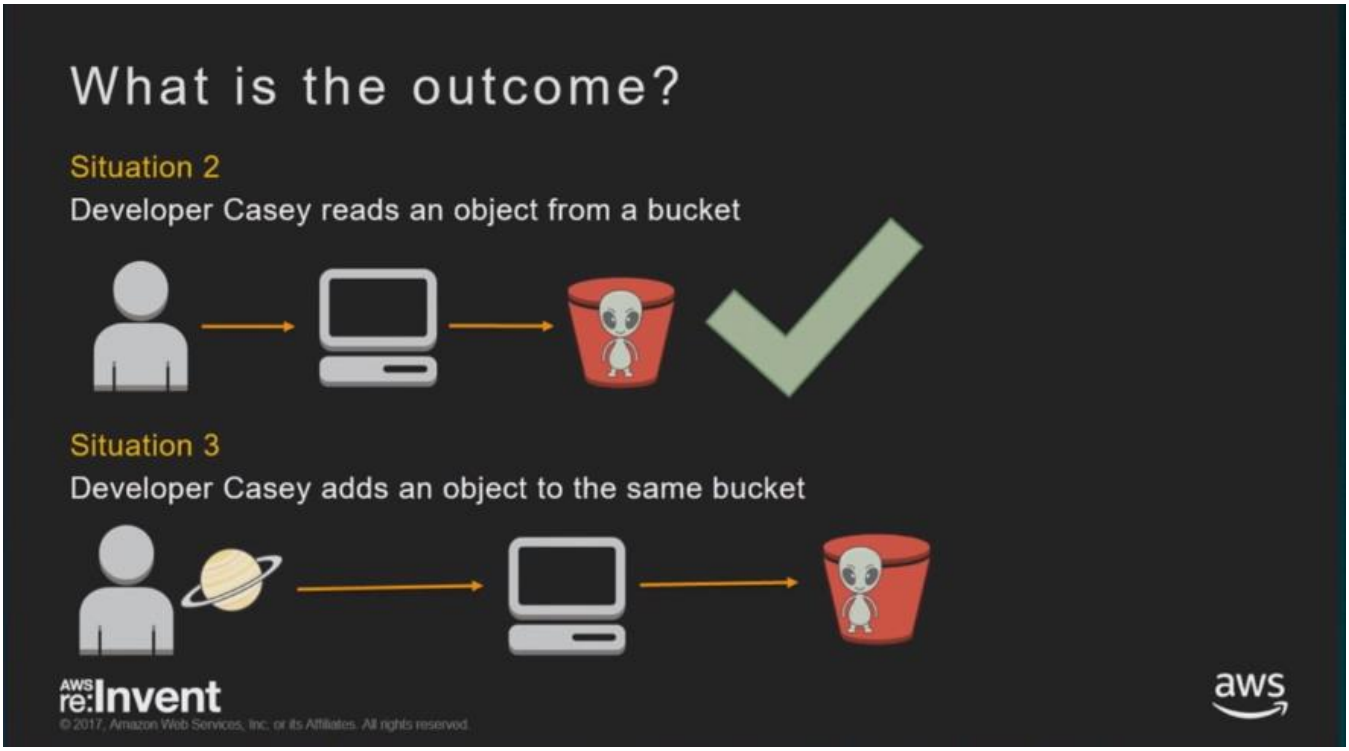


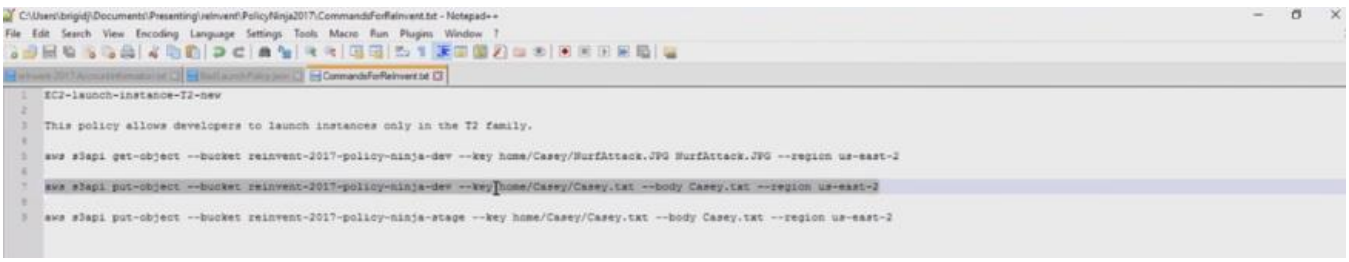
We are going to use the command line for this as below

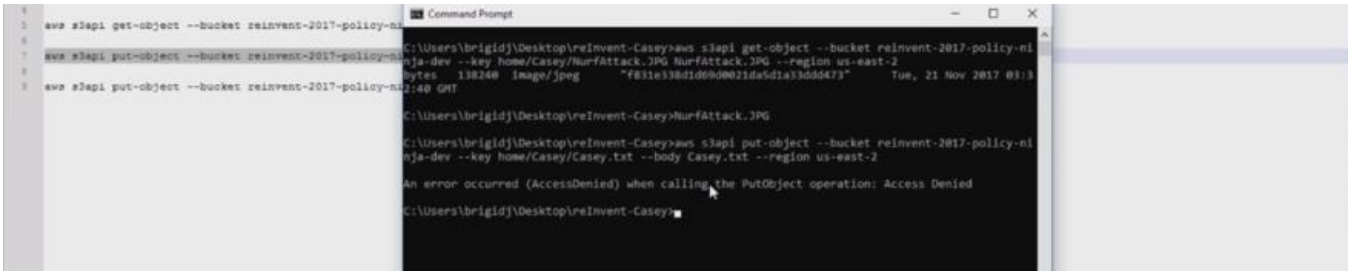


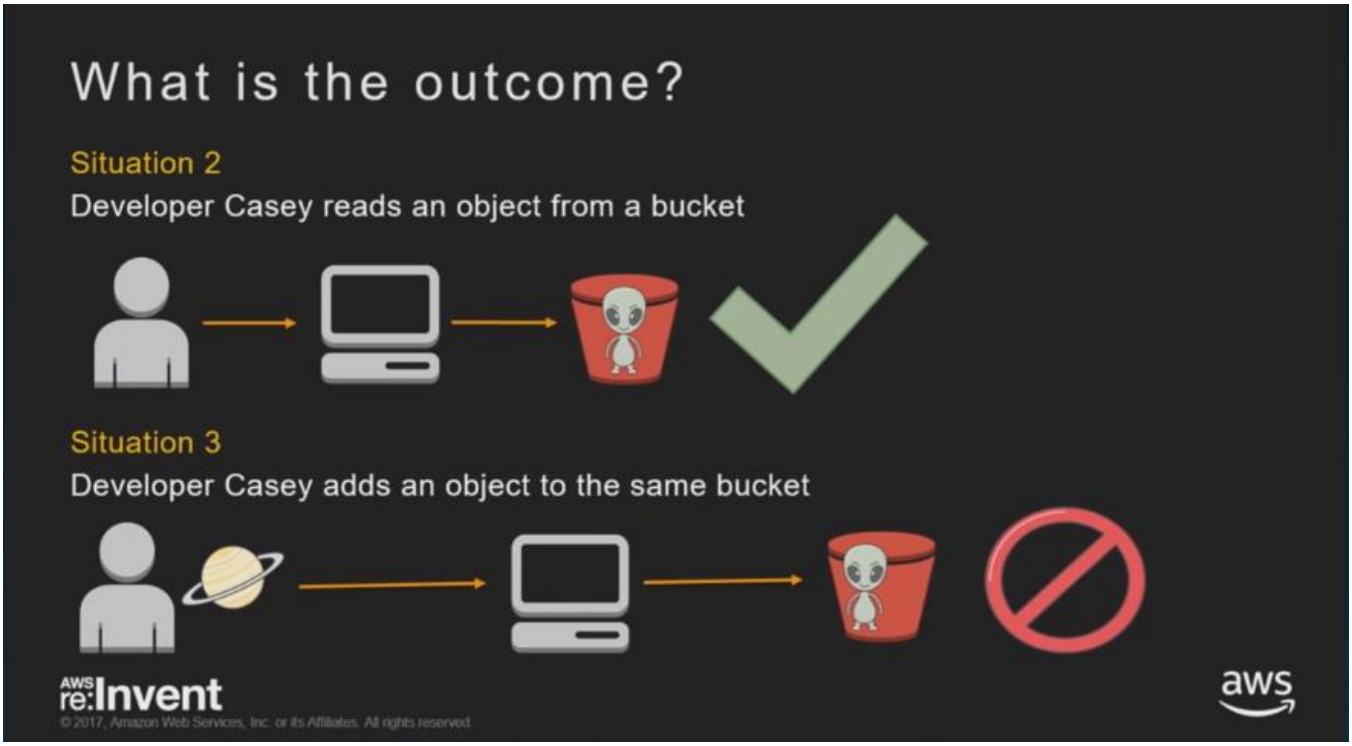


Casey can download/read a file from the bucket

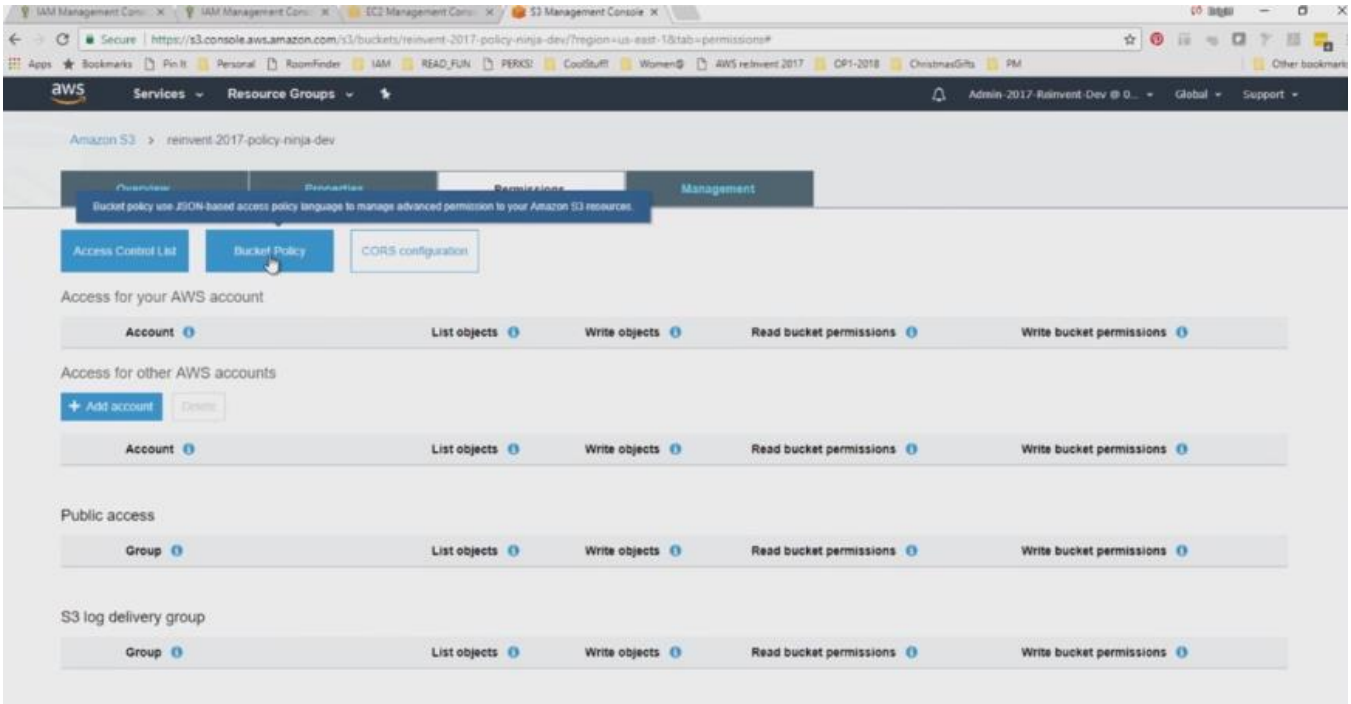


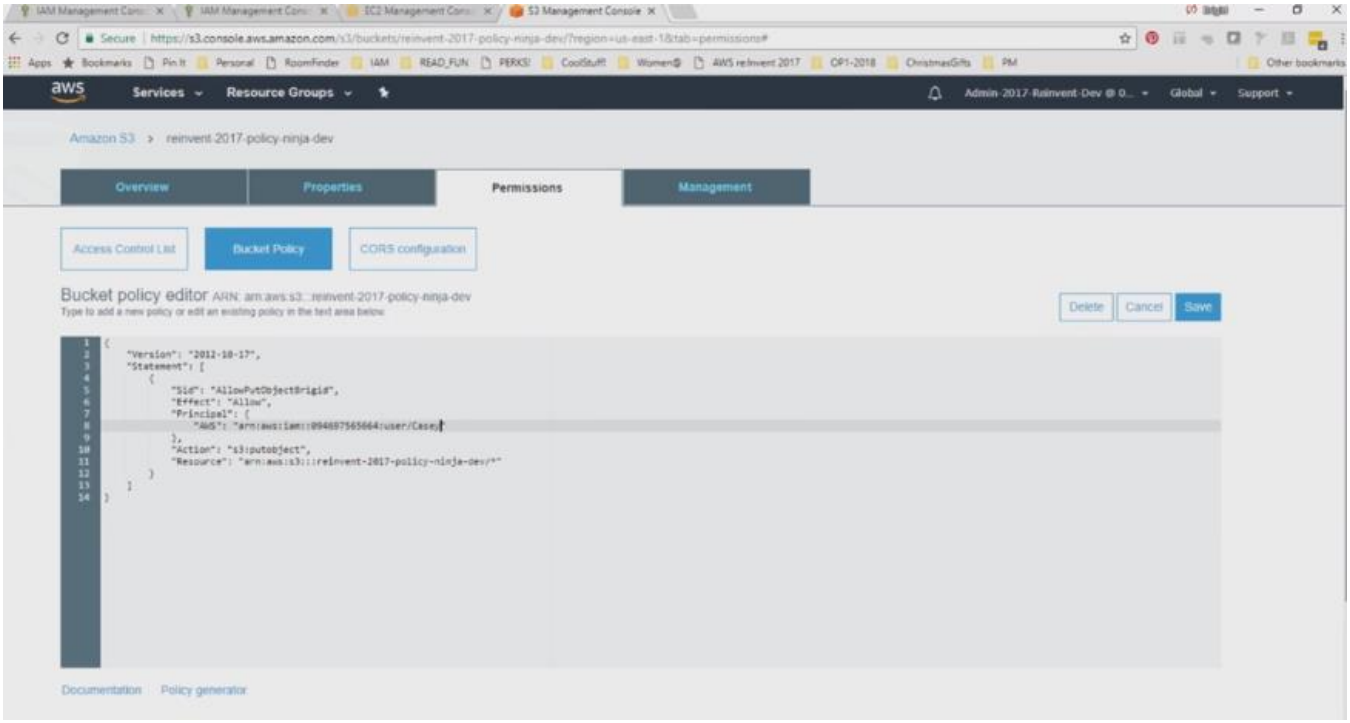






No PutObject permission is granted to Casey





Demo with SCP, IAM, and bucket policy

SCP policy

- Allows *.*
- Denies all Directory Service and Amazon Polly
- Denies CloudTrail to stop trail logging

IAM policy – Attached to Casey

- Allows S3 read to a specific bucket

S3 bucket policy

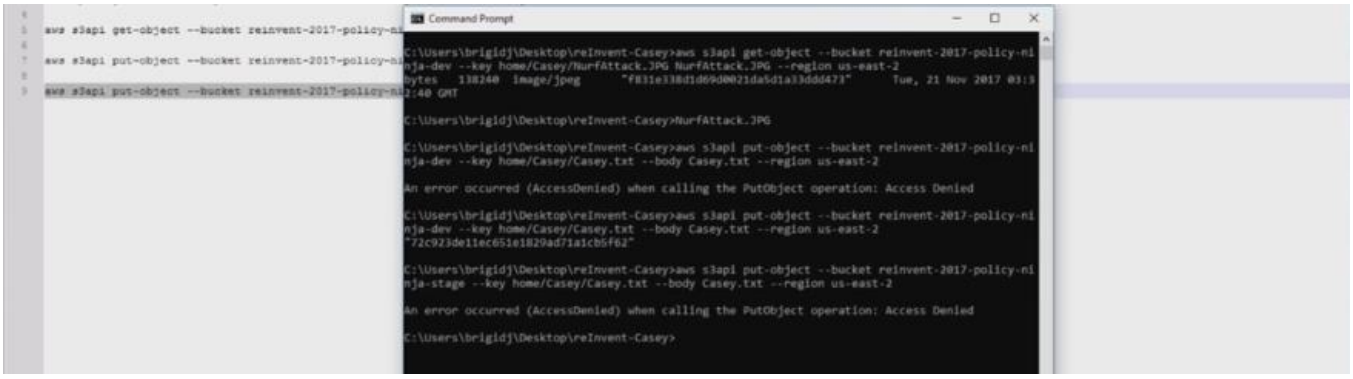
- Allows S3 write for Casey

Now we add a bucket policy

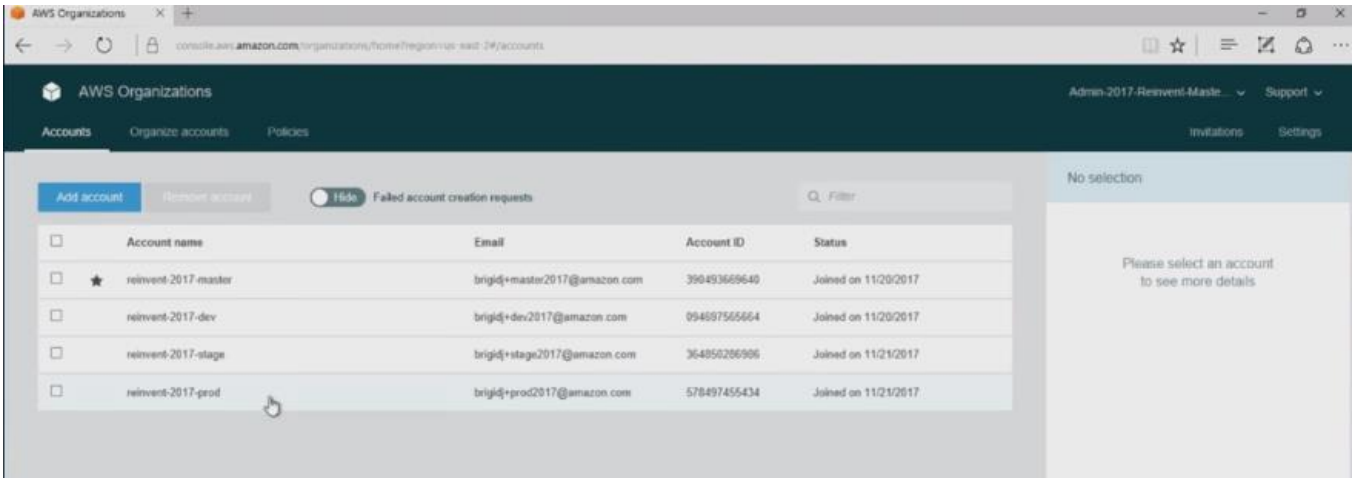
```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Sid": "AllowPutObject",
        "Effect": "Allow",
        "Principal": {
            "AWS": "arn:aws:iam::123456789123:user/Casey"
        },
        "Action": "s3:putobject",
        "Resource": "arn:aws:s3:::reinvent-2017-policy-ninja-dev/*"
    }]
}
```

