



CONCLUSION

[HOME](#)[AMIS VISION](#)[CAREERS @ AMIS](#)[EVENTS](#)[ABOUT](#)[CONTACT](#)**YOU ARE AT:** [Home](#) » [Cloud](#) » [AWS](#) » [Creating policy's, groups and users in AWS](#)

Creating policy's, groups and users in AWS

0

BY FREDERIQUE RETSEMA ON MARCH 7, 2020

AWS, CLOUD, DEVOPS

Today, I'll demonstrate how you can add policy's, groups and users within AWS. In a couple of days, I'll demonstrate the use of AWS Elastic Container Services (ECS) to a group of people. After the demonstration, they can play with ECS themselves.

It is, of course, not the intention to give these people permissions on AWS to other services than ECS: we don't want them for example to create DynamoDB tables, or create Route53 DNS entries. We also want to limit the use of ECS to region EU-Central-1 (Frankfurt).

Policies

The place to start is service IAM, menu policies. In this screen, you will see a lot of predefined policies.



ABOUT AUTHOR



Frederique Retsema

Frederique Retsema is active in IT since 1993. Senior Consultant and developer on diverse areas including SQL and Java. She likes to work with automation tools like Bamboo, Jenkins, Ansible, Terraform and CloudFormation.

[View all posts](#)

FOLLOW US ON LINKEDIN

[Following](#) 2,875

POPULAR TAGS

[Agile](#) [analytical function](#) [apex](#) [api](#) [application](#) [container](#) [cloud](#)[service](#) [Architecture](#) [Azure](#) [BPEL](#) [bpm](#) [cloud](#)[Database](#) [docker](#) [devops](#)

Groups	AccessAnalyzerServiceRolePolicy	AWS managed	None	Allow Access Analyzer to analyze resource metadata
Users	AmazonS3OutpostsAccess	Job function	Permissions policy (3)	Provides full access to AWS services and resources.
Roles	AlexaForBusinessDeviceSetup	AWS managed	None	Provides device setup access to AlexaForBusiness ser
Policies	AlexaForBusinessFullAccess	AWS managed	None	Grants full access to AlexaForBusiness resources and
Identity providers	AlexaForBusinessGatewayExecution	AWS managed	None	Provide gateway execution access to AlexaForBusine
Account settings	AlexaForBusinessNetworkProfileServicePolicy	AWS managed	None	This policy enables Alexa for Business to perform aut
Access reports	AlexaForBusinessPolicyDelegatedAccessPolicy	AWS managed	None	Provide access to Play AWS devices
Services analyzer				

We will create our own policy, so click on the blue button "Create policy". Now, click on the link "Choose a service"

Services

Resource Groups

Create policy

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in t

Visual editor

JSON

Expand all | Collapse all

Select a service

Service

Choose a service

Actions

Choose a service before defining actions

Resources

Choose actions before applying resources

Request conditions

Choose actions before specifying conditions

In this screen, services are shown either based on their abbreviation (like "EC2") or their full name (like "Elastic Container Services"). When you cannot find the services based on the abbreviation, just try the full name.

Service

close

Select a service below

Enter service r

Find a service

Access Analyzer

EI

Mobile Hub

Account

EKS

MQ

Alexa for Business

Elastic Beanstalk

MSK

Amplify

Elastic Block Store

Neptune

API Gateway

Elastic Container Registry

Network Manager

App Mesh

Elastic Container Service

OpsWorks

App Mesh Preview

Elastic Transcoder

CpsworksSCM

AppConfig

ElastiCache

Organizations

Application Auto Scaling

Elasticsearch Service

Outposts

Application Discovery

ELB

Performance Insights

Application Discovery Arsenal

ELB v2

Personalize

AppStream 2.0

EMR

Pinpoint

AppSync

EventBridge

Pinpoint Email

Artifact

EventBridgeSchemas

Pinpoint SMS Voice

Athena

ExecuteAPI

Polly

Auto Scaling

Firehose

Price List

Backup

Firewall Manager

Private Marketplace

Select a service

Service

close

Select a service below

Find a service

Elastic Container Registry

Elastic Container Service

Click on Elastic Container Services, you will now see all the actions that are possible within ECS. Click for example on List, you will see all the permissions that deal with showing ECS objects.

