Identity and access management

Tom Rochette <tom.rochette@coreteks.org>

December 21, 2025 — 77e1b28a

0.1 Context

0.2 Learned in this study

0.3 Things to explore

1 Overview

2 AWS

AWS has the concept of groups, users, roles and policies.

2.1 Groups

A group is a list of users to which are assigned policies (permissions).

A useful feature they also provide is the *Access Advisor*, which allows administrators to observe when certain policies are being used and by which users.

To keep things simple, groups cannot be nested into other groups.

2.2 Users

Users represent an entity that has access to the AWS platform. They may access AWS either programmatically and/or through the console to administer the account.

During the using creation process, the creator has the ability to assign the user to a group, copy permissions from an existing user or attach policies to the user.

Upon creation of a user with programmatic access, the user is generally given an access key with its corresponding secret.

2.3 Roles

Roles are very similar to groups conceptually, but instead of being "persistent", they are temporary. While you will generally assign one or many groups to a user, roles are not assigned to a user (or vice-versa, a user is not assigned one or many roles) but given based on certain criteria. A user takes a given role and only receives this role's policies/permissions until he returns to his own identity.

2.4 Policies

A policy is a set of rules that determines the permissions given to a user, group or role. It has a name, description and a policy document, which describes the permissions.

The policy document is a JSON formatted object which contains a version and a list of statements. Each statement has an effect (Allow/Deny), a list of actions and a list of resources to which it applies.

You can learn more about AWS policies evaluation logic. Their IAM Policy Reference might also prove useful in understanding the various bits that compose the policy document.

2.4.1 Actions

Within AWS, each service has a unique lowercase identifier[1]. Within each of these services, a list of actions exists, which can be granted (or denied) to an identity. Examples of actions are:

- `*`: Allowed to use all actions under all services
- `s3:*`: Allowed to use all actions under the `s3` service
- `s3:GetObject`: Allowed to use the `GetObject` action under the `s3` service

2.4.2 Resources

Resources represent entities within their given service. For example, `arn:aws:s3:::some-bucket/*` represents the content under the `some-bucket` bucket in the `s3` service.

Format : `arn:$partition:$service:$region:$account-id:$resource`[2]

3 See also

- Policy evaluator

4 References

- https://blog.gopheracademy.com/advent-2015/hydra-auth/

[1] http://docs.aws.amazon.com/general/latest/gr/aws-arns-and-namespaces.html#genref-aws-service-namespaces
[2] http://docs.aws.amazon.com/general/latest/gr/aws-arns-and-namespaces.html#genref-arns