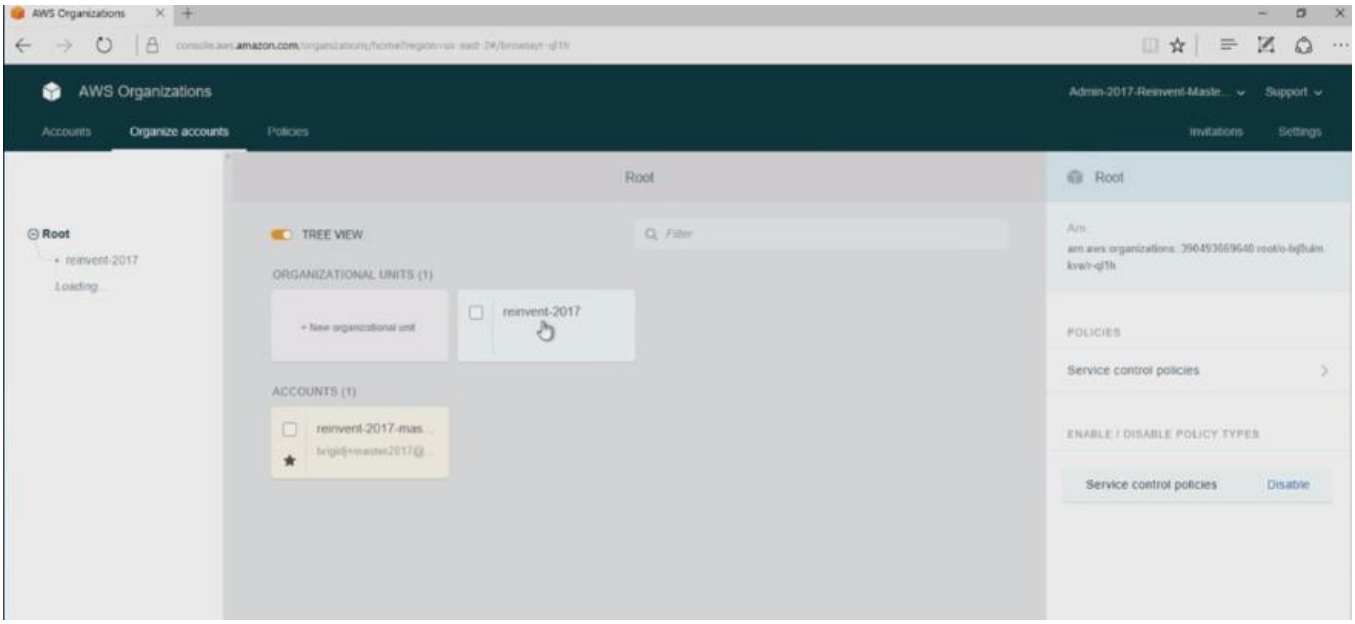
Now Casey can put images into the S3 bucket

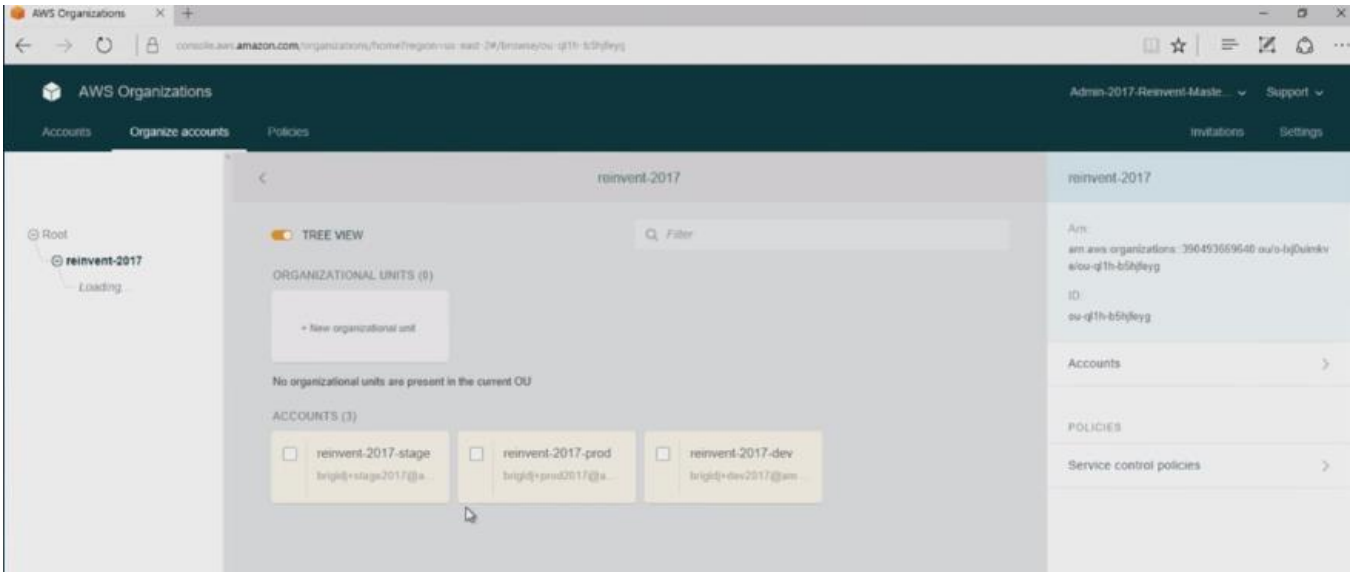
We can let Casey try to put the image into the Staging account and not the Dev account

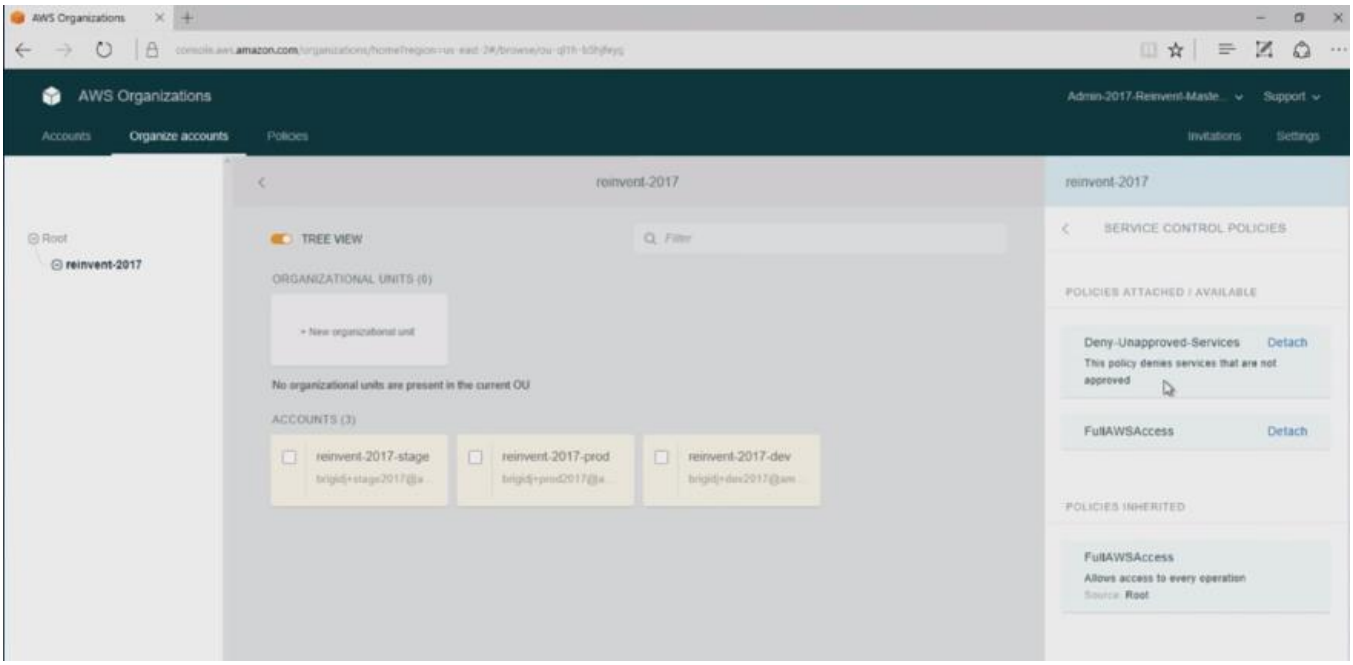


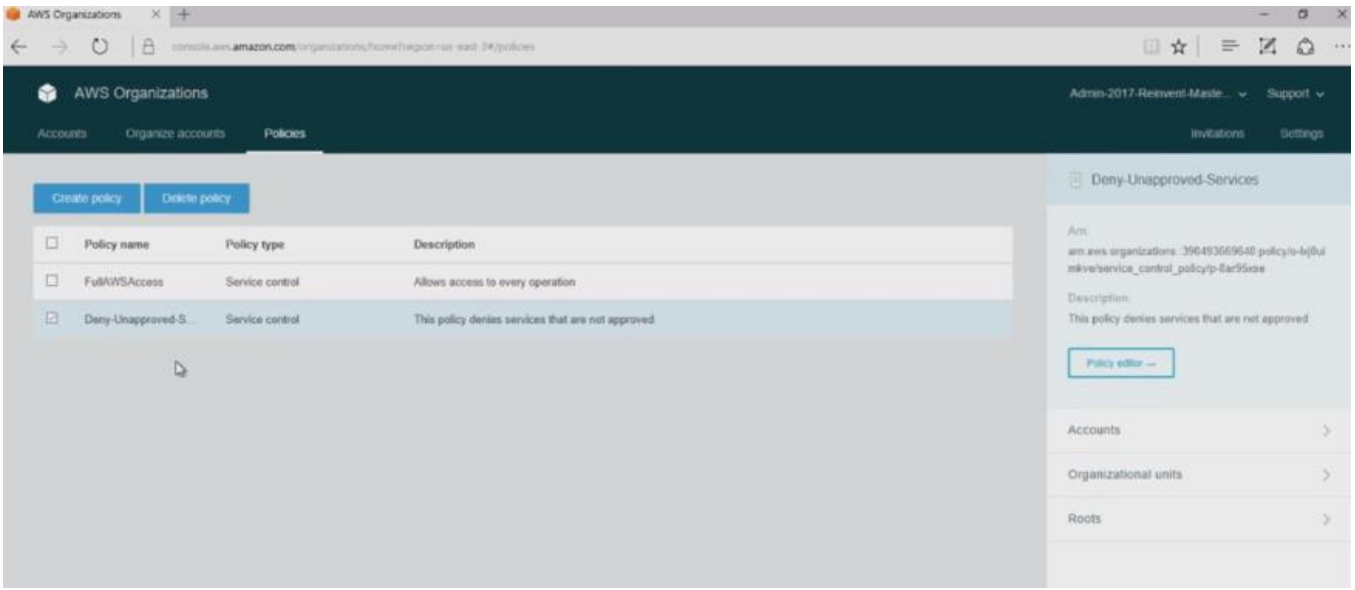
Casey is denied access to put objects into the staging S3 bucket, the only way to fix this is to grant access via IAM or a new policy.


AWS Organizations
Admin-2017-Reinvent-Maste...
Support
Accounts
Organize accounts
Policies
Invitations
Settings
Add account
Remove account
Hide
Failed account creation requests
Filter
Account name
Email
Account ID
Status
reinvent-2017-master
brigidj+master2017@amazon.com
390493669640
Joined on 11/20/2017
reinvent-2017-dev
brigidj+dev2017@amazon.com
094697565664
Joined on 11/20/2017
reinvent-2017-stage
brigidj+stage2017@amazon.com
364850286906
Joined on 11/21/2017
reinvent-2017-prod
brigidj+prod2017@amazon.com
578497455434
Joined on 11/21/2017
No selection
Please select an account
to see more details

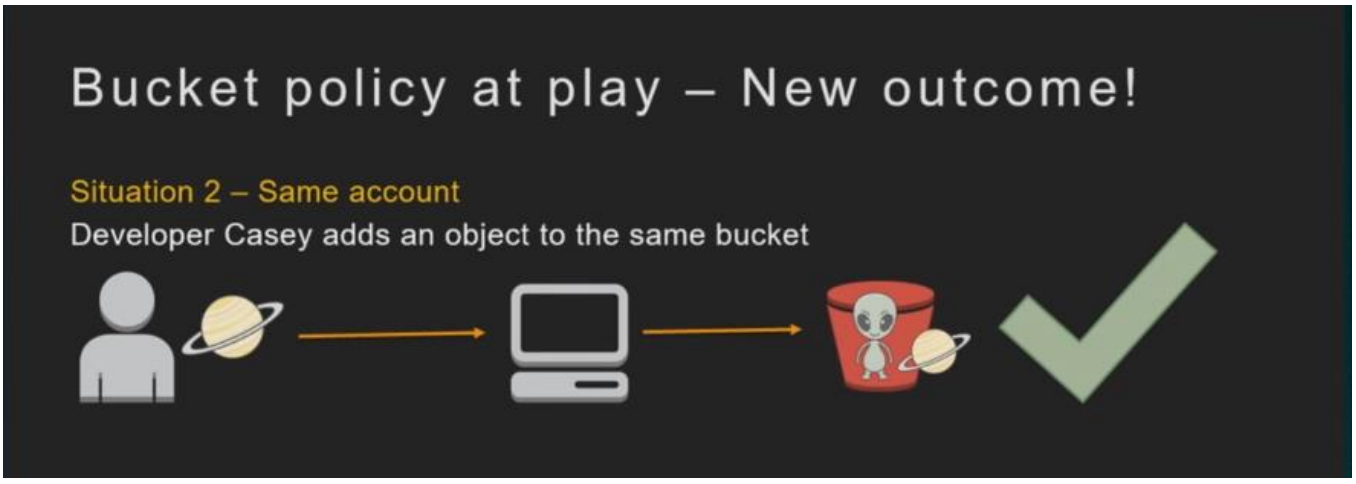

AWS Organizations
Admin-2017-Reinvent-Maste...
Support
Accounts
Organize accounts
Policies
Invitations
Settings
Root
Root
reinvent-2017
Loading...
TREE VIEW
Filter
ORGANIZATIONAL UNITS (1)
+ New organizational unit
reinvent-2017
ACCOUNTS (1)
reinvent-2017-mas...
Root
Arn:
POLICIES
Service control policies
ENABLE / DISABLE POLICY TYPES
Service control policies
Disable


AWS Organizations
Admin-2017-Reinvent-Maste...
Support
Accounts
Organize accounts
Policies
Invitations
Settings
reinvent-2017
Root
reinvent-2017
Loading...
TREE VIEW
Filter
ORGANIZATIONAL UNITS (0)
+ New organizational unit
No organizational units are present in the current OU
ACCOUNTS (3)
reinvent-2017-stage
reinvent-2017-prod
reinvent-2017-dev
reinvent-2017
Arn:
ID:
Accounts
POLICIES
Service control policies





This is how you can go and create the multi-account setup and allow or deny access as needed.



What about cross-account?

SCP policy

- Allows *.*
- Denies all Directory Service and Amazon Polly.
- Denies CloudTrail to stop trail logging

IAM policy – Attached to Casey

- Allows S3 read to a specific bucket

S3 bucket policy in **a different account**

- Allows S3 write for Casey

Cross-account S3 bucket

Situation 2 – Different account

Developer Casey adds an object to a bucket in a different account

Recap

- Review of IAM policy language
- Review of how policies work together to control access
- Policy tools – Create, test, and correct policies
- Policy Foo – Ninja moves with Amazon EC2
- Additional resources

Policy Tools – Policy summaries

An easier way to understand the permissions your policies grant

- Summary of service and access level
- Four types of access levels for actions
  - List – See a list of resources
  - Read – Read the content in resources
  - Write – Create, delete, or modify resources
  - Permissions – Grant or modify permissions to resources
- Show remaining services and actions
- Identify and correct errors in your policies

Policy tools – Visual policy editor

Point and click your way through granting permissions in AWS

- Create and modify IAM policies
- Select from a list of AWS services
- Select from a list of available actions
- Explore services and actions with inline documentation
- Understand the resources you can specify for service actions
- Select from a list of available conditions for actions you select
- Easily include dependent actions required to perform tasks in AWS
- Import existing policies and modify

Demo: Correct an Existing Policy

Goal: You have an existing policy that enables developers to use the EC2 console to:

- Start, stop, and terminate instances with a specific tag

Situation: It does not work (oh, bother!)

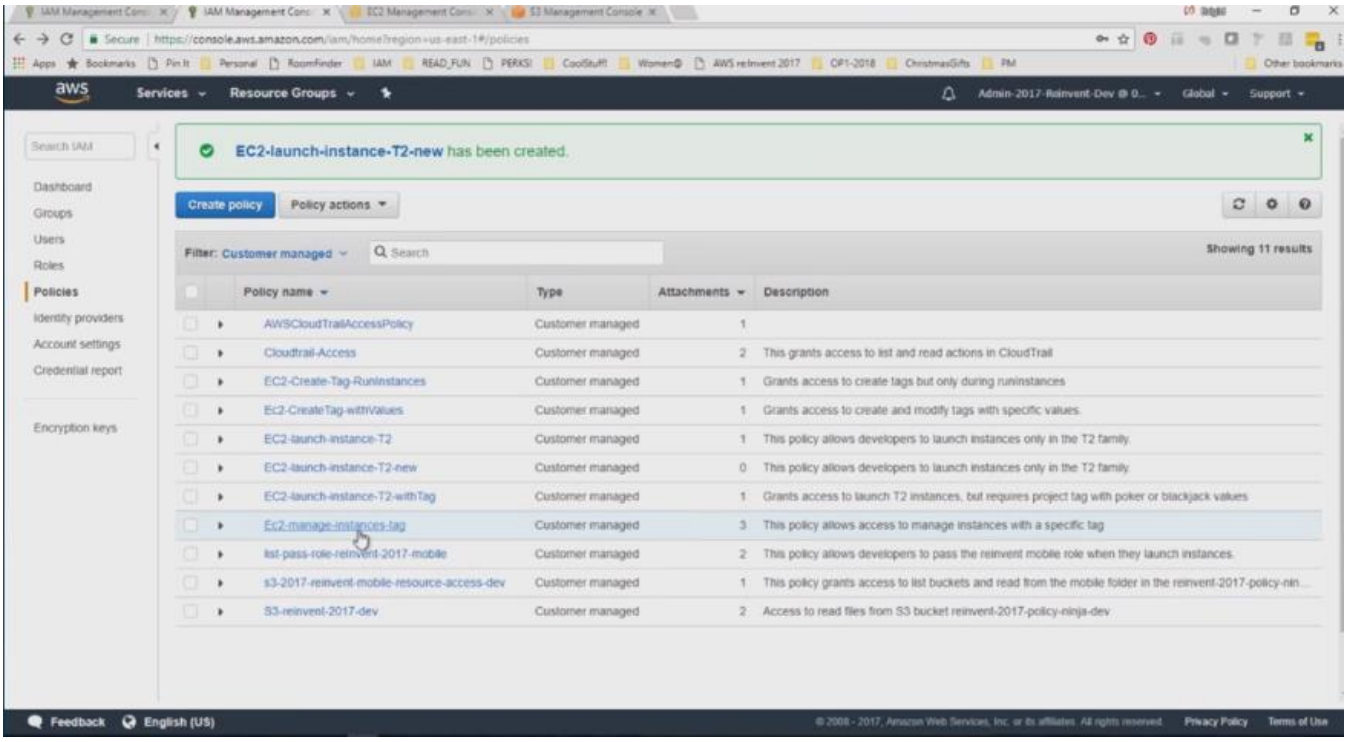
Pro tip! Use policy summaries and the editor to correct your existing policies