Visual editor

JSON

Import managed policy

Expand all | Collapse all

Elastic Container Service

Clone

Remove

Service

Elastic Container Service

Actions

close

Specify the actions allowed in Elastic Container Service

Switch to deny permissions

Filter actions

Manual actions (add actions)

All Elastic Container Service actions (ecs*)

Access level

List

Read

Tagging

Write

container Database Docker DVT h

integration iot **Java** javascript jms

json kafka kubernetes linux maven monitoring

node oci oracle cloud oracle cloud infrastructure

oracle database oracle xml db OSB paas performance

tuning **plsql** provisioning puppet push python **RES**

saas Scrum **soa sql** Vagrant virtual box

visualization vm XML

FOLLOW US ON TWITTER

Tweets by @AMISnl

AMIS Conclusion
@AMISnl

New Article : Creating policy's, groups and users in AV
ift.tt/3aAFXkn by Frederique Retsema

Creating policy's, groups and users in AWS - ...

Today, I'll demonstrate how you can add policy's, groups and users within AWS. In a couple of days, technology.amis.nl

AMIS Conclusion
@AMISnl

Gecontroleerd en herhaalbaar IT-omgevingen opbouw en bijwerken met de schaalbaarheid en flexibiliteit van gevirtualiseerde omgevingen, zoals Cloud. De voordelen? Lees ze in de [#knowwhyweekly](#) van An van Dalen.bit.ly/2xd98vz

Automation van Infrastructure as Code: gecon...

Automation van Infrastructure as Code: gecontroleerd en herhaalbaar IT-omgevingen
amis.nl

Mar 5, 20

AMIS Conclusion
@AMISnl

The latest The AMIS Oracle Technology Weekly!
paper.li/AMIS_Services/... #java #word

Getting started with NoSQL Database Service ...
technology.amis.nl This morning I discovered a new entry in the menu on my Oracle Cloud Infrastructure paper.li

Mar 2, 2020

AMIS Conclusion
@AMISnl

New Article : Getting started with NoSQL Database Service on Oracle Cloud Infrastructure ift.tt/2l8RACQ t Lucas Jellema

Getting started with NoSQL Database Service ...
This morning I discovered a new entry in the menu on my Oracle Cloud Infrastructure Tenancy (on technology.amis.nl

Feb 29, 2020

AMIS Conclusion
@AMISnl

Data is het nieuwe goud. Of het nieuwe Uranium? Van ruwe data naar de beloofde toegevoegde waarde? Van technische know-how naar praktische know-why; lees het in de nieuwe [#knowwhyweekly](https://www.knowwhyweekly.nl/) van Andre van Dal bit.ly/32zjbq#AMISTechnologyBlog #datascience

Is data het nieuwe uranium?
Is data het nieuwe uranium?
amis.nl

Feb 28, 2020

Embed

View on Twitter

META

Log in

Entries feed

Comments feed

WordPress.org

Resources Choose actions before applying resources

Request conditions Choose actions before specifying conditions

[Add additional permissions](#)

Visual editor JSON Import managed policy

Expand all Collapse all

Elastic Container Service [Clone](#) [Remove](#)

Service Elastic Container Service

Actions Specify the actions allowed in Elastic Container Service [Switch to deny permissions](#)

Filter actions

Manual actions (add actions)

All Elastic Container Service actions (ecs:*)

Access level

List

ListAccountSettings ListContainerInstances ListTaskDefinitionFamilies

ListAttributes ListServices ListTaskDefinitions

ListClusters ListTagsForResource ListTasks

Read

Tagging

I want to give the people in my workshop all the permissions in ECS, so I simply click on "All Elastic Container Service actions (ecs:*)". AWS assumes that you want to connect specific resources to the actions, so I get some warnings.

