

1.2 Understanding EC2

In this course, I will :

- Review EC2 instance settings related to Windows and Linux virtual machines
- Deploy Linux instances using the AWS Management Console, AWS CLI and AWS Tools for PowerShell followed by remotely managing those instances through SSH
- Deploy Windows instances using the AWS Management Console, the AWS CLI and AWS Tools for PowerShell followed by remotely managing windows instances using RDP
- Resize EC2 instances

AWS EC2 Instances

- Cloud based VM (Virtual Machine)

Amazon Machine Image (AMI)

- EC2 instance template
- Required when launching an EC2 instance

EC2 AMI Categories

- Windows server
- Linux variants: Ubuntu or CentOS
- MacOS

You can also specify the underlying horsepower for your virtual machine, for your instance.

EC2 Instance Type

- vCPUs: can have number of virtual CPU to balance the workload
- RAM: additional to match your workload
- Architecture: x86, x64
- Network Performance

Launching EC2 instance

- Security group: which traffic is allowed
- Storage: we can add disks to VM to add the extra storage required
- Network interfaces: multi-homed VM. Can add one or two networks to match the workload
- Tags: specifies key – value pairs. Metadata. Optional, don't have to define them

- VPC and subnet
- Auto-assign public IP
- Domain join directory
- IAM role
- Shutdown behavior
- CloudWatch detailed monitoring

Public/private key pair

- Public key is stored in AWS
- Private key is stored on your computer
- Windows instances: private key decrypts administrator password
- Linux instances: private key allows SSH public key authentication

Deploying windows instance using CLI

```
Administrator: Command Prompt
D:\>aws ec2 run-instances --image-id ami-03295ec1641924349 --count 1 --instance-type t2.micro --k
ey-name Pair1 --security-group-ids sg-aa748081 --subnet-id subnet-d5a244f4 --region us-east-1
```

```
D:\>aws ec2 create-tags --resources i-0198b64d3428f49c7 --tags Key=Name,Value=WinSrv2019-3 --regi
on us-east-1

D:\>aws ec2 describe-instances --region us-east-1
```

```
Administrator: Command Prompt
D:\>aws ec2 describe-instance-status --instance-ids i-0198b64d3428f49c7 --region us-east-1
```

```
D:\>aws ec2 stop-instances --instance-ids i-0198b64d3428f49c7 --region us-east-1
```

```
{
  "StoppingInstances": [
    {
      "CurrentState": {
        "Code": 64,
        "Name": "stopping"
      },
      "InstanceId": "i-0198b64d3428f49c7",
      "PreviousState": {
        "Code": 16,
        "Name": "running"
      }
    }
  ]
}
```

```
D:\>_
```

```
D:\>aws ec2 terminate-instances --instance-ids i-0198b64d3428f49c7 --region us-east-1
```

```
{
  "TerminatingInstances": [
    {
      "CurrentState": {
        "Code": 48,
        "Name": "terminated"
      },
      "InstanceId": "i-0198b64d3428f49c7",
      "PreviousState": {
        "Code": 80,
        "Name": "stopped"
      }
    }
  ]
}
```

```
D:\>_
```