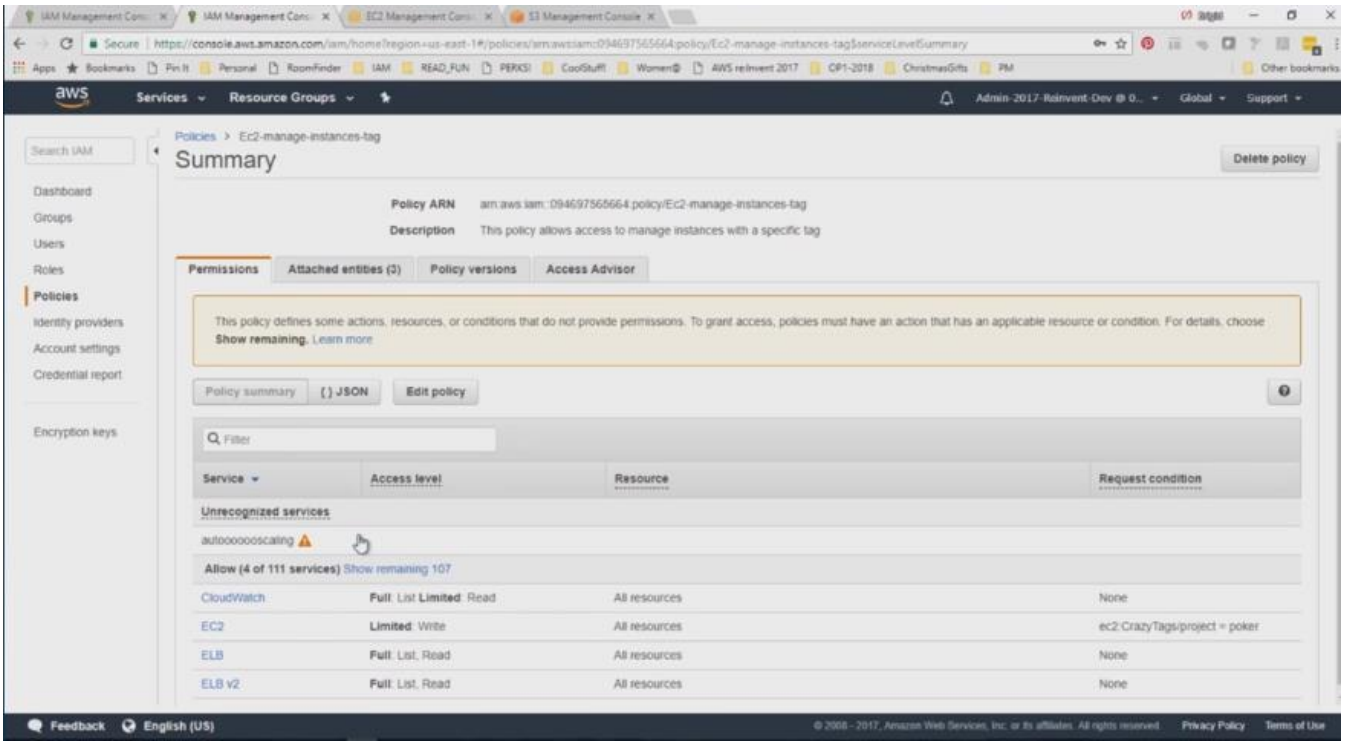
Existing policy – Does not work

```
{   "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "ec2:Describe*",
            "Resource": "arn:aws:ec2:region:account-id:security-group/sg-12345678"
        },
        {   "Effect": "Allow",
            "Action": ["elasticloadbalancing:Describe*", "cloudwatch:ListMetrics"
                "cloudwatch:GetMetricStatistics", "cloudwatch:Describe*",
                "autooooooscaling:Describe*"
            ],
            "Resource": "*"
        }, <continued>
```

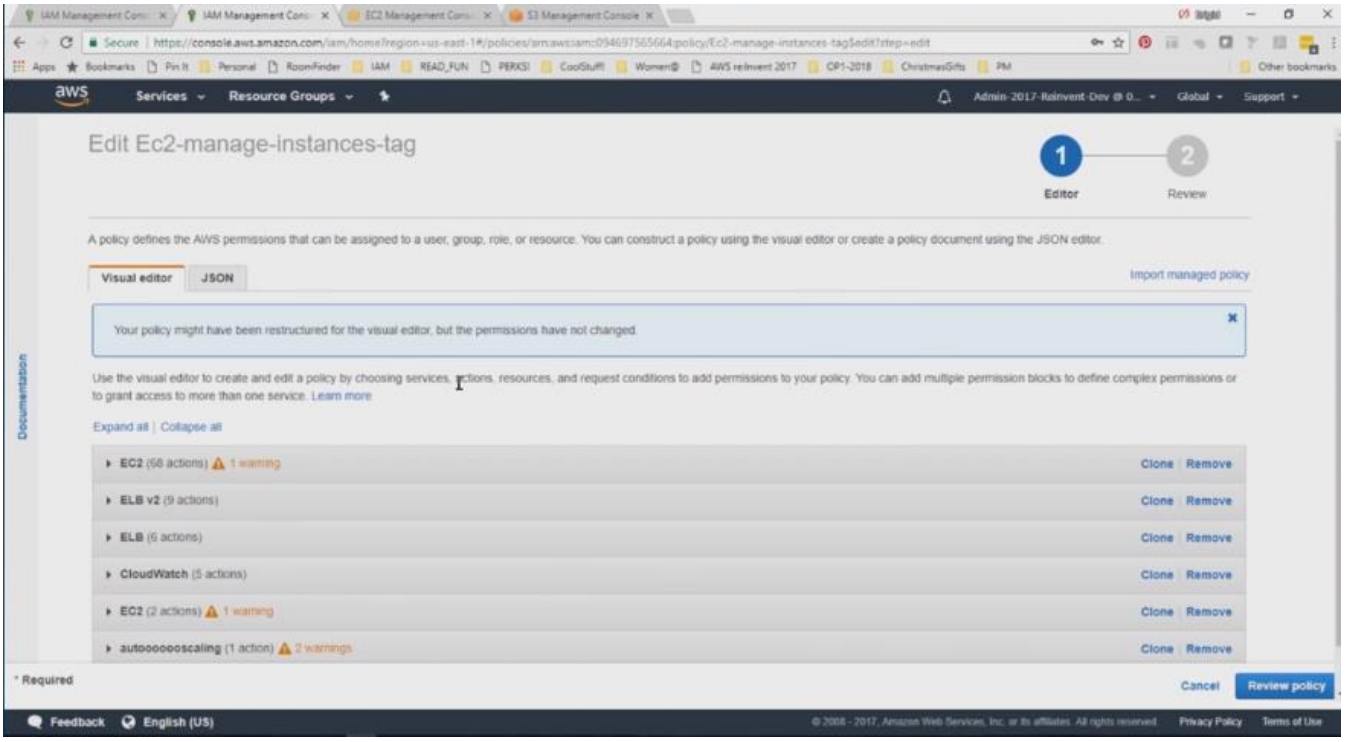
Proposed policy – Does not work

```
        …{
            "Effect": "Allow",
            "Action": ["ec2:StartInstances", "ec2:StopInstances"],
            "Condition": {
                "StringEquals": {
                    "ec2:CrazyTags/project": "poker"
                }
            },
            "Resource": "*"
        }
    ]
}
```

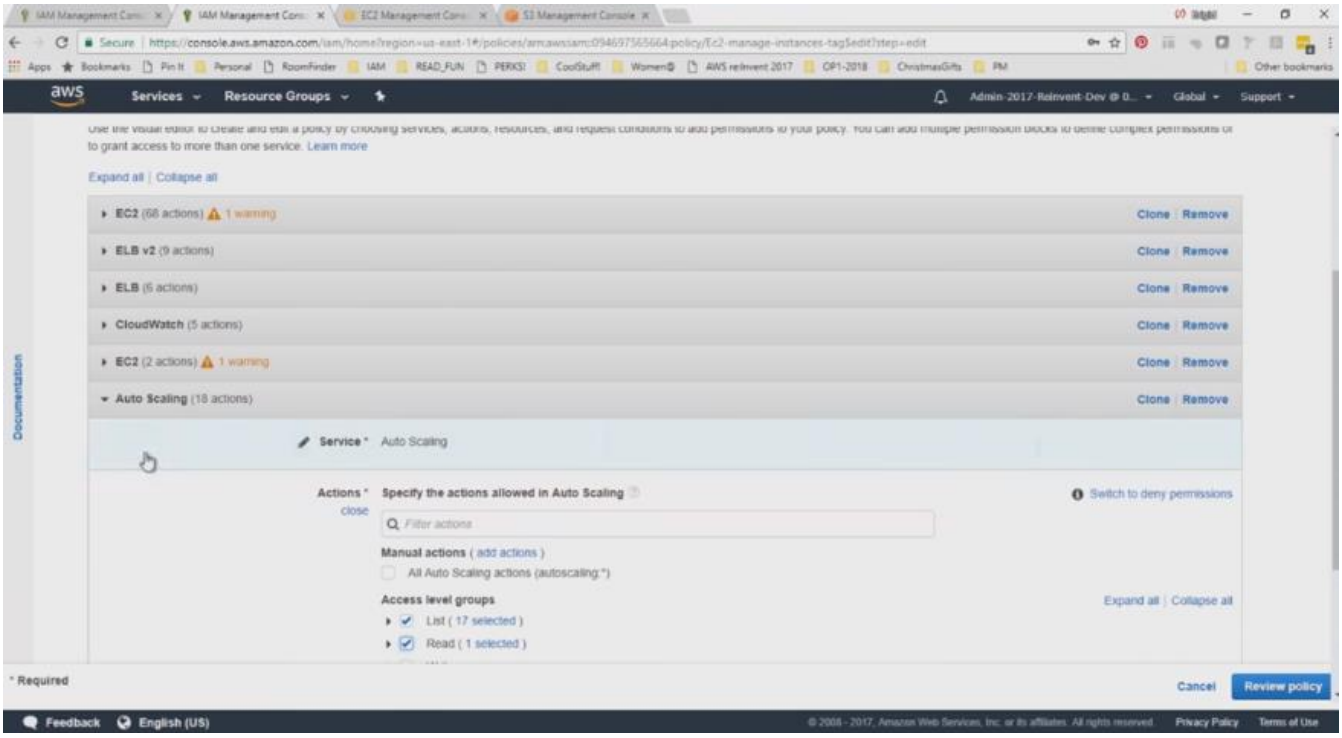


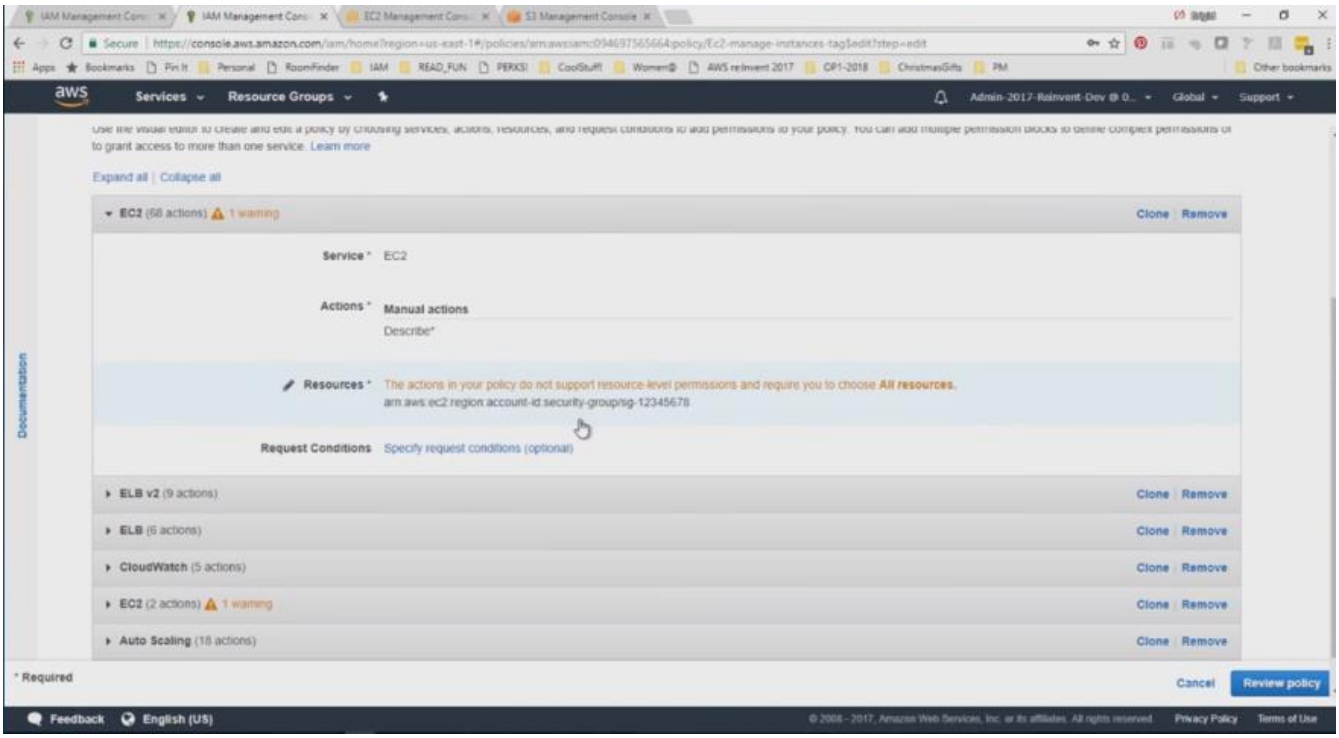


There is an error, we can edit to fix each warning

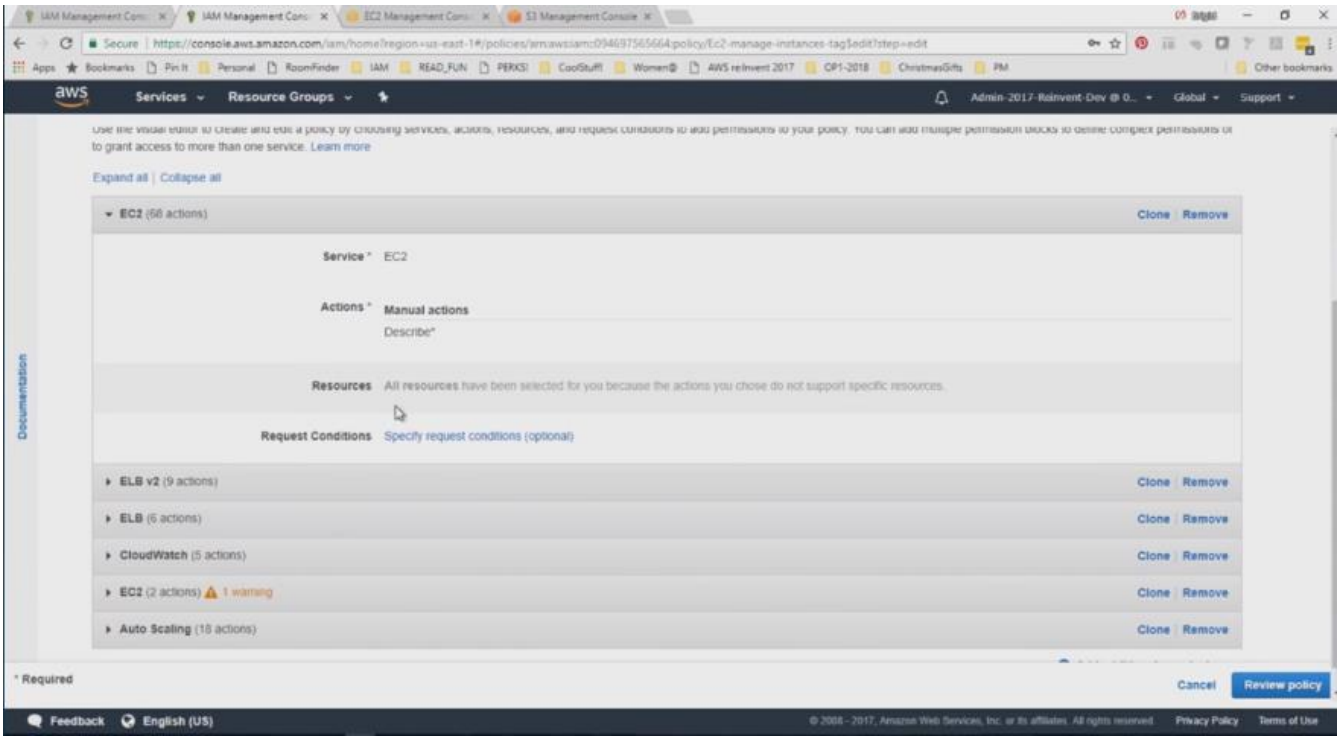


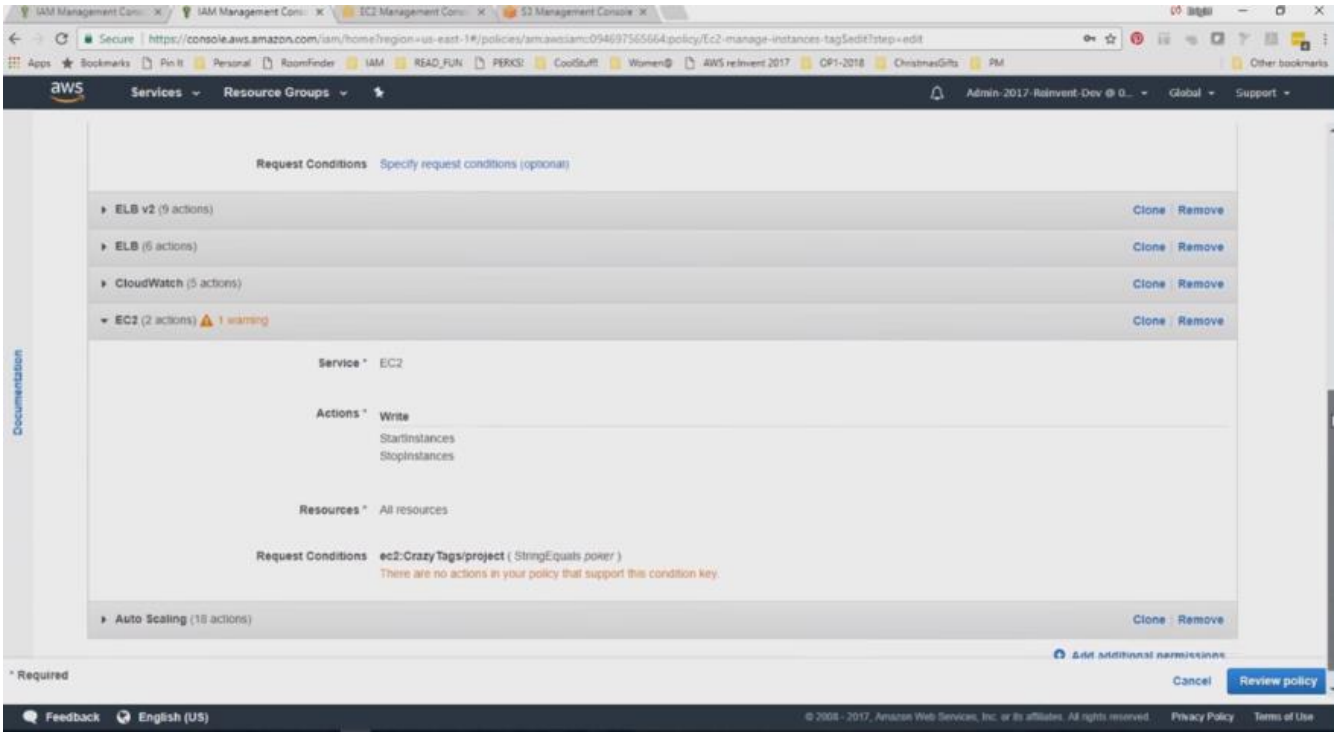
We can remove the last faulty policy and add a new correct one in its place as below



