AWS Solution Architect Training

Module 02
Identity and Access Management (IAM)

Instructor: Tim Platt, Cloud Solution Architect

Identity and Access Management (IAM)

Identity – Authentication and Authorization

Key Points

- Authentication – prove WHO you are – by supplying a username, password, multi-factor authentication (MFA) code
- Authorization – WHAT are you allowed to do? Lists of permissions (called policies) that describe what the user is allowed to do in AWS
- **Identities** (User, Group, Role) can have permissions to take actions against resources
- **Policies** are written to describe the permissions that the Identities have.

IAM Identities

IAM identities can have permissions attached to them – which allows them to take ACTIONs against RESOURCES – such as launching an EC2 instance, deleting an S3 bucket, etc.

IAM User

Users are human beings
(or should be)

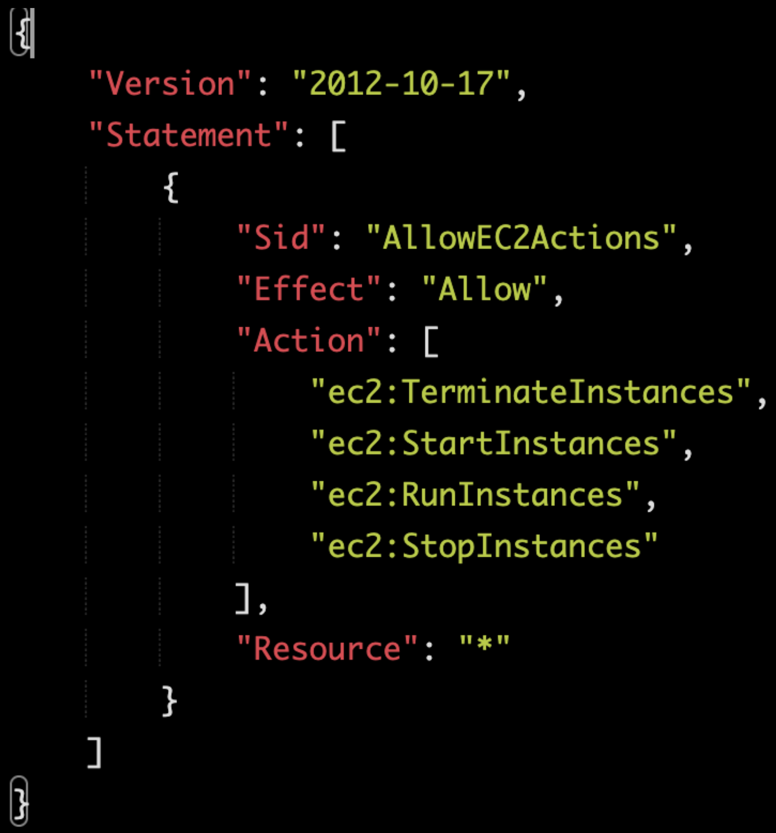
IAM Group (of Users)

We can put Users with similar
needs in a Group (of Users)

IAM Roles

Roles are **temporary** credentials
These can be utilized by Users
or Services / Resources / Servers

IAM Identity Policy Example



```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "AllowEC2Actions",
            "Effect": "Allow",
            "Action": [
                "ec2:TerminateInstances",
                "ec2:StartInstances",
                "ec2:RunInstances",
                "ec2:StopInstances"
            ],
            "Resource": "*"
        }
    ]
}
```

What does this allow someone to do?

It allows them to create (”Run”), start, stop, and delete any EC2 instance in the account

NOTE: This is an Identity Policy. It is attached to an Identity (User, Group, Role)

IAM Resource Policy Example

```
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "PublicReadGetObject",
         "Effect": "Allow",
         "Principal": "*",
         "Action": "s3:GetObject",
         "Resource": "arn:aws:s3:::BUCKET-NAME-HERE/*"
      }
   ]
}
```

What does this allow someone to do?

It allows ANYONE to anonymously download a file from the bucket.

NOTE: This is a Resource Policy. It is attached to some resource, such as an S3 bucket

Let’s create a bucket, and make it Public

This is a two-step process:

1. Turn OFF “Block Public Access”
2. Add a Bucket Policy (Resource Policy) that allows “GetObject” for * (that ‘s a wildcard meaning ANYONE)

IAM Roles

IAM Role

Key Points

- An IAM Role is an Identity
- It can be given permissions to do things
- The role can be used by:
  - Human beings (IAM Users)
  - AWS Services (Such as AWS Lambda, EC2, etc.)
  - By our application code (Python, JavaScript, C#, Java programs that we write
  - Used in ”Identity Federation” scenarios (Single Sign On)
  - Can be used to access resources in different AWS accounts (Cross-Account access)

SUPERPOWER: The Role provides a set of TEMPORARY credentials – they expire in a very short amount of time – and security teams LOVE this

Let’s create an IAM Role and assign it to our EC2 – so it can have permissions

We’re going to launch our “Apache Web Server Simple Example” again, and give it permissions to talk with “AWS Systems Manager” (SSM) so we can manage the server easily

We’ll create the IAM Role first

1. Create an IAM role that the EC2 service can ”assume”
2. Attach this policy to the role: AmazonSSMManagedInstanceCore
3. Launch the EC2 instance (From AMI) and set the “Instance Profile” to the IAM Role just created. (This is under “Advanced Details”)

Services that help us manage EC2 Instances at scale

These services are used by cloud system administrators to perform the necessary work to manage a fleet of computer servers

AWS Systems Manager (SSM)

Manage our EC2 instances remotely (patching, maintenance, etc.)