Deploying windows instance using PowerShell

```
Administrator: Windows PowerShell
PS D:\> New-EC2Instance -ImageId ami-03295ec1641924349 -MinCount 1 -MaxCount 1 -KeyName Pair1 -SecurityGroupId sg-aa748081 -InstanceType m1.small -SubnetId subnet-d5a244f4 -Region us-east-1

GroupNames      : {}
Groups          : {}
Instances       : {Pair1}
OwnerId         : 483297109440
RequesterId     : 
ReservationId   : r-09b8a1ab4ab6c3a68

PS D:\>
```

```
Administrator: Windows PowerShell
PS D:\> get-ec2instancestatus -instanceid i-0d6b75f9dd5070451 -region us-east-1

AvailabilityZone : us-east-1c
Events          : {}
InstanceId      : i-0d6b75f9dd5070451
InstanceState   : Amazon.EC2.Model.InstanceState
OutpostArn      : 
Status         : Amazon.EC2.Model.InstanceStatusSummary
SystemStatus    : Amazon.EC2.Model.InstanceStatusSummary

PS D:\> (get-ec2instancestatus -instanceid i-0d6b75f9dd5070451 -region us-east-1).instancetype

Code Name
----
16    running

PS D:\>
```

```
PS D:\> New-EC2Tag -ResourceId i-0d6b75f9dd5070451 -Tag @{Key="Name"; Value="WinSrv2019-2"} -region us-east-1
PS D:\>
```

```
Administrator: Windows PowerShell
PS D:\> get-ec2tag -Filter @({Name="resource-type";Values="instance"}) -Region us-east-1

Key              ResourceId      ResourceType Value
---              -
Name             i-01e99b09c82d94041 instance      WinSrv2019-1
Name             i-035ba7e684b88a466 instance      AmazonLinux-1
Name             i-03db36617fd12569d instance      CloudFormation-Linux-1
aws:cloudformation:logical-id i-03db36617fd12569d instance      EC2Instance
aws:cloudformation:stack-id i-03db36617fd12569d instance      arn:aws:cloudformation:us-east-1:483297109440:s...
aws:cloudformation:stack-name i-03db36617fd12569d instance      MyLinuxStack
Name             i-04d003ae2ea2182db instance      AmazonLinux-2
Name             i-06396ee3e30479b29 instance      AmazonLinux-3
Name             i-0d6b75f9dd5070451 instance      WinSrv2019-2
Name             i-0ecd2dc6f56901beb instance      AmazonLinux-2

PS D:\>
```

Administrator: Windows PowerShell

```
PS D:\> stop-ec2instance -instanceid i-0d6b75f9dd5070451 -region us-east-1
```

CurrentState	InstanceId	PreviousState
-----	-----	-----
Amazon.EC2.Model.InstanceState	i-0d6b75f9dd5070451	Amazon.EC2.Model.InstanceState

```
PS D:\>
```

Deploying windows instance using Template

Description: Basic Windows Server 2019 template.

Parameters:

VpcId:

Description: VPC id

Type: AWS::EC2::VPC::Id

Default: vpc-542e1a2e

SubnetId1:

Description: Subnet Id where instance will create

Type: AWS::EC2::Subnet::Id

KeyName:
Description: Name of an existing EC2 KeyPair
Type: AWS::EC2::KeyPair::KeyName
ConstraintDescription: must be the name of an existing EC2 KeyPair.

I

InstanceType:
Description: EC2 instance type
Type: String
Default: t2.micro
AllowedValues: [t2.micro]
ConstraintDescription: Must be a valid EC2 instance type.

^

^^

RDPLocation:
Description: The IP address range that can be used to RDP to the EC2 instances
Type: String
MinLength: 9
MaxLength: 18
Default: 0.0.0.0/0
AllowedPattern:
(\d{1,3})\.\d{1,3})\.\d{1,3})\.\d{1,3})/(\d{1,2})
ConstraintDescription: must be a valid IP CIDR range of the form x.x.x.x/x.

I

LatestAmiId:
Type: 'AWS::SSM::Parameter::Value<AWS::EC2::Image::Id>'
Default: '/aws/service/ami-windows-latest/Windows_Server-2019-English-Full-Base'

Resources:

InstanceSecurityGroup:
Type: AWS::EC2::SecurityGroup
Properties:
VpcId: !Ref VpcId
GroupDescription: Enable RDP access

I

Resources:

```
InstanceSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    VpcId: !Ref VpcId
    GroupDescription: Enable RDP access
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 3389
        ToPort: 3389
        CidrIp: !Ref 'RDPLocation'
```

```
EC2Instance:
  Type: AWS::EC2::Instance
  Properties:
    InstanceType: !Ref InstanceType
    SecurityGroupIds:
      - !Ref InstanceSecurityGroup
    KeyName: !Ref KeyName
    ImageId: !Ref LatestAmiId
    SubnetId: !Ref SubnetId1
```

```
Outputs:
  InstanceId:
    Description: InstanceId of the newly created EC2 instance
    Value: !Ref 'EC2Instance'
  AZ:
    Description: Availability Zone of the newly created EC2
```