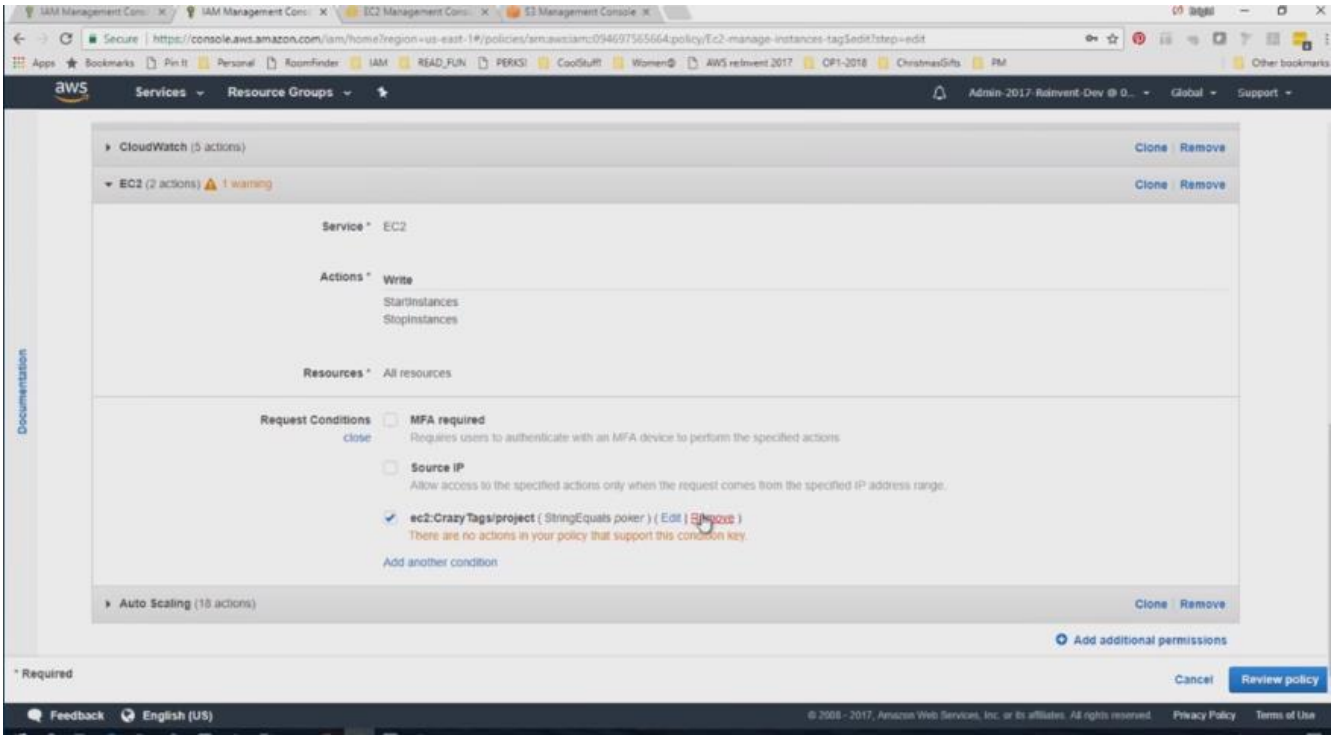


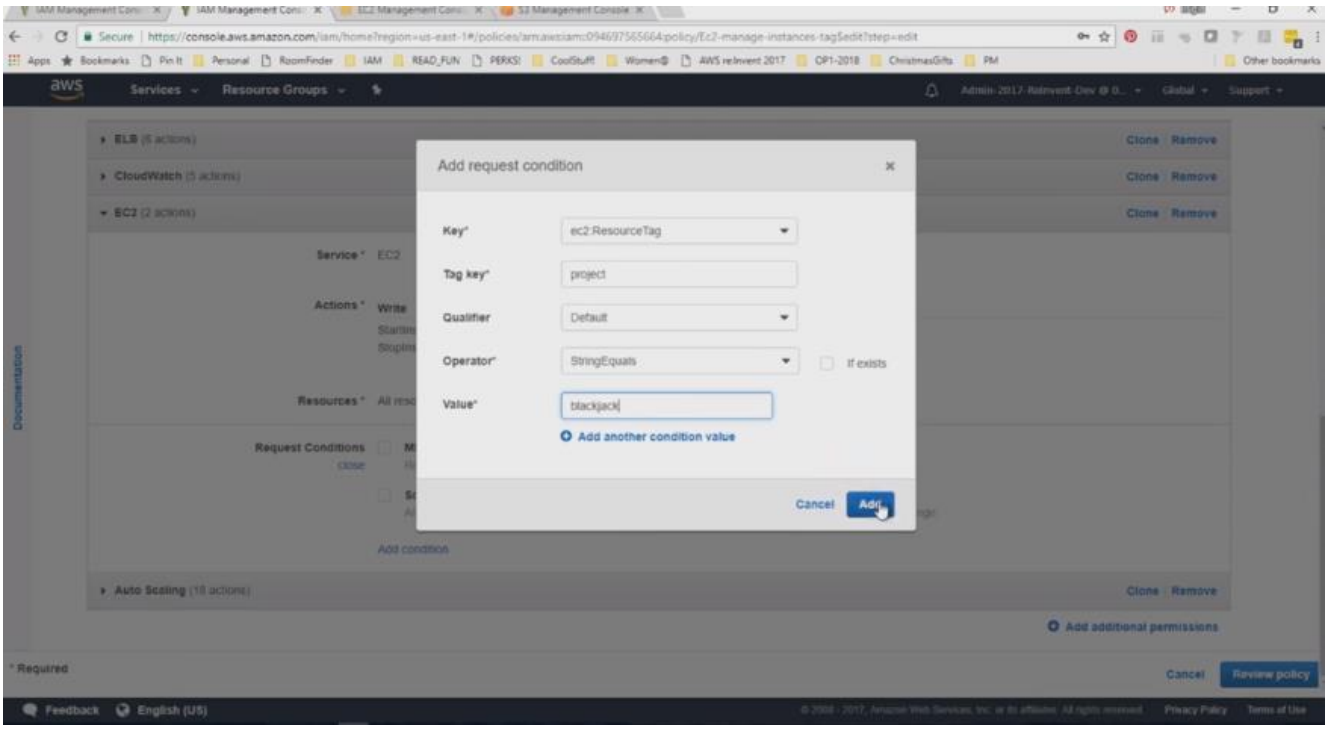
We need to fix this also

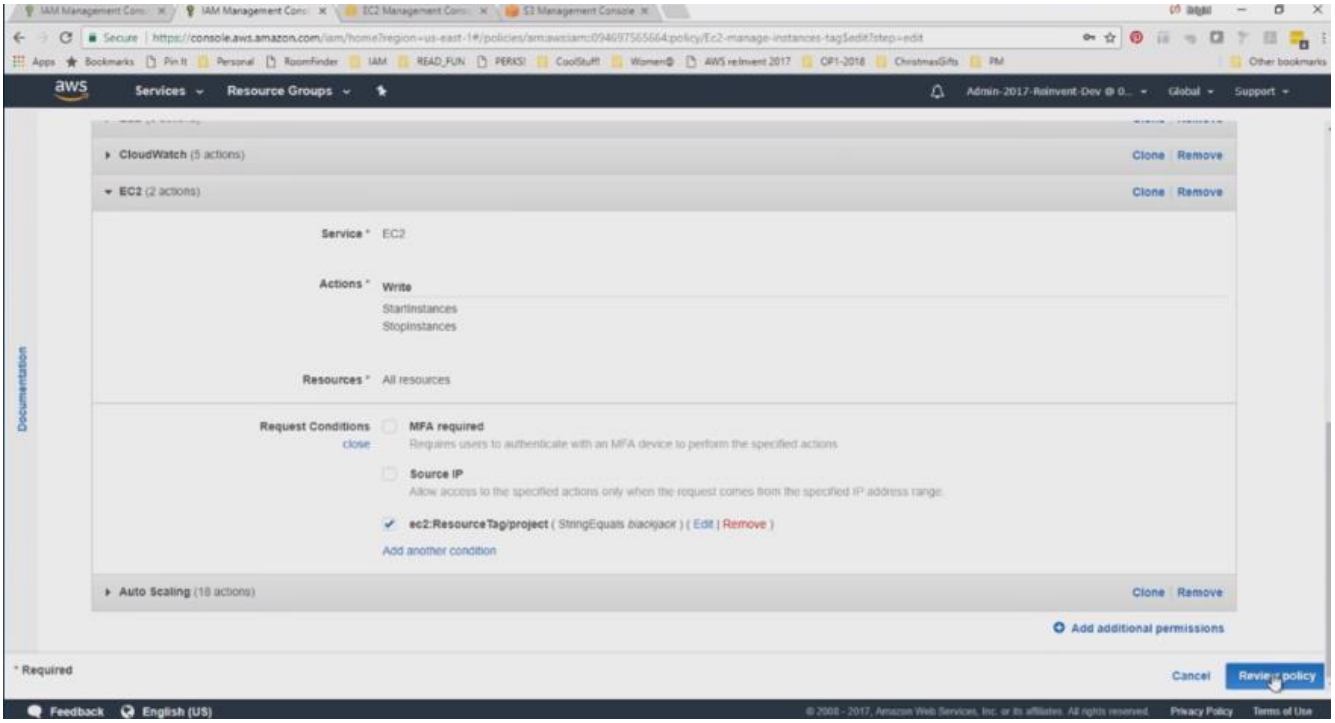


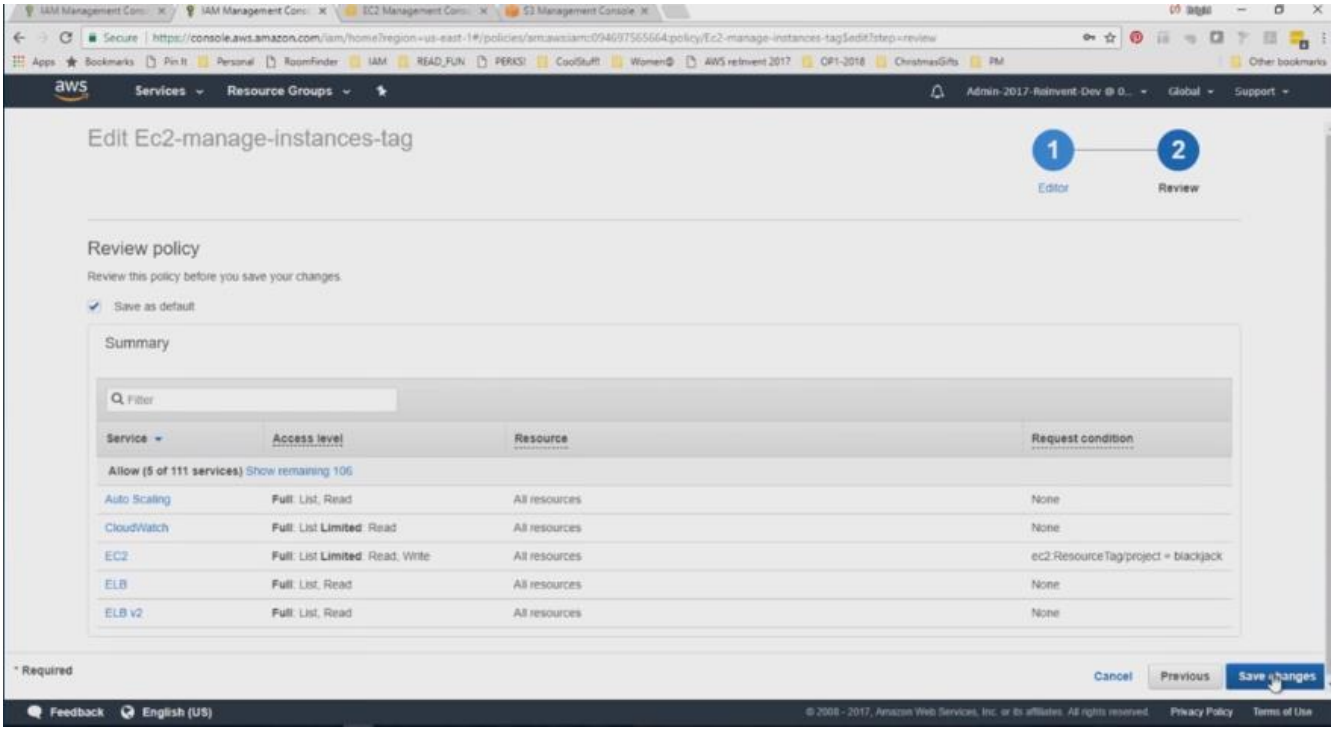


We need to fix this too

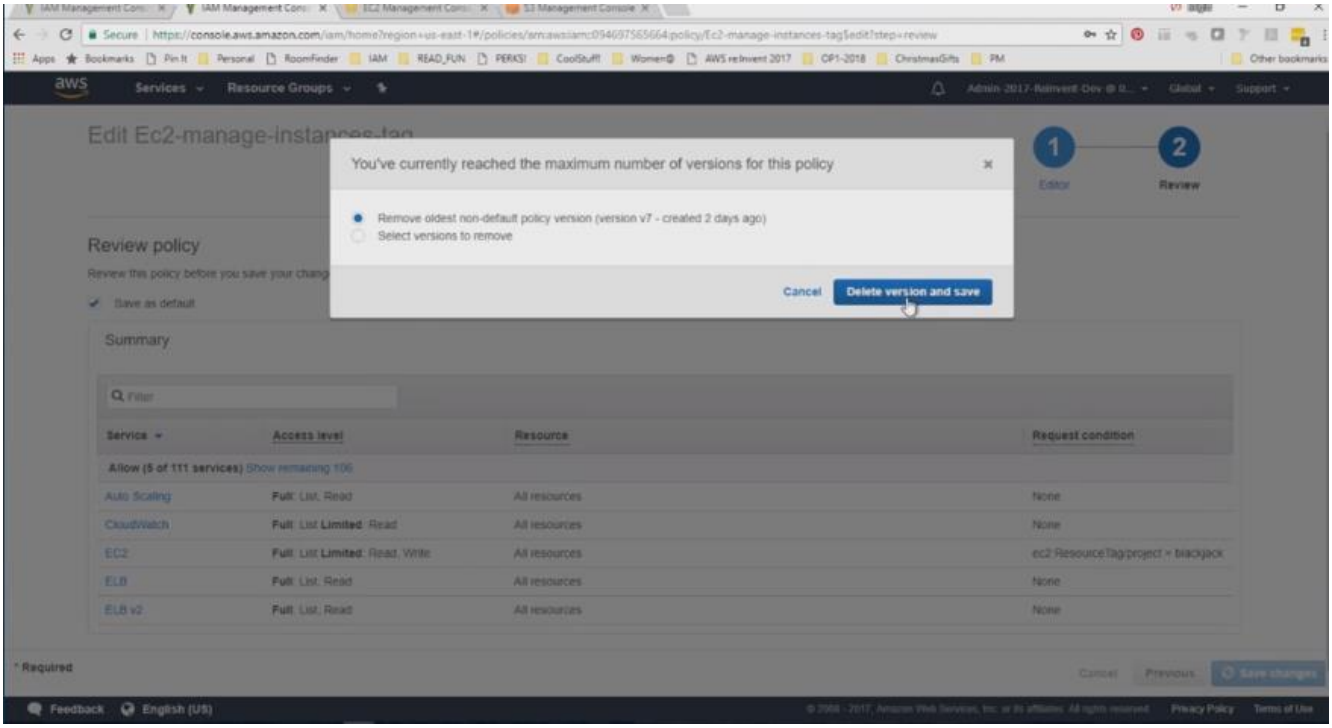

Services
Resource Groups
CloudWatch (5 actions)
Clone | Remove
EC2 (2 actions) 1 warning
Clone | Remove
Service * EC2
Actions * Write
StartInstances
StopInstances
Resources * All resources
Request Conditions
close
MFA required
Requires users to authenticate with an MFA device to perform the specified actions
Source IP
Allow access to the specified actions only when the request comes from the specified IP address range.
ec2:CrazyTags/project (StringEquals poker) (Edit | Remove)
There are no actions in your policy that support this condition key.
Add another condition
Auto Scaling (18 actions)
Clone | Remove
Add additional permissions
* Required
Cancel
Review policy
Feedback
English (US)
Privacy Policy
Terms of Use
Documentation

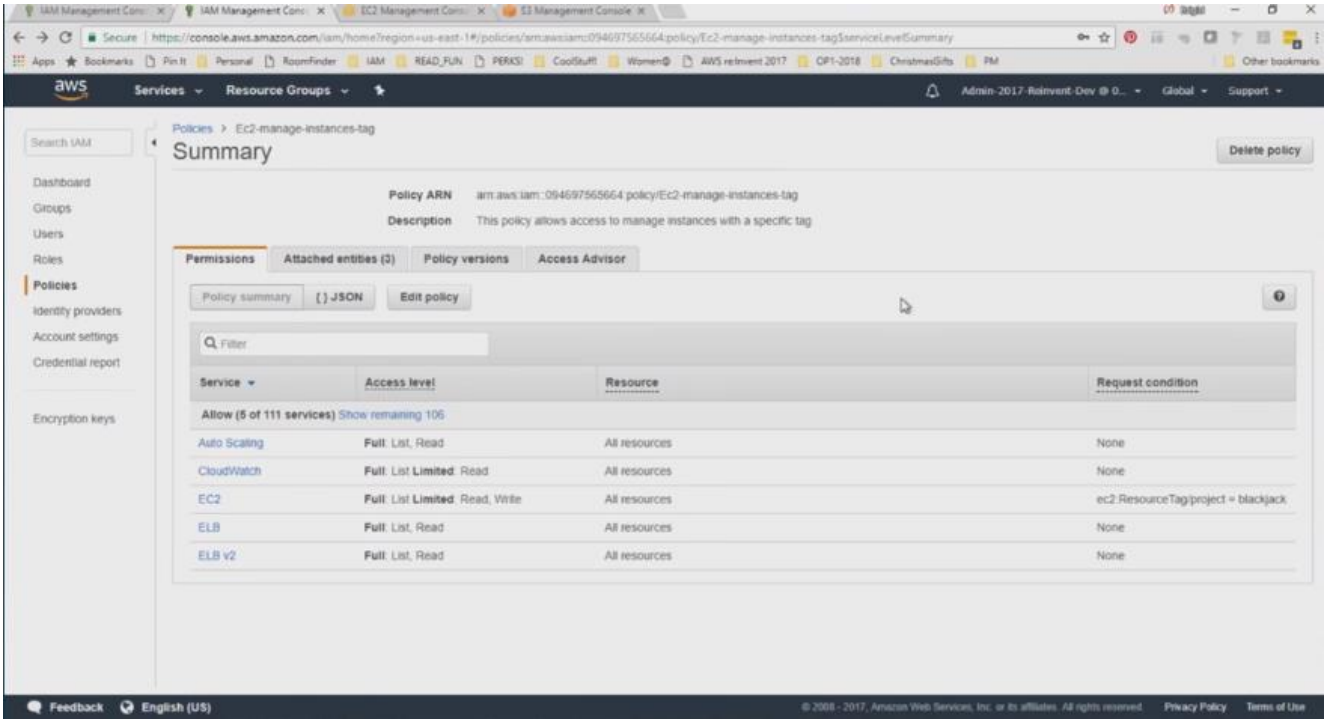

Add request condition
Key* ec2:ResourceTag
Tag key* project
Qualifier Default
Operator* StringEquals
If exists
Value* blackjack
Add another condition value
Cancel
Add
Services
Resource Groups
Review policy
Feedback
English (US)
Documentation

