AWS

Q. How to modify EBS storage capacity?

Ans: We can modify the EBS storage capacity in two ways.

1. Through Snapshot (Old Method)

* Create a snapshot of the volume which you wants to modify. Note: Make sure instance is stopped while creating the snapshot.
* Create volume form the snapshot and while creating the snapshot give the required volume size and type.
* Then detach the old EBS volume from the EC2 and attach the new EBS volume which you have created from snapshot with required volume capacity.
* Make sure while attaching the volume make it root device by providing the proper Device type/path. Exa: linux= /dev/xvda and Windows= /dev/xvda

Note: **Please check the device type of existing EBS volume before detaching it.**

1. Through Clicking the Modify volume (Latest technology)

* Select the volume – click on Action (or right click on volume) – click on **Modify** volume – Then provide the volume size or type.

Q. What is the Different types of Tenancy of EC2 instance?

Ans: I. Shared – Run a shared hardware instance: Means the host hardware will be shared with other customers also. Each time you shut down and start instance it may start in different hardware.

Note: In reboot instance won’t change the host hardware. It will reboot in same host.

II. Dedicated – Run a dedicated instance: Means Dedicated Instances are Amazon EC2 instances that run in a VPC on hardware that's dedicated to a single customer. Your Dedicated instances are physically isolated at the host hardware level from instances that belong to other AWS accounts. Dedicated instances may share hardware with other instances from the same AWS account that are not Dedicated instances. Here also . Each time you shut down and start instance it may start in different hardware but in that hardware there won’t be any other customer’s instance.

Disadvantages: If particular hardware goes down, then your all instance will be down and it wont auto recover.

Note: In reboot instance won’t change the host hardware. It will reboot in same host.

III. Dedicated host – Lunch This instance into Dedicated Host: That means instance always will be in a particular hardware host. No matter how many time you shut down or start the instance will be always lunch only in the particular hardware.

**Note: Status Check**

1. System status Check: check the status of host Hardware
2. Instance status check: Check the status of your instance.

**Auto Recovery:** If the host hardware goes down, then your EC2 instance will be rebooted (on new hardware if necessary) but will retain its Instance Id, IP Address, Elastic IP Addresses, EBS Volume attachments, and other configuration details. In order for the recovery to be complete, you’ll need to make sure that the instance automatically starts up any services or applications as part of its initialization process.

**Applicable Instance Types and Environments**  
This feature is currently available for the C3, C4, M3, R3, and T2 instance types running in the US East (N. Virginia) region; we plan to make it available in other regions as quickly as possible.  The instances must be running within a VPC, must use EBS-backed storage, but cannot be Dedicated Instances.

Q. How to configure auto recovery?

Url: <https://aws.amazon.com/blogs/aws/new-auto-recovery-for-amazon-ec2/>

Note: Elastic IP address is chargeable if you are not using or attached to a instance and instance is stoped. You won’t be charged is the Elastic IP is attached to an instance and the instance is running.

Every Elastic IP can be assigned against one particular private IP.

**Security Group**

* Security groups are applied to instances
* By default allow all outbound traffic
* There is no deny rule
* There is no preference (priority)
* We can assign multiple groups to Instance

Note: If we will block all outbound traffic, it won’t access any web site or network. But it can communicate and transfer the information through inbound request (through allowed inbound traffic).

**Network ACL**

* NACL applied to VPC
* Outbound traffic blocked and always filtered
* We can deny particular IP or network for inbound and outbound traffic.
* We can set the priority by using Rule number( lowest number has the highest priority)
* We can assign only one NACL per subnet

**VPC Peering**

**Condation/Restrictions**