Visual editor JSON Import managed policy

Expand all Collapse all

Elastic Container Service (All actions) 3 warnings [Clone](#) [Remove](#)

Service Elastic Container Service

Actions Specify the actions allowed in Elastic Container Service [Switch to deny permissions](#)

Filter actions

Manual actions (add actions)

All Elastic Container Service actions (ecs:*)

Access level

List (9 selected)

Read (6 selected)

Tagging (2 selected)

Write (29 selected)

Resources Specify container-instance resource ARN for the DescribeContainerInstances and 7 more actions

Specify service resource ARN for the DeleteService and 4 more actions

Specify task-set resource ARN for the DescribeTaskSets and 2 more actions

Request conditions Specify request conditions (optional)

[Add additional permissions](#)

Character count: 39 of 6144

Cancel [Review policy](#)

Click on resources:

Elastic Container Service (All actions) 3 warnings [Clone](#) [Remove](#)

Service Elastic Container Service

Actions Manual actions

Resources Specific All resources

cluster You have not specified resource with type cluster. Add ARN to restrict access. Any

container-instance Specify container-instance resource ARN for the DescribeContainerInstances and 7 more actions. Add ARN to restrict access. Any

service Specify service resource ARN for the DeleteService and 4 more actions. Add ARN to restrict access. Any

task You have not specified resource with type task. Add ARN to restrict access. Any

task-definition You have not specified resource with type task-definition. Add ARN to restrict access. Any

task-set Specify task-set resource ARN for the DescribeTaskSets and 2 more actions. Add ARN to restrict access. Any

Request conditions Specify request conditions (optional)

[Add additional permissions](#)

Character count: 39 of 6144

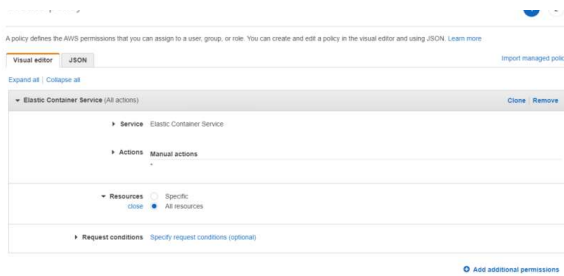
Cancel [Review policy](#)

Click on "all resources", so people will be able to create their own objects:

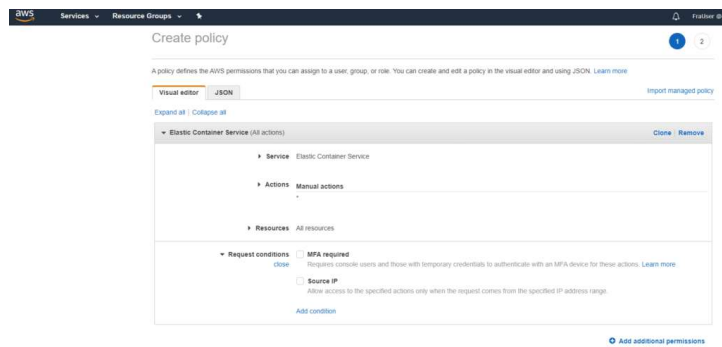
BWS Services Resource Groups

Create policy

2



To restrict on the region that is used, click on "Request conditions":



The condition that we need isn't shown yet, so click on "Add condition":

Add request condition

Condition key

Select condition key

Qualifier

Default

Operator

Select operator

Value

Enter Value

Cancel

Add

When you open the condition key, you will see a lot of options. We need something with "region", when we scroll through this list, we find `aws:RequestedRegion`:

Add request condition

Condition key

aws:RequestedRegion

Qualifier

aws:PrincipalType

Operator

aws:Referer

Value

aws:RequestTag

Cancel

Add

In qualifier, we select "For all values in request":

Add request condition

Condition key

aws:RequestedRegion

Qualifier

Default

Operator

Default

For any value in request

For all values in request

Enter value

Value

Cancel

Add

In operator, we select "String Equals":

Add request condition

✕

Condition key

aws:RequestedRegion

Qualifier

For all values in request

Operator

Select operator

StringEquals

StringNotEquals

StringEqualsIgnoreCase

StringNotEqualsIgnoreCase

StringLike

StringNotLike

Null

Value

Cancel

Add

For Frankfurt, we can use the text "eu-central-1". For a complete list of regions, see this AWS webpage [1].