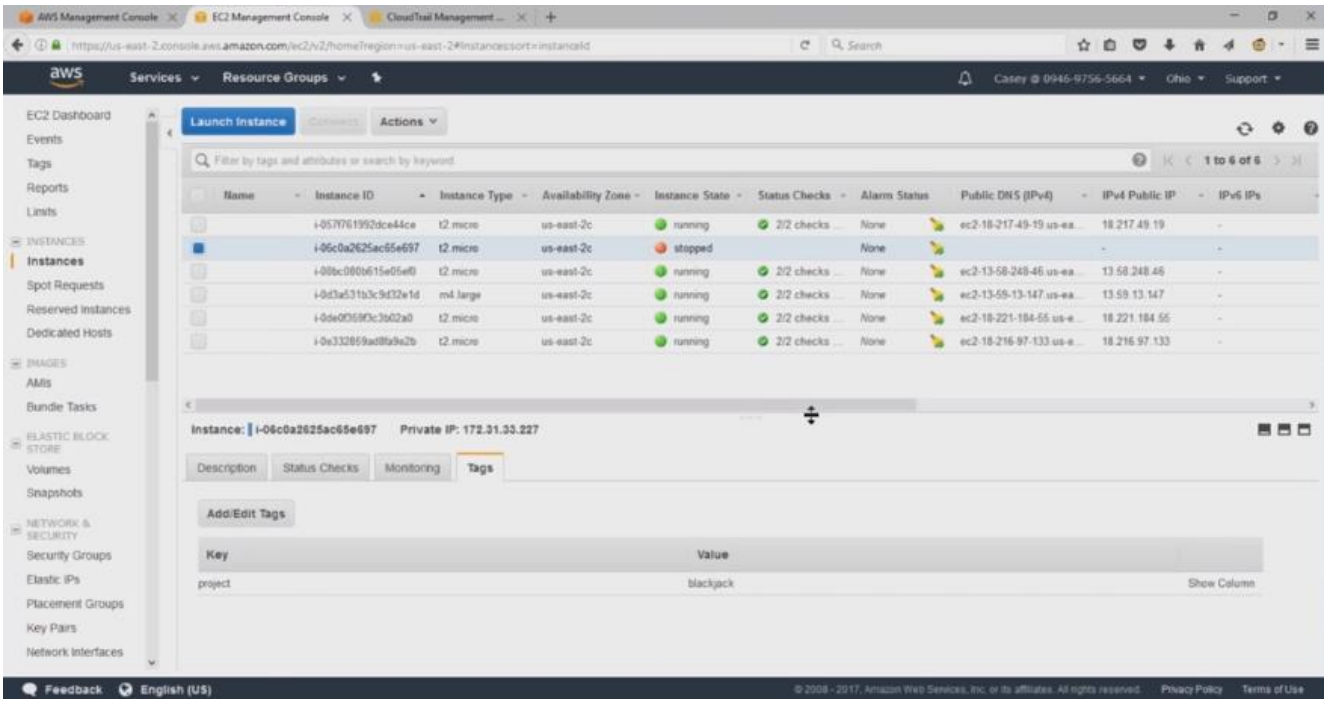




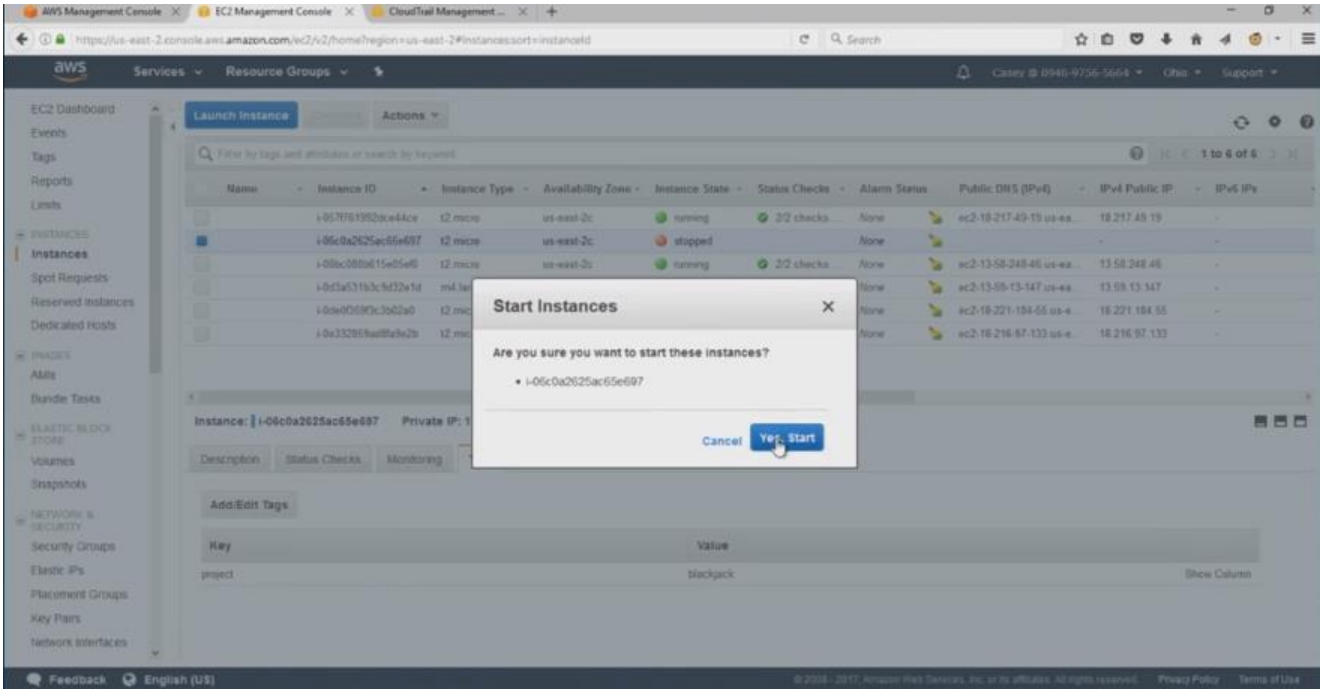
No more errors

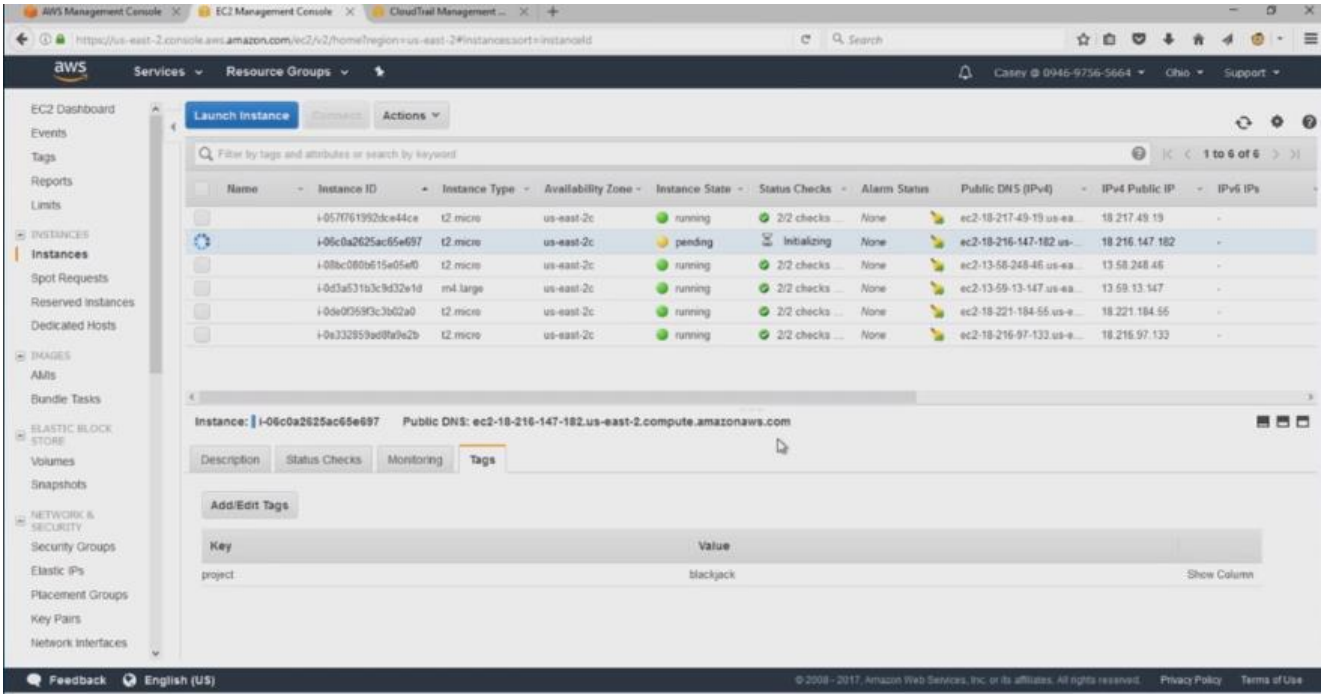

You've currently reached the maximum number of versions for this policy
Remove oldest non-default policy version (version v7 - created 2 days ago)
Select versions to remove
Cancel
Delete version and save
Review policy
Summary
Service
Access level
Resource
Request condition
Allow (5 of 111 services) Show remaining 106
Auto Scaling
CloudWatch
EC2
ELB
ELB v2
* Required


Policies > Ec2-manage-instances-tag
Summary
Delete policy
Policy ARN
arn:aws:iam::094697565664:policy/Ec2-manage-instances-tag
Description
This policy allows access to manage instances with a specific tag
Permissions
Attached entities (3)
Policy versions
Access Advisor
Policy summary
{ } JSON
Edit policy
Service
Access level
Resource
Request condition
Allow (5 of 111 services) Show remaining 106
Auto Scaling
CloudWatch
EC2
ELB
ELB v2
Dashboard
Groups
Users
Roles
Policies
Identity providers
Account settings
Credential report
Encryption keys
Feedback
English (US)

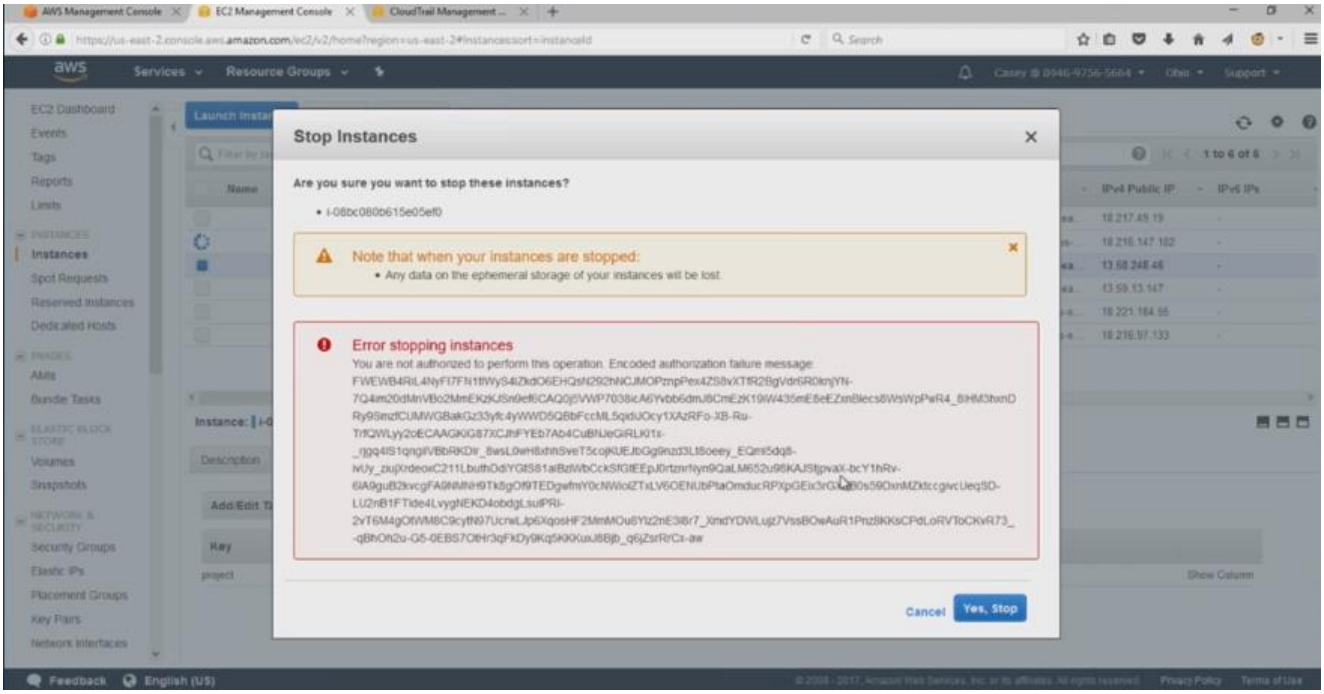


Casey can now see an EC2 instance tagged with the key/value of project/blackjack. Casey can then start an instance





Casey can indeed start this kind of specific instance



Trying to start a different instance results in the above error screen due to no permissions for that action

Demo: Correct an existing policy

We found four issues:

1. Typo in our services
2. Mismatch of supported resource and action
3. Condition key not supported
4. Missing actions

Recap

- Review of IAM policy language
- Review of how policies work together to control access
- Policy tools – Create, test, and correct policies
- Policy Foo – Ninja moves with EC2

- Additional resources

Tags are very useful as metadata on your EC2 Resources that are key/values pairs for ***discovering your resources***, ***organizing your resources***, ***cost allocation for auditing***, and ***access control to your resources***. As an admin, you can control how users can add a tag and also which tags users can create.

Permissions overview

Allow adding tags only during instance launch

Require specific tags with values at instance launch

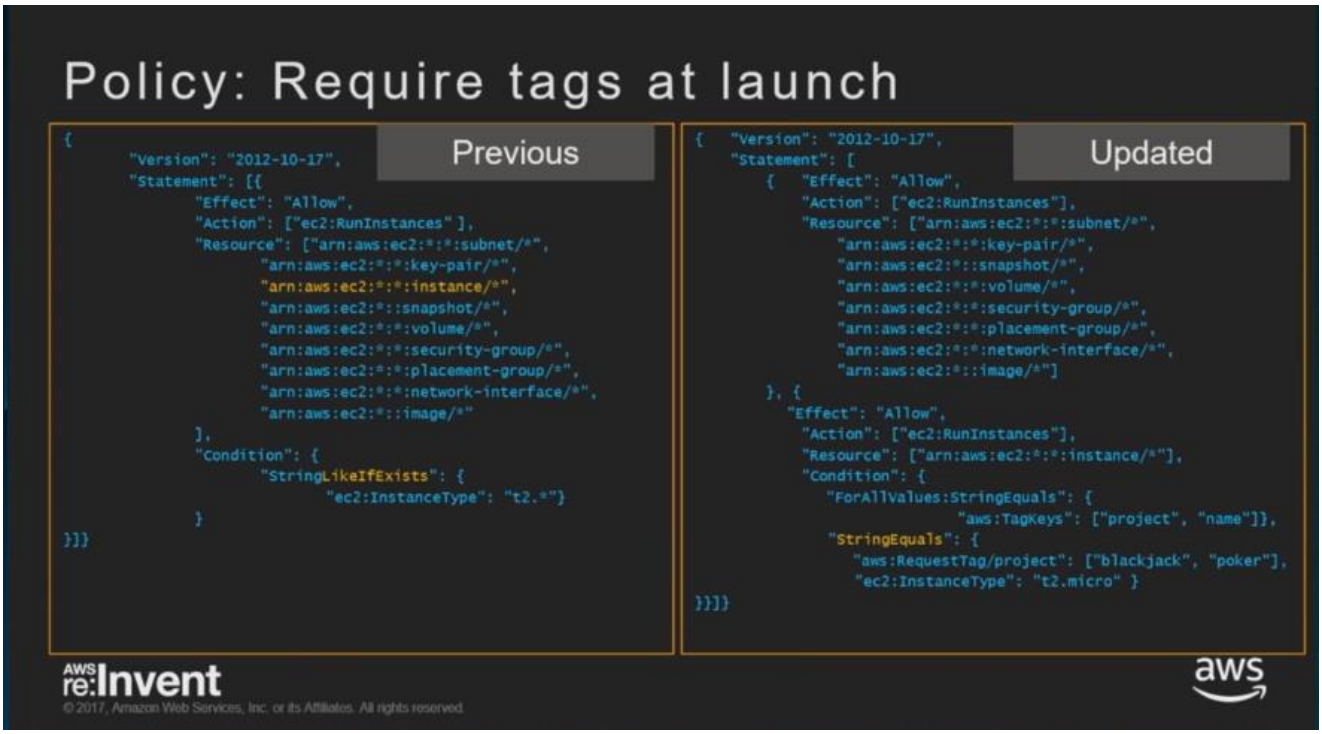
Bonus: Allow modify and create specific tags and resources

Policy: Create tag with RunInstances

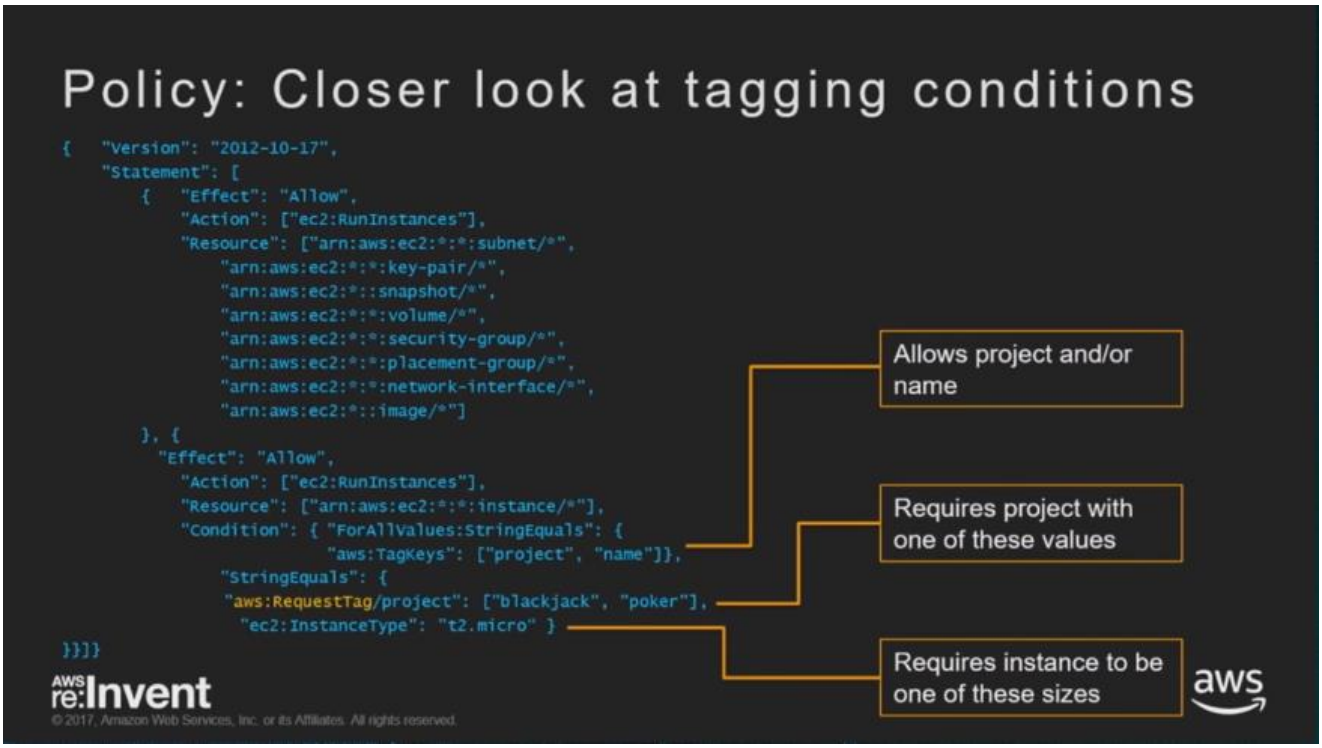
```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": "ec2:CreateTags",
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "ec2:CreateAction": "RunInstances"
            }
        }
    }]
}
```

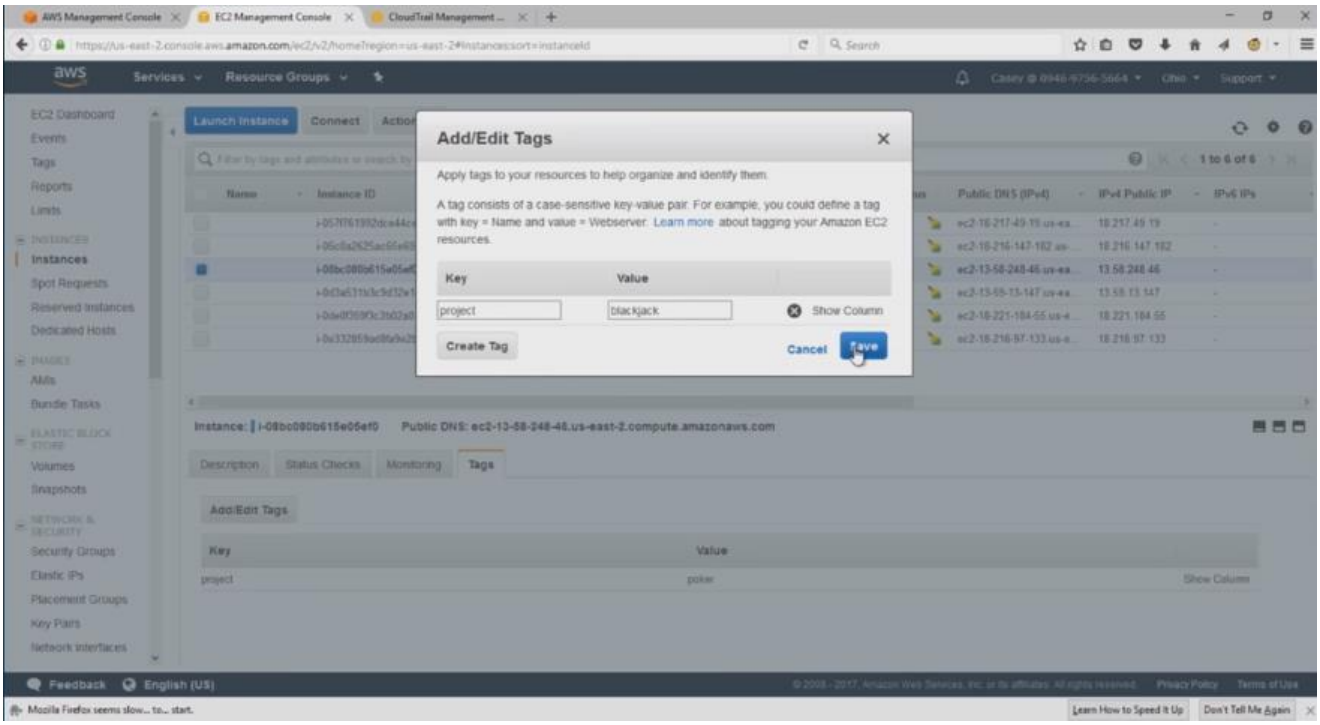
Allows creation of tags

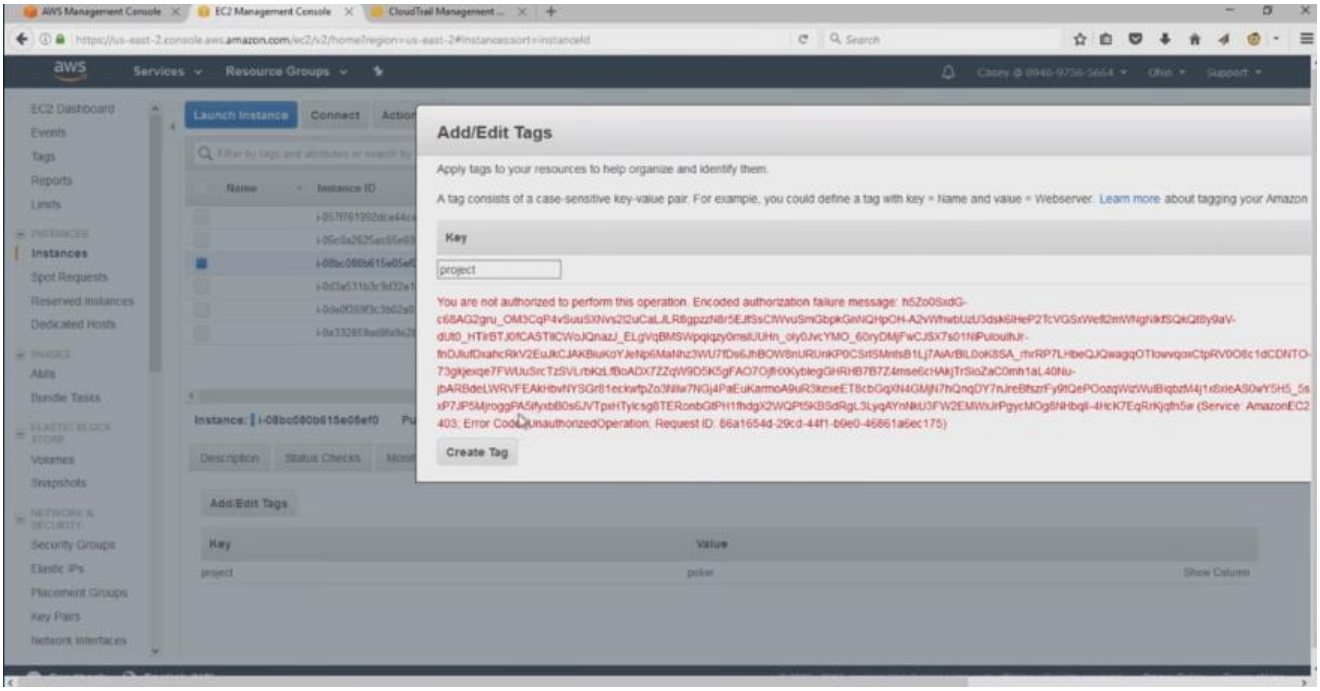
But only during RunInstances calls



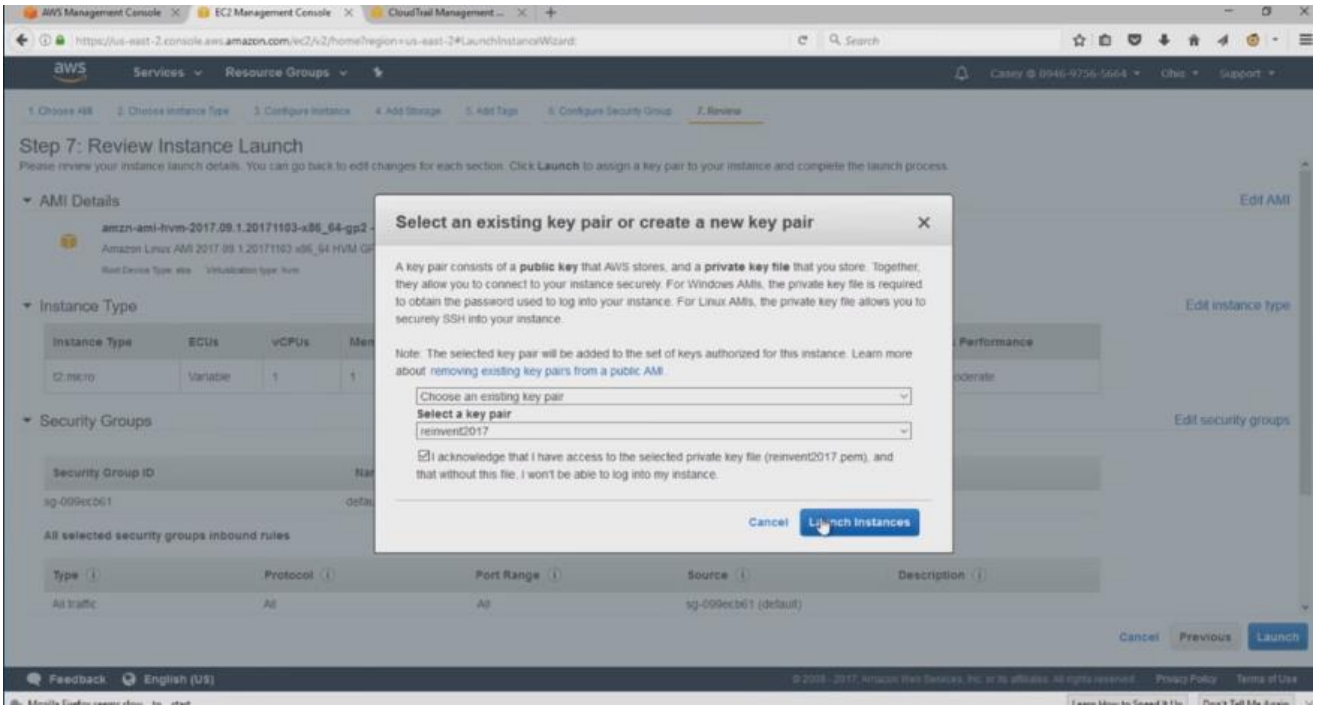
We have now broken up the Previous Condition into 2 separate conditions statements in the Updated part.

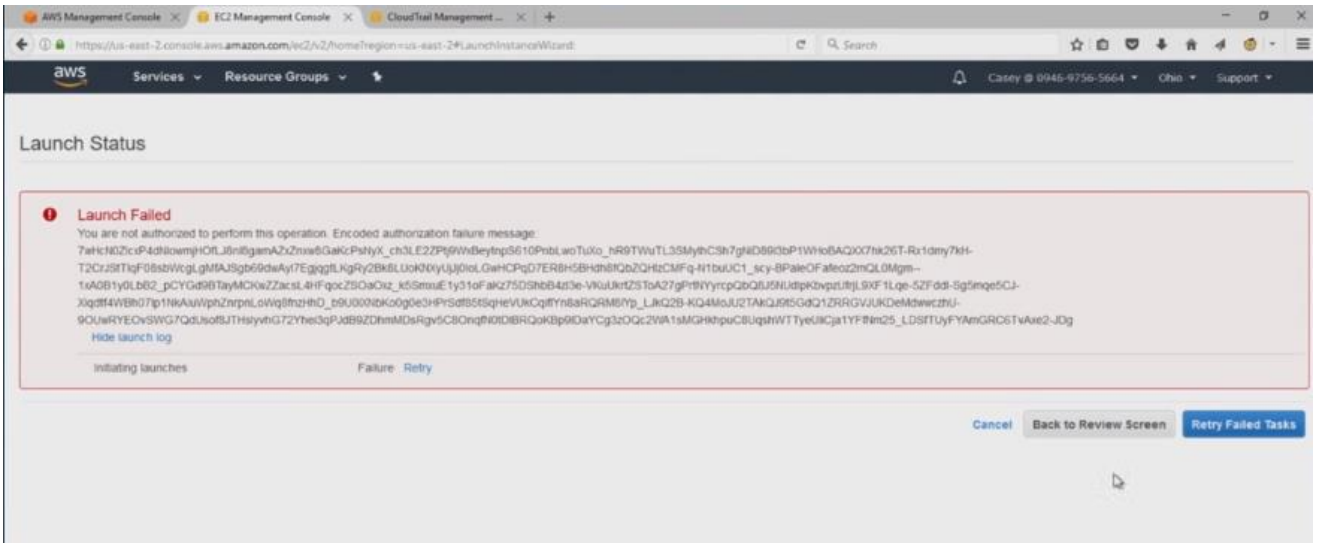




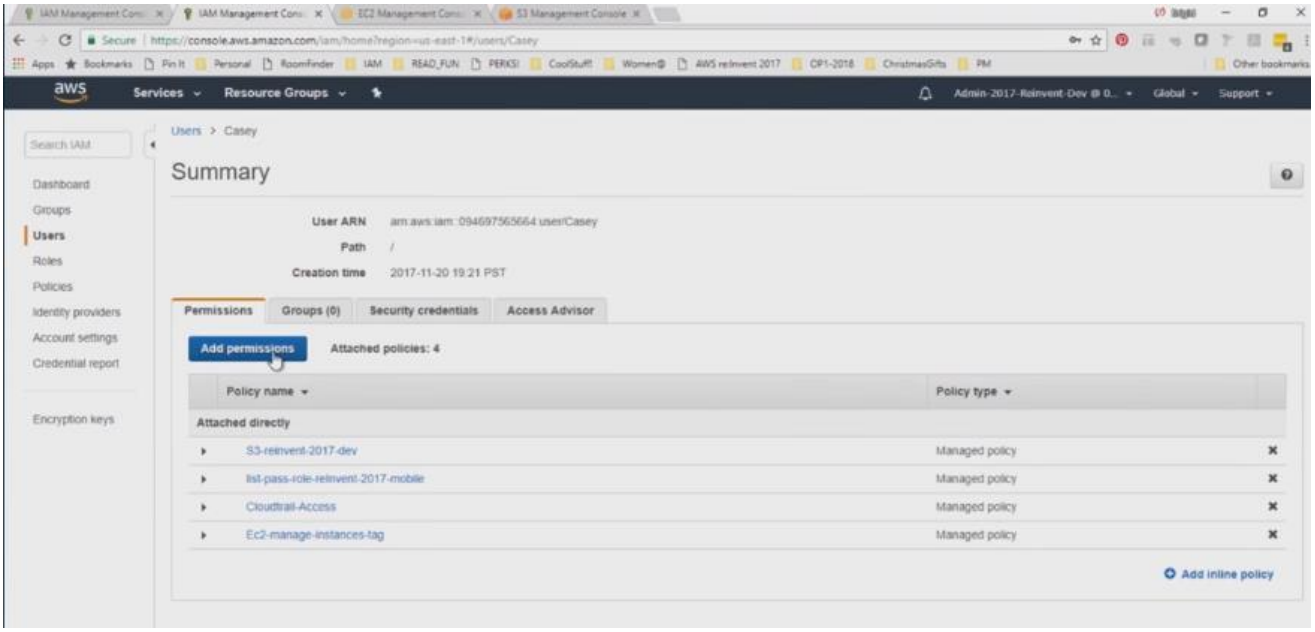


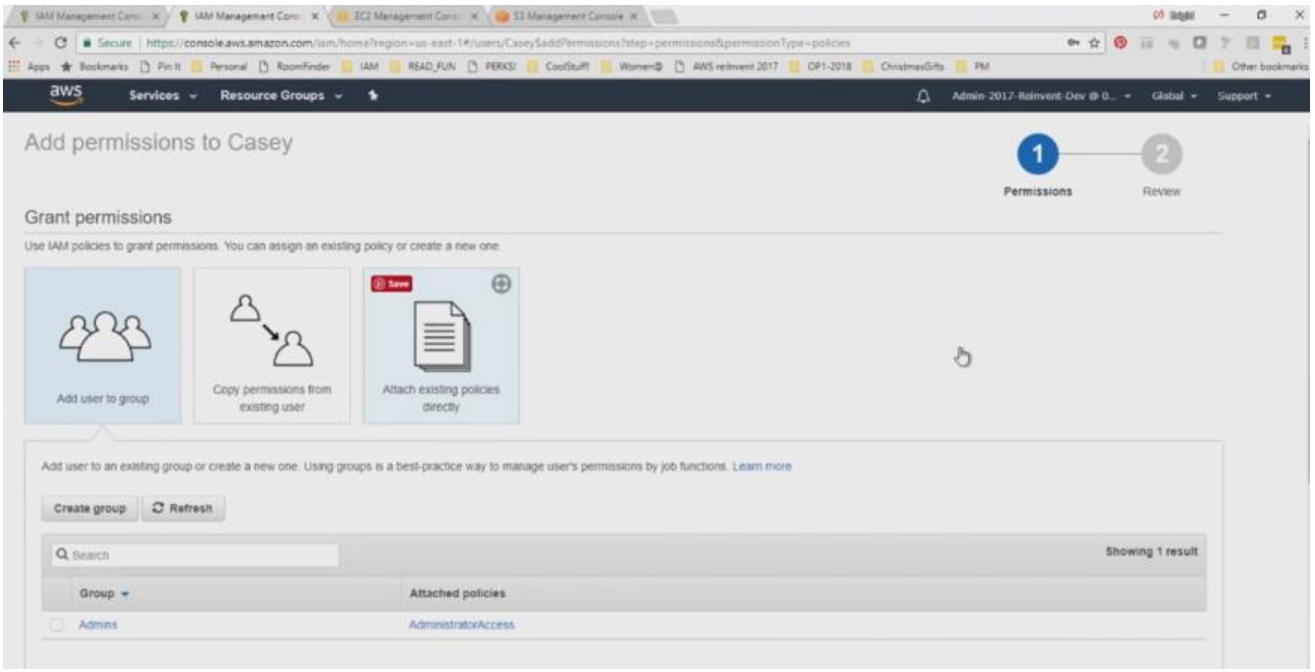
Casey can’t create a tag

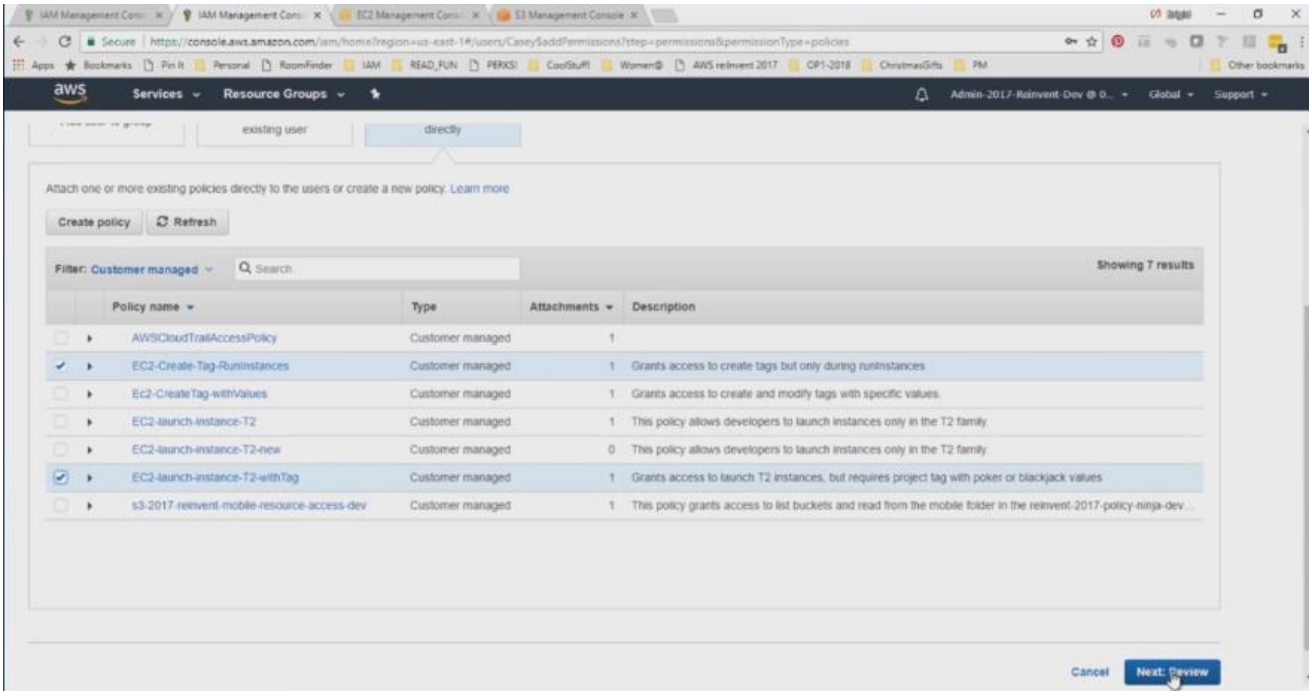


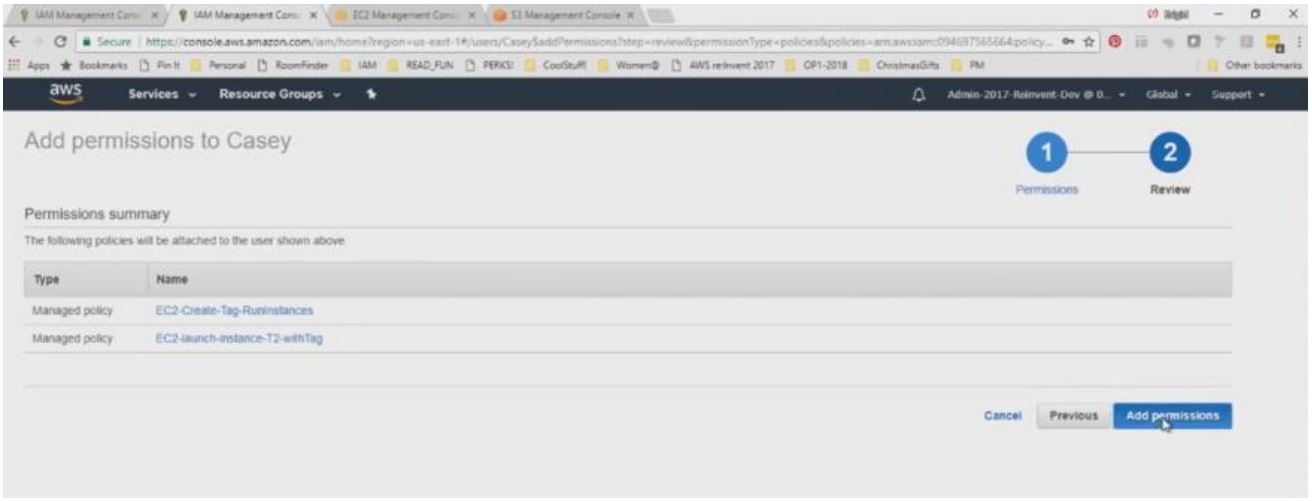


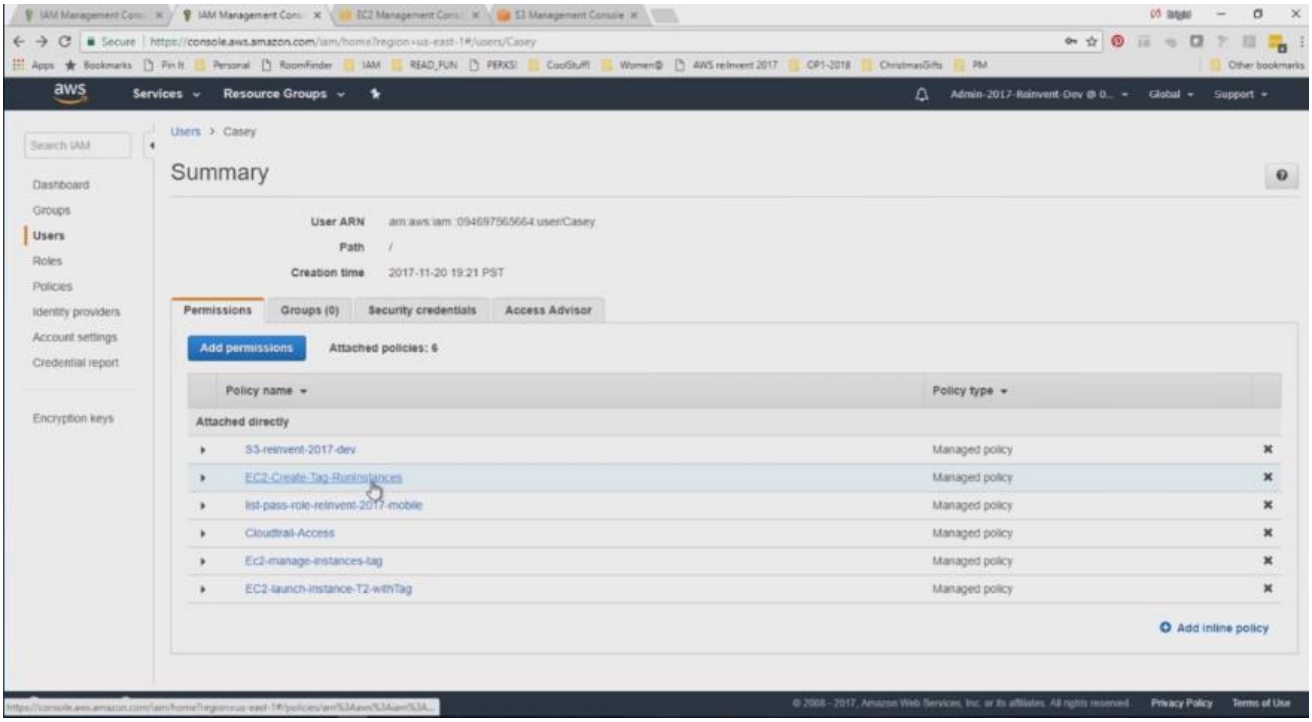
Casey also does not have permissions to launch instances at the moment. Let us go and add those permissions below

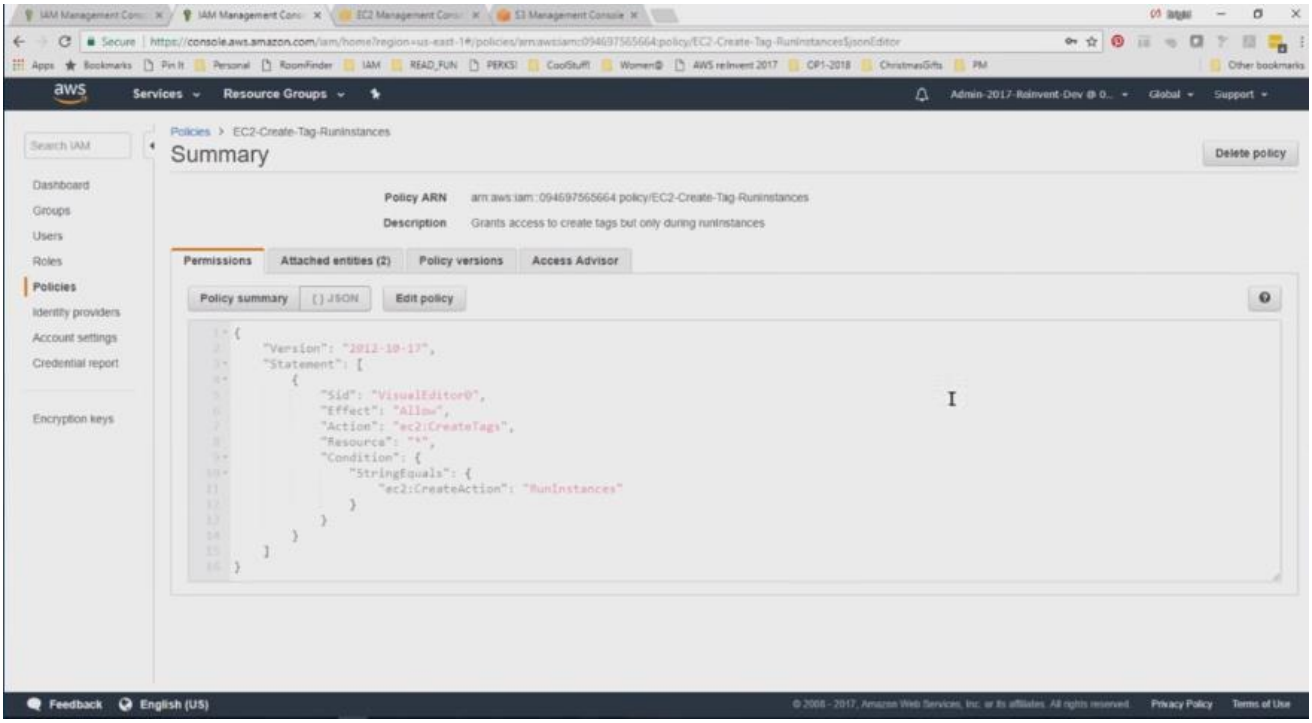

Services
Resource Groups
Search IAM
Dashboard
Groups
Users
Roles
Policies
Identity providers
Account settings
Credential report
Encryption keys
Users > Casey
Summary
User ARN arn:aws:iam::094697565664:user/Casey
Path /
Creation time 2017-11-20 19:21 PST
Permissions
Groups (0)
Security credentials
Access Advisor
Add permissions
Attached policies: 4
Policy name
Policy type
Attached directly
S3-reinvent-2017-dev
list-pass-role-reinvent-2017-mobile
Cloudtrail-Access
Ec2-manage-instances-tag
Managed policy
Add inline policy

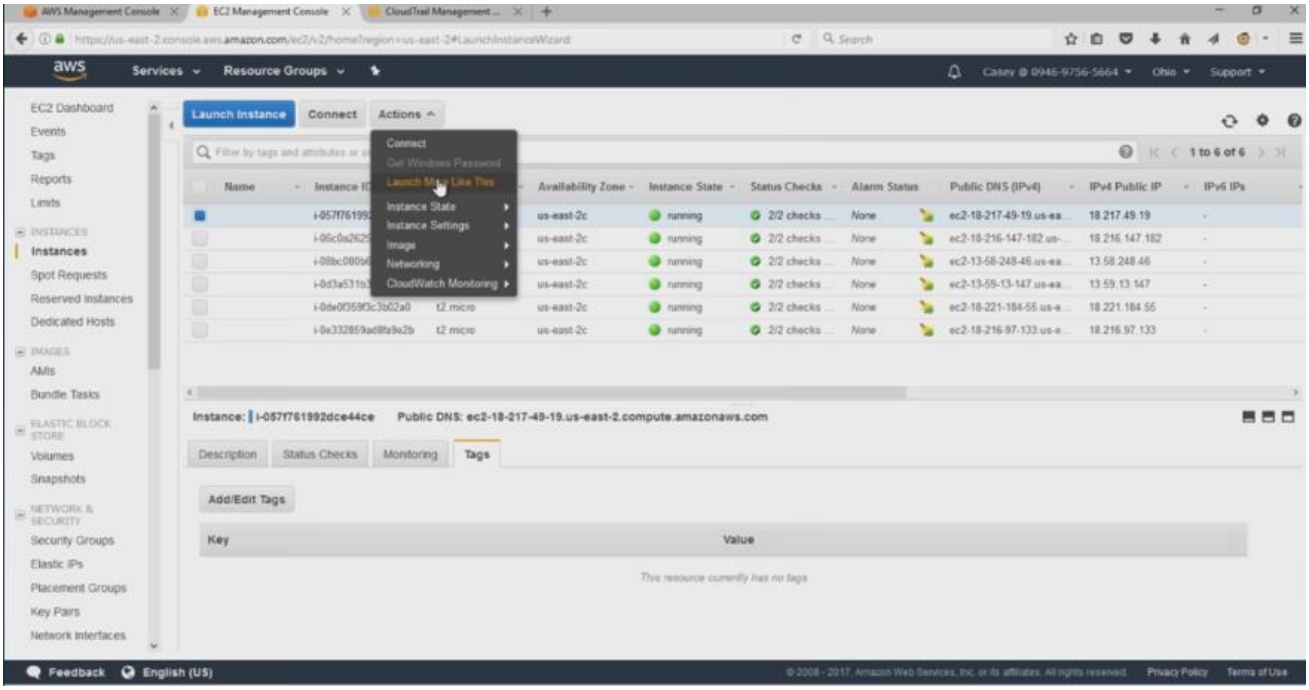

Services
Resource Groups
Add permissions to Casey
1
Permissions
2
Review
Grant permissions
Use IAM policies to grant permissions. You can assign an existing policy or create a new one.
Add user to group
Copy permissions from existing user
Attach existing policies directly
Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. Learn more
Create group
Refresh
Search
Showing 1 result
Group
Attached policies
Admins
AdministratorAccess

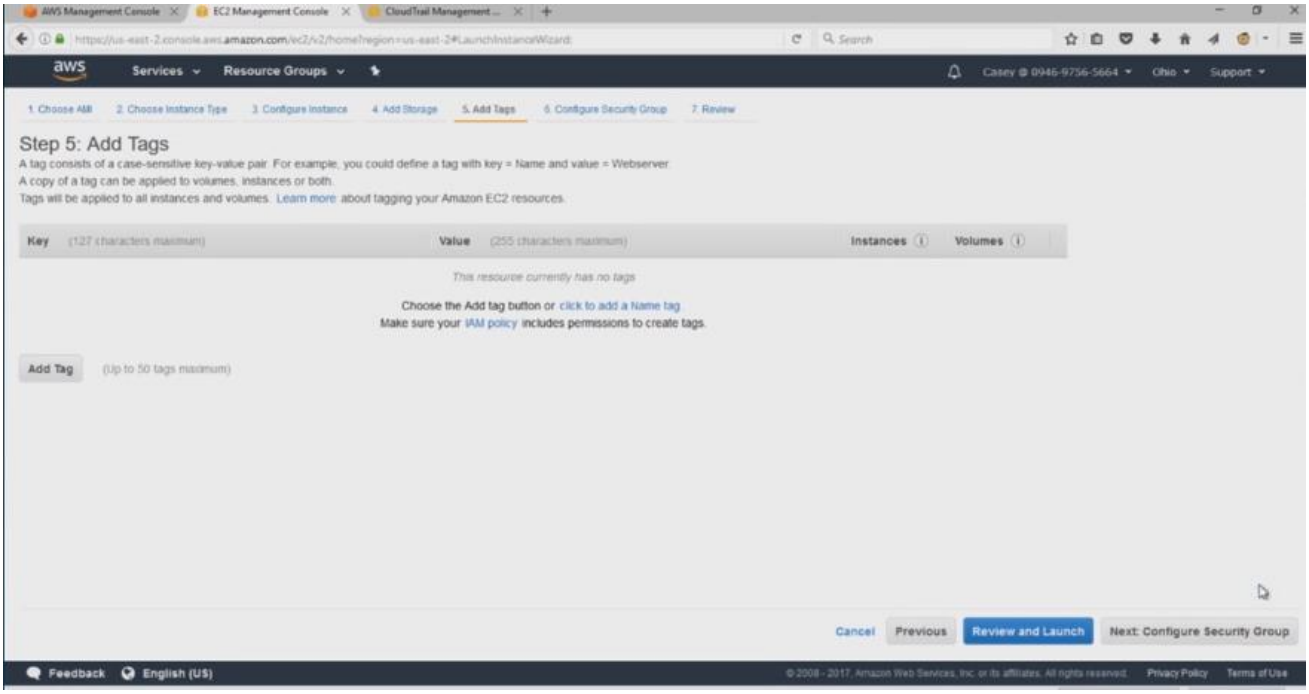

Services
Resource Groups
existing user
directly
Attach one or more existing policies directly to the users or create a new policy. Learn more
Create policy
Refresh
Filter: Customer managed
Search
Showing 7 results
Policy name
Type
Attachments
Description
AWSCloudTrailAccessPolicy
Customer managed
EC2-Create-Tag-RunInstances
Grants access to create tags but only during runInstances
Ec2-CreateTag-withValues
Grants access to create and modify tags with specific values.
EC2-launch-instance-T2
This policy allows developers to launch instances only in the T2 family.
EC2-launch-instance-T2-new
EC2-launch-instance-T2-withTag
Grants access to launch T2 instances, but requires project tag with poker or blackjack values
s3-2017-reinvent-mobile-resource-access-dev
Cancel
Next: Review

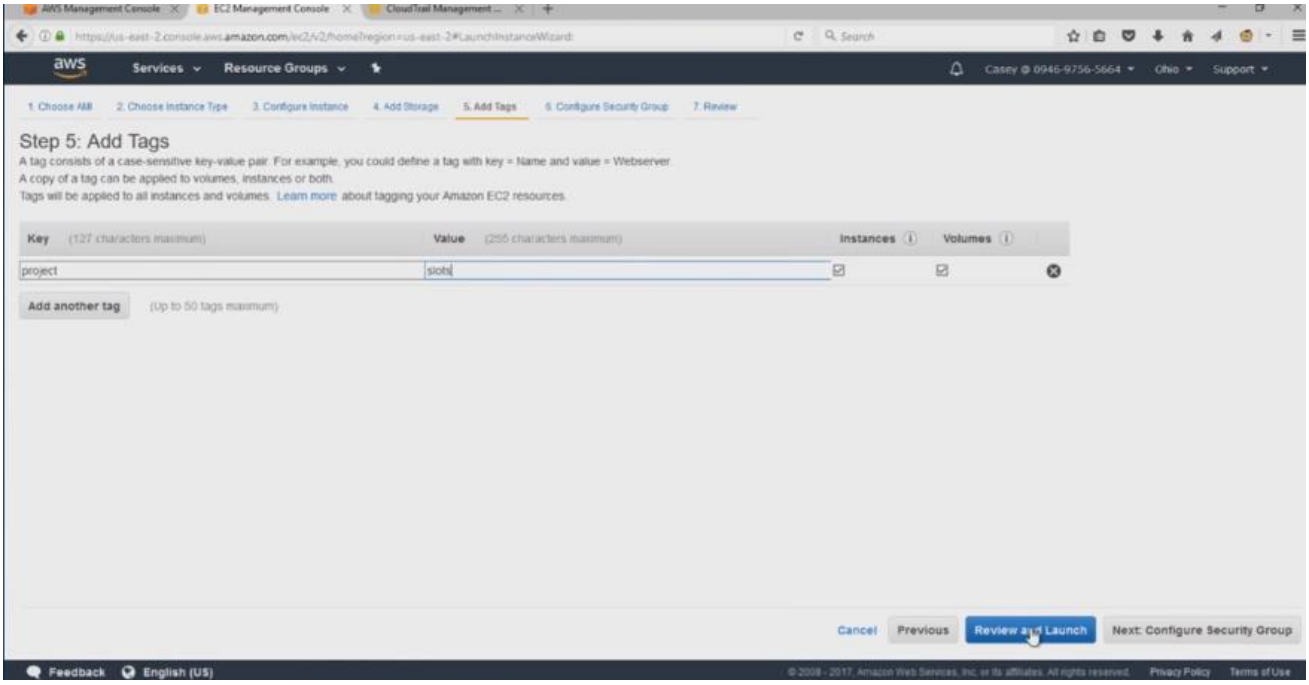

Services
Resource Groups
Add permissions to Casey
1
Permissions
2
Review
Permissions summary
The following policies will be attached to the user shown above.
Type
Name
Managed policy
EC2-Create-Tag-RunInstances
EC2-launch-instance-T2-withTag
Cancel
Previous
Add permissions

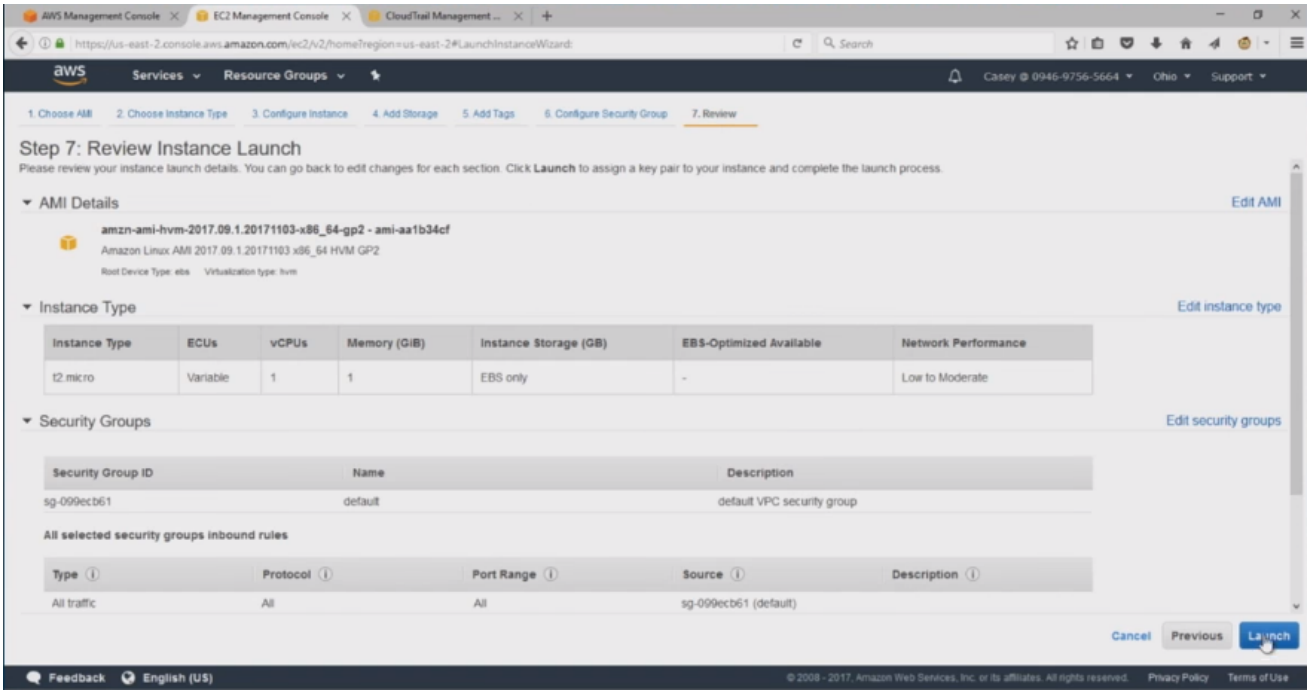

Services
Resource Groups
Users > Casey
Summary
User ARN arn:aws:iam::094697565664:user/Casey
Path /
Creation time 2017-11-20 19:21 PST
Permissions
Groups (0)
Security credentials
Access Advisor
Add permissions
Attached policies: 6
Policy name
Policy type
Attached directly
S3-reinvent-2017-dev
EC2-Create-Tag-RunInstances
list-pass-role-reinvent-2017-mobile
Cloudtrail-Access
Ec2-manage-instances-tag
EC2-launch-instance-T2-withTag
Managed policy
Add inline policy

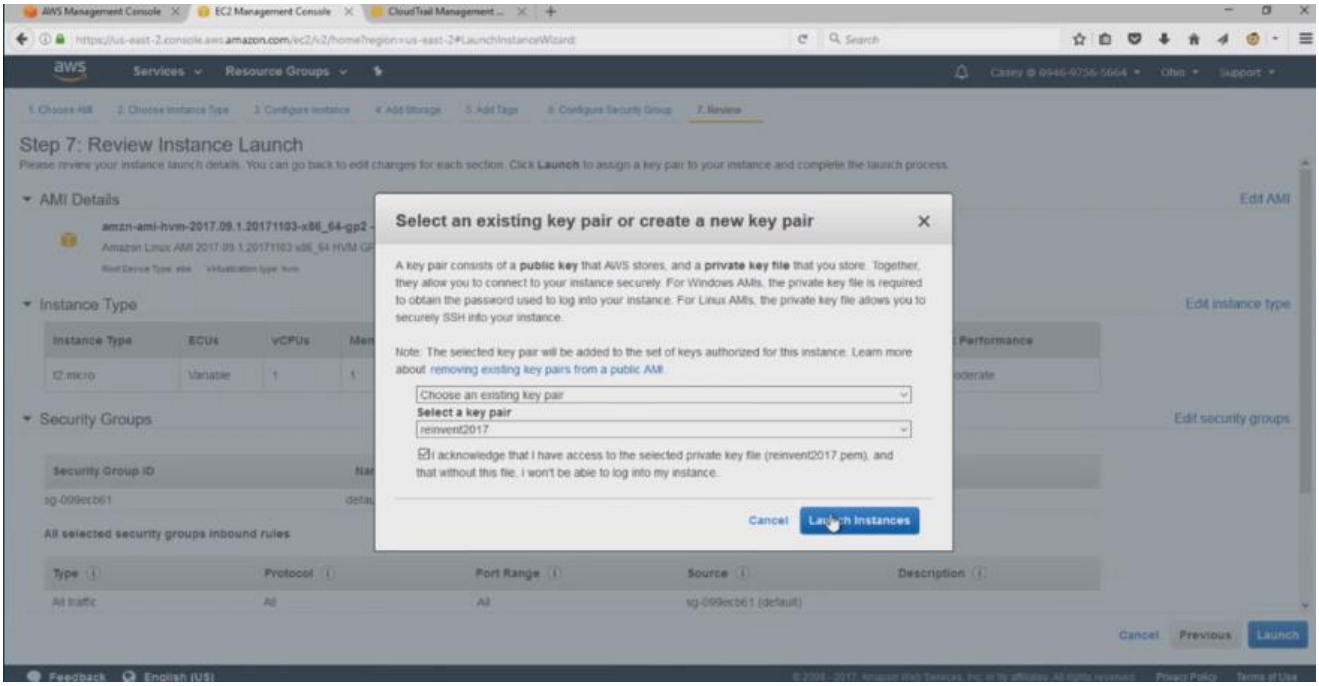

Services
Resource Groups
Policies > EC2-Create-Tag-RunInstances
Summary
Delete policy
Policy ARN arn:aws:iam::094697565664:policy/EC2-Create-Tag-RunInstances
Description Grants access to create tags but only during runInstances
Permissions
Attached entities (2)
Policy versions
Access Advisor
Policy summary
{} JSON
Edit policy
{
"Version": "2012-10-17",
"Statement": [
{
"Sid": "VisualEditor0",
"Effect": "Allow",
"Action": "ec2:CreateTags",
"Resource": "*",
"Condition": {
"StringEquals": {
"ec2:CreateAction": "RunInstances"
}
}
}
]
}
Feedback
English (US)

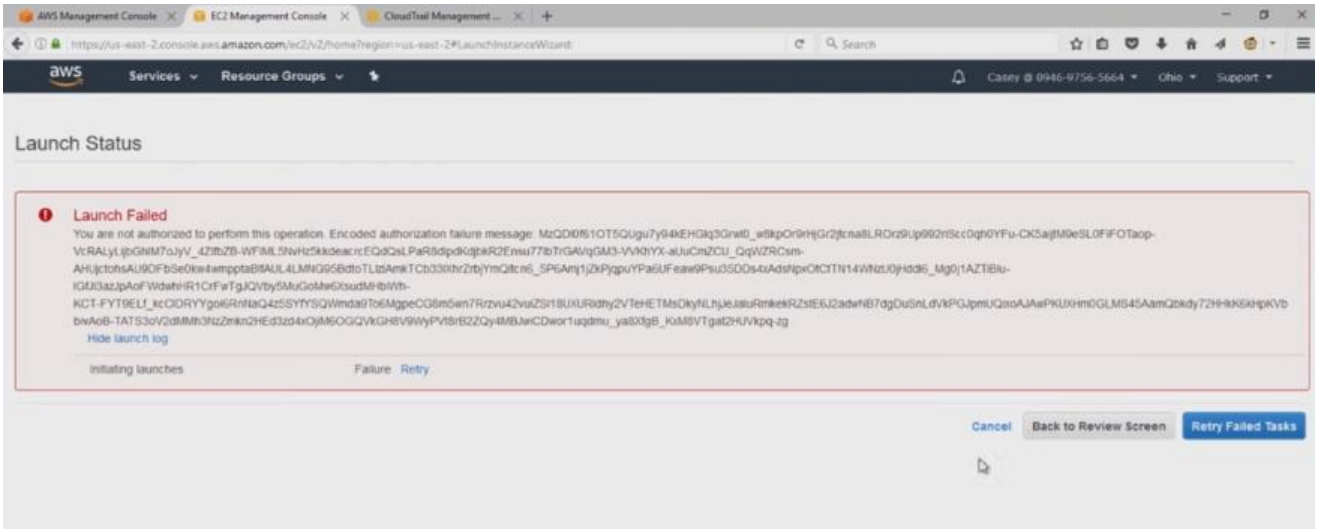

Launch Instance
Connect
Actions
Connect
Get Windows Password
Launch More Like This
Instance State
Instance Settings
Image
Networking
CloudWatch Monitoring
Instance: i-057f761992dce44ce
Public DNS: ec2-18-217-49-19.us-east-2.compute.amazonaws.com
Description
Status Checks
Monitoring
Tags
Add/Edit Tags
Key
Value
This resource currently has no tags


Step 5: Add Tags
A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.
A copy of a tag can be applied to volumes, instances or both.
Tags will be applied to all instances and volumes. Learn more about tagging your Amazon EC2 resources.
Key
Value
Instances
Volumes
This resource currently has no tags
Choose the Add tag button or click to add a Name tag.
Make sure your IAM policy includes permissions to create tags.
Add Tag
Cancel
Previous
Review and Launch
Next: Configure Security Group

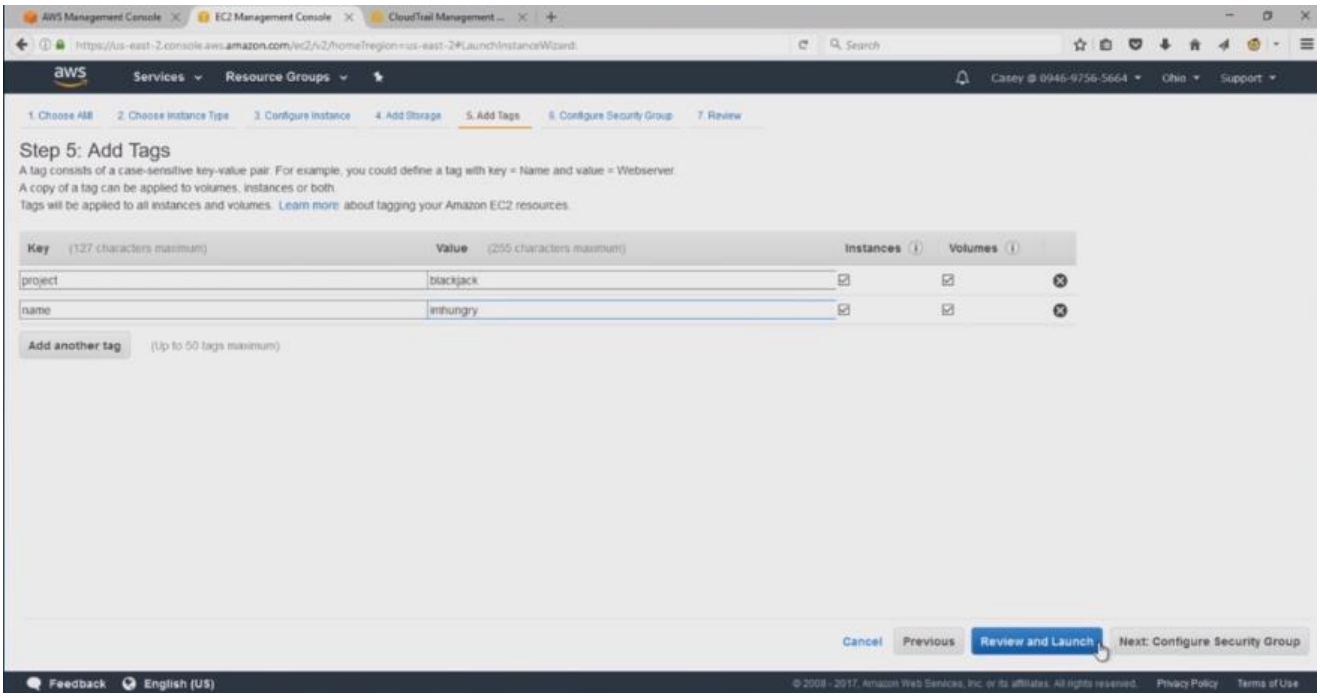

Services
Resource Groups
1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
Step 5: Add Tags
A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.
A copy of a tag can be applied to volumes, instances or both.
Tags will be applied to all instances and volumes. Learn more about tagging your Amazon EC2 resources.
Key (127 characters maximum)
Value (256 characters maximum)
Instances
Volumes
project
slots
Add another tag
(Up to 50 tags maximum)
Cancel
Previous
Review and Launch
Next: Configure Security Group
Feedback
English (US)

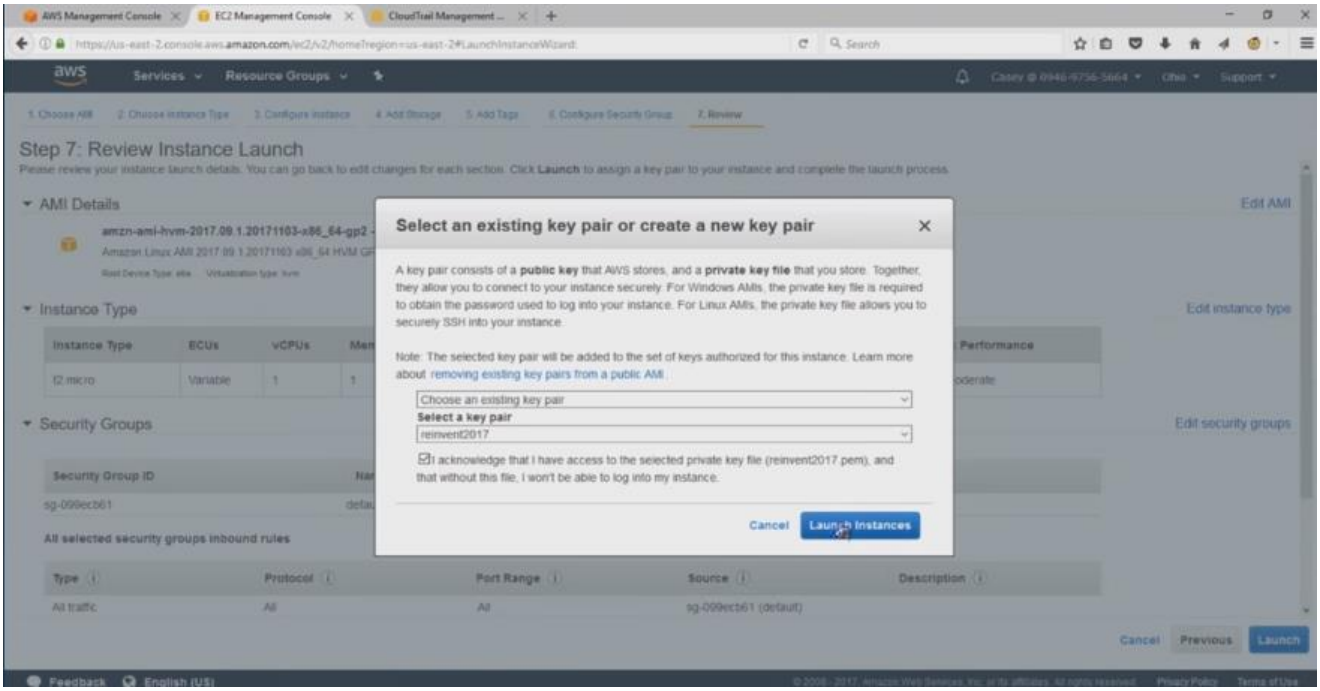

Services
Resource Groups
1. Choose AMI
2. Choose Instance Type
3. Configure Instance
4. Add Storage
5. Add Tags
6. Configure Security Group
7. Review
Step 7: Review Instance Launch
Please review your instance launch details. You can go back to edit changes for each section. Click Launch to assign a key pair to your instance and complete the launch process.
AMI Details
Edit AMI
amzn-ami-hvm-2017.09.1.20171103-x86_64-gp2 - ami-aa1b34cf
Amazon Linux AMI 2017.09.1.20171103 x86_64 HVM GP2
Instance Type
Edit instance type
Instance Type
ECUs
vCPUs
Memory (GiB)
Instance Storage (GB)
EBS-Optimized Available
Network Performance
t2.micro
Variable
1
1
EBS only
-
Low to Moderate
Security Groups
Edit security groups
Security Group ID
Name
Description
sg-099ecb61
default
default VPC security group
All selected security groups inbound rules
Type
Protocol
Port Range
Source
Description
All traffic
All
All
sg-099ecb61 (default)
Cancel
Previous
Launch
Feedback
English (US)

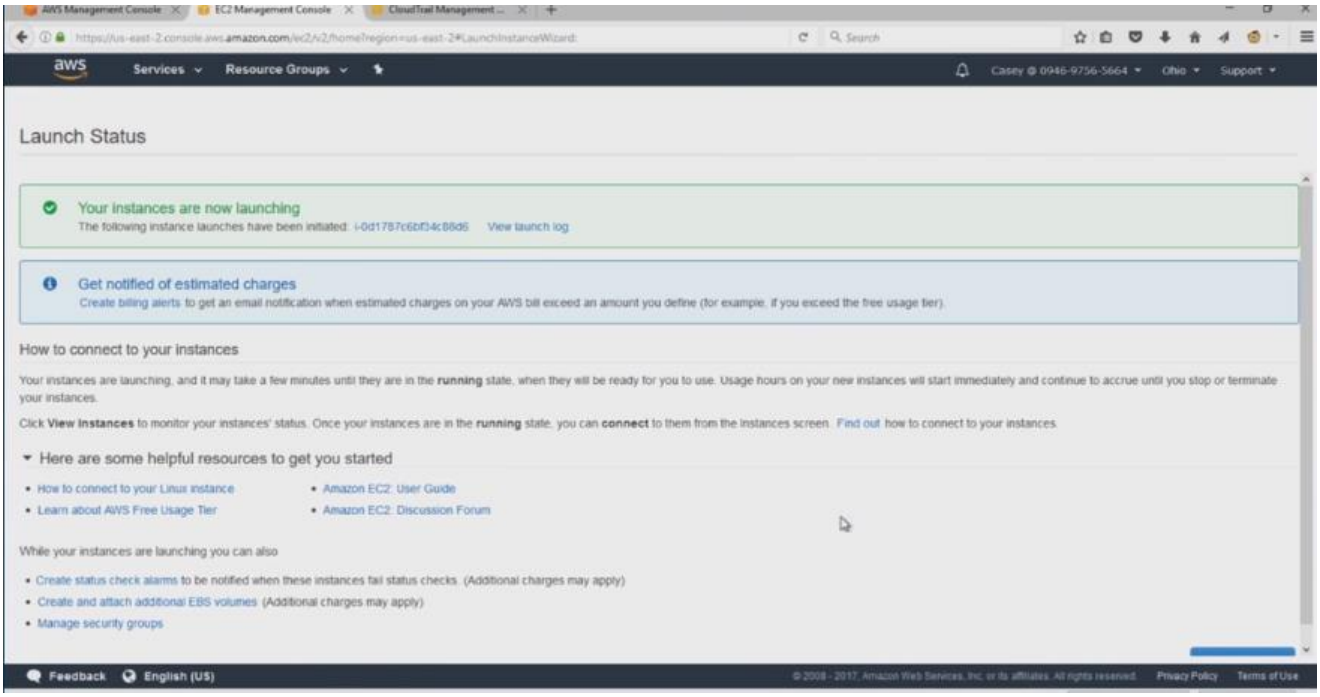




Casey can’t launch instances with tags that are not blackjack


Step 5: Add Tags
A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver.
A copy of a tag can be applied to volumes, instances or both.
Tags will be applied to all instances and volumes. Learn more about tagging your Amazon EC2 resources.
Key (127 characters maximum)
Value (255 characters maximum)
Instances
Volumes
project
blackjack
name
inthungry
Add another tag
(Up to 50 tags maximum)
Cancel
Previous
Review and Launch
Next: Configure Security Group
Feedback
English (US)


Step 7: Review Instance Launch
Select an existing key pair or create a new key pair
A key pair consists of a public key that AWS stores, and a private key file that you store. Together, they allow you to connect to your instance securely. For Windows AMIs, the private key file is required to obtain the password used to log into your instance. For Linux AMIs, the private key file allows you to securely SSH into your instance.
Note: The selected key pair will be added to the set of keys authorized for this instance. Learn more about removing existing key pairs from a public AMI.
Choose an existing key pair
Select a key pair
reinvent2017
I acknowledge that I have access to the selected private key file (reinvent2017.pem), and that without this file, I won't be able to log into my instance.
Cancel
Launch Instances
Edit AMI
Edit instance type
Edit security groups
Cancel
Previous
Launch



Casey can launch instances having the project/blackjack and name/<anything> tags as seen above

Policy: Control tags using EC2

```
{
    "Version": "2012-10-17",
    "Statement": [{
        "Effect": "Allow",
        "Action": "ec2:CreateTags",
        "Resource": "*",
        "Condition": {
            "StringEquals": {
                "ec2:ResourceTag/project": ["blackjack","poker"]
            },
            "ForAllValues:StringEquals": {
                "aws:TagKeys": ["project","name"]
            },
            "StringEqualsIfExists": {
                "aws:RequestTag/project": ["blackjack","poker"]
            }}}]}
```

Only tag resources with these tags

Tag with either of these keys

For *project*, you specify only these values

Allow users to create tags on resources tagged with project=blackjack *or* project=poker *and only tag with* project *or* name. *If they use* project, *they must use* blackjack *or* poker.

Additional resources

AWS Security Blog: Announcements, use cases, and best practices
https://aws.amazon.com/blogs/security/

Other Sessions:
SID303 - How You can use AWS' Identity Services to be Successful on Your AWS Cloud Journey

SID309 - Credentials, Credentials, Credentials, Oh My!

SID321 - How Capital One uses AWS Organizations