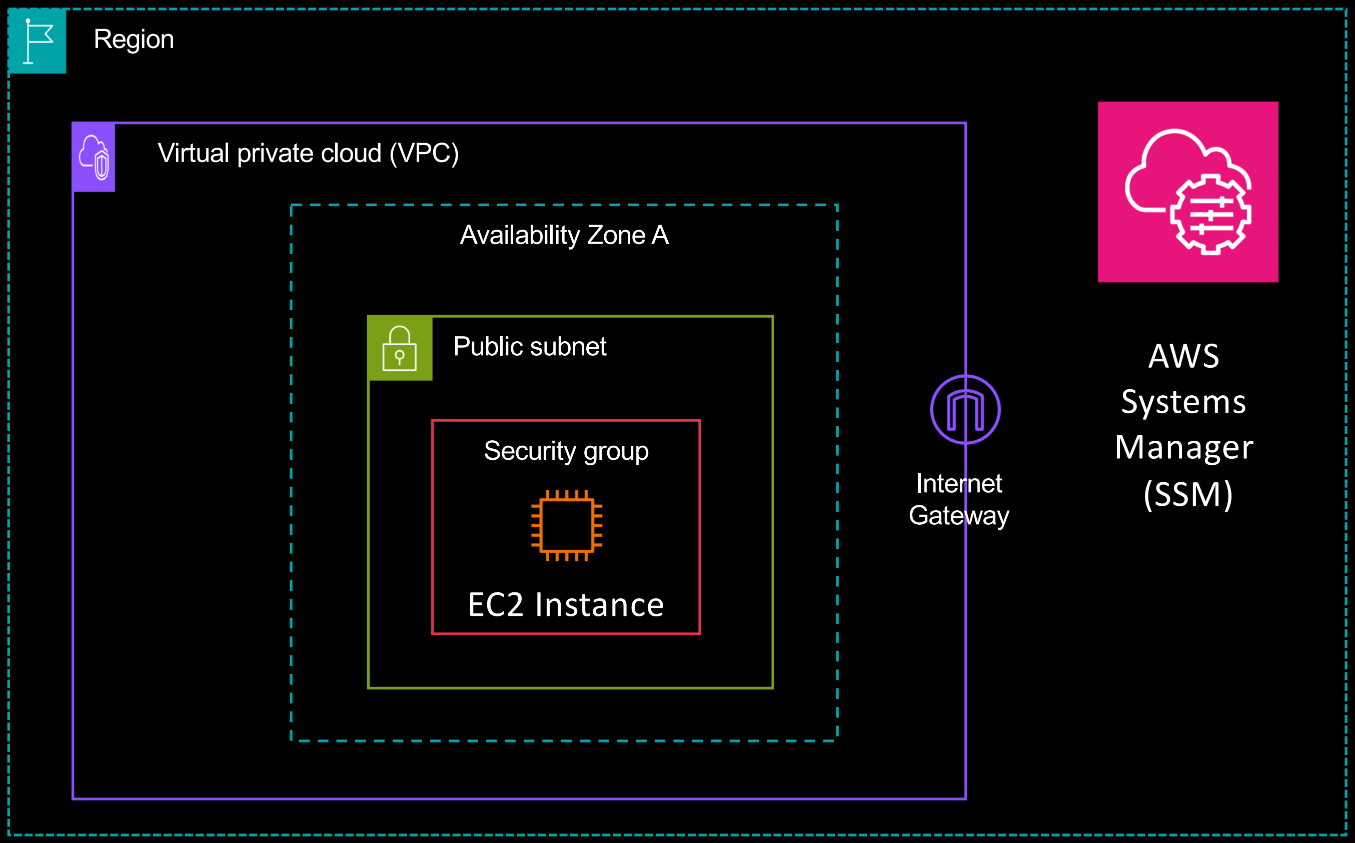
Amazon Inspector

Scan our EC2 instances (and other things) for VULNERABILITIES (things that need to be patched)

AWS Config

Understand CHANGES and CONFIGURATION SETTINGS and enforce RULEs about configuration

What we’ve just accomplished in our AWS Account



AWS SSM is a WEB SERVICE (accessed over the Internet).

We can manage our web server with SSM , provided:

1. Our server has OUTBOUND INTERNET access (to talk to the SSM service)
2. Our EC2 instance has PERMISSIONS to be allowed to talk to the SSM Service (via Role and attached Policy)

Links

- AWS IAM User Guide: https://docs.aws.amazon.com/IAM/latest/UserGuide/introduction.html
- IAM Best Practices (Know these for the exam!): https://docs.aws.amazon.com/IAM/latest/UserGuide/best-practices.html
- Identities (User, Role, User Group): https://docs.aws.amazon.com/IAM/latest/UserGuide/id.html
- IAM Policy Examples: https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_examples.html