Add request condition

✕

Condition key

aws:RequestedRegion

Qualifier

For all values in request

Operator

StringEquals

☐ If exists

Value

eu-central-1

+ Add another condition value

Cancel

Add

Now, click on Add:

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. [Learn more](#)

Visual editor JSON Import managed policy

Expand all Collapse all

▼ Elastic Container Service (All actions)

Service Elastic Container Service

Actions Manual actions

Resources All resources

Request conditions

☐ MFA required

Requires console users and those with temporary credentials to authenticate with an MFA device for these actions. [Learn more](#)

☐ Source IP

Allow access to the specified actions only when the request comes from the specified IP address range.

☒ aws:RequestedRegion (StringEquals eu-central-1) (Edit Remove)

[Add another condition](#)

[Add additional permissions](#)

Character count: 192 of 6,144

Cancel Review policy

Without being aware of it, we just created a JSON file for permissions. Let's look at that JSON, by clicking on the JSON tab on top of this screen:

Create policy

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. [Learn more](#)

Visual editor JSON Import managed policy

```

1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "VisualEditor0",
6       "Effect": "Allow",
7       "Action": "ecs:*",
8       "Resource": "*",
9       "Condition": {
10        "ForAllValues:StringEquals": {
11          "aws:RequestedRegion": "eu-central-1"
12        }
13      }
14    }
15  ]
16 }
```

This looks fine to me, so click the "Review Policy" button on the bottom of the screen. We can give a name and a description here:

Create policy

Review policy

Name ECSWorkshop

Description ECS Workshop on 10 March 2020

Summary

Service	Access level	Resource	Request condition
Allow (1 of 224 services) Show remaining 223			
Elastic Container Service	Full access	All resources	aws:RequestedRegion = eu-central-1

* Required

Cancel Previous Create policy

After that, click on "Create Policy" – we're done!

Groups

I'd like to have all users in the same group. So, click on groups:

console.aws.amazon.com/iam/home?region=eu-west-1#/policies

aws Services Resource Groups

Identity and Access Management (IAM)

Dashboard

Access management

Groups

Users

Roles

Policies

Identity providers

Account settings

ECSWorkshop has been created.

Create policy Policy actions

Filter policies Search

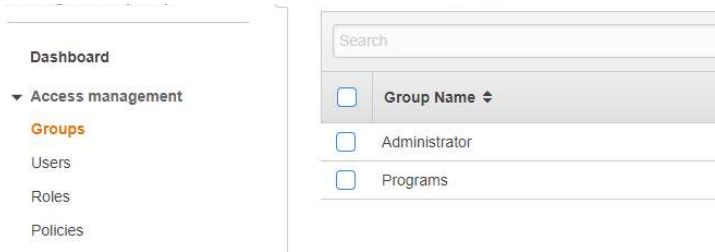
	Policy name	Type
<input type="radio"/>	AccessAnalyzerServiceRolePolicy	AWS managed
<input type="radio"/>	AdministratorAccess	Job function
<input type="radio"/>	AlexaForBusinessDeviceSetup	AWS managed

Create a new group:

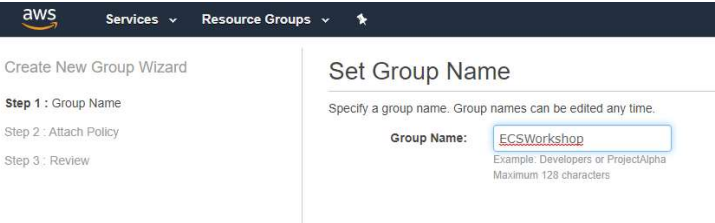
aws Services Resource Groups

Identity and Access Management (IAM)

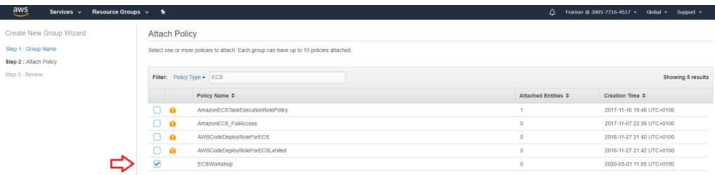
Create New Group Group Actions



Give a group name:



Click on the checkbox before the previous created policy. You can find it, by typing the first characters of the policy name:

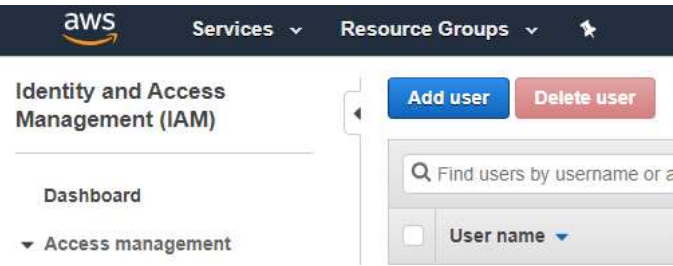


You can see that from these five policies, the first four are default policies that are created by AWS. If possible, use these policies instead of your own ones: when services change, AWS will change the policies with them. In our case, we need a specific one, because AWS doesn't restrict the access to regions by default.

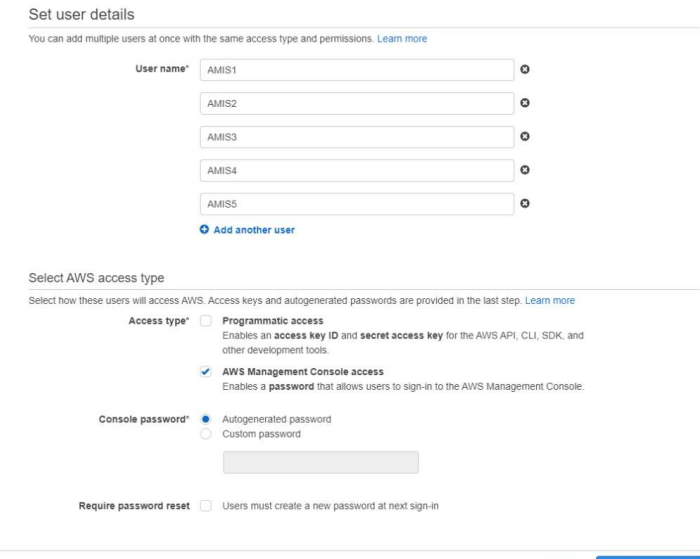
When you click on "Next step", you can review your changes. You can click on the "Create group" button on the bottom of the screen to create this group.

Users

Click in the menu on the left on Users, you will see a button to add a new user:



Fortunately, we can add a maximum of 10 users at the same time. My workshop will be attended by five people, so I will add the users AMIS1 to AMIS5 to this list:



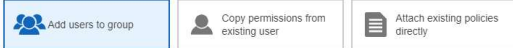
I want them to use the AWS console, they don't need to use the Command Line Interface (CLI) or programs to connect with these users. By using an auto generated password, they will get a strong password. The password doesn't need to be reset: by un-clicking this checkbox, I encourage that they will keep using the strong passwords that I give them: there are no IAM policy rules in the policy that we just wrote, though they will be able to change the password by clicking on their account name and using menu option My Security Credentials.

When you click on "Next: permissions", you can add permissions. We already created a group for this workshop, so we only have to click the checkbox before ECSWorkshop.

Add user

1 2 3 4 5

Set permissions



Add users to an existing group or create a new one. Using groups is a best-practice way to manage users' permissions by job functions. [Learn more](#)

Add user to group

Create group Refresh

Search		Showing 3 results
Group	Attached policies	
<input type="checkbox"/> Administrator	AllowAdministratorRole	
<input checked="" type="checkbox"/> ECSWorkshop	ECSWorkshop	

Click in the bottom of the screen on "Next: Tags". Tags can be used for many things, one of them is letting your colleagues know which users use these accounts. Let's add a tag "Workshop" with value "AMIS 10 March":

Add user

1 2 3 4 5

Add tags (optional)

IAM tags are key-value pairs you can add to your user. Tags can include user information, such as an email address, or can be descriptive, such as a job title. You can use the tags to organize, track, or control access for this user. [Learn more](#)

Key	Value (optional)	Remove
Workshop	AMIS 10 March	X
Add new key		

You can add 49 more tags.

Click on the button "Next: Preview": you can look at the configuration that should be added. When this is right, press the "Create users" button.

Review

Review your choices. After you create the users, you can view and download autogenerated passwords and access keys.

User details

User names	AMIS1, AMIS2, AMIS3, AMIS4, and AMIS5
AWS access type	AWS Management Console access - with a password
Console password type	Autogenerated
Require password reset	No
Permissions boundary	Permissions boundary is not set

Permissions summary

The users shown above will be added to the following groups.

Type	Name
Group	ECSWorkshop

Tags

The new users will receive the following tag

Key	Value
Workshop	AMIS 10 March

In the next screen, you can get the password or send logon instructions to an e-mail account. Please mind, that it is impossible to get the password after this step. For my workshop, it is nice to get the passwords via the button "Download .csv":

Add user

1 2 3 4 5

Success

You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://300577164517.signin.aws.amazon.com/console>

Download .csv

	User	Password	Email login instructions
▶	AMIS1	***** Show	Send email ↗
▶	AMIS2	***** Show	Send email ↗
▶	AMIS3	***** Show	Send email ↗
▶	AMIS4	***** Show	Send email ↗
▶	AMIS5	***** Show	Send email ↗

When you open this file, you will see something like this:

	A	B	C	D	E
1	User name	Password	Access key ID	Secret access key	Console login link
2	AMIS1	abCd-gF1G2#			https://123456789012.signin.aws.amazon.com/console
3	AMIS2	1~2abC!@#\$De			https://123456789012.signin.aws.amazon.com/console
4	AMIS3	~AB!@Cde1#\$\$			https://123456789012.signin.aws.amazon.com/console
5	AMIS4	ab~1cDE!f2g@			https://123456789012.signin.aws.amazon.com/console
6	AMIS5	1ab2~1c3DE@#			https://123456789012.signin.aws.amazon.com/console

Testing...

When you would use the newly created users AMIS1 to AMIS5 to create (f.e.) a cluster in ECS, this will not work. This is, because ECS depends heavily on other AWS services. How to solve this will be the topic of my next blog.

Conclusion

The wizard is very useful, to have a quick overview of the names that we can use to allow or deny tasks in an AWS service. This will work for simple services like EC2 or VPC, but not for services (like ECS) that depend on other AWS services. How to find out which services this specific service depends on and how we can achieve our goal of restricting access to our workshop users will be explained in my next blog.

Footnote

[1] <https://docs.aws.amazon.com/general/latest/gr/rande.html>

Related Posts:



Differences between CloudFormation, Amazon web services



Application deployment to Oracle



Docker, WebLogic Image on Amazon



How to start with Amazon cloud server



Oracle Compute Cloud – Uploading

[AWS](#) [Groups](#) [IAM](#) [policies](#) [Users](#) [Wizard](#)


ABOUT AUTHOR



FREDERIQUE RETSEMA


Frederique Retsema is active in IT since 1993. Senior Consultant and developer on diverse areas including SQL and Java. She likes to work with automation tools like Bamboo, Jenkins, Ansible, Terraform and CloudFormation.

RELATED POSTS




FEBRUARY 26, 2020 0

Setup and use of oVirt on CentOS7



FEBRUARY 15, 2020 0

Using PXE to deploy a DNS server



FEBRUARY 8, 2020 1

Deploying CentOS 8 using PXE

Leave a Reply

Enter your comment here...

This site uses Akismet to reduce spam. [Learn how your comment data is processed.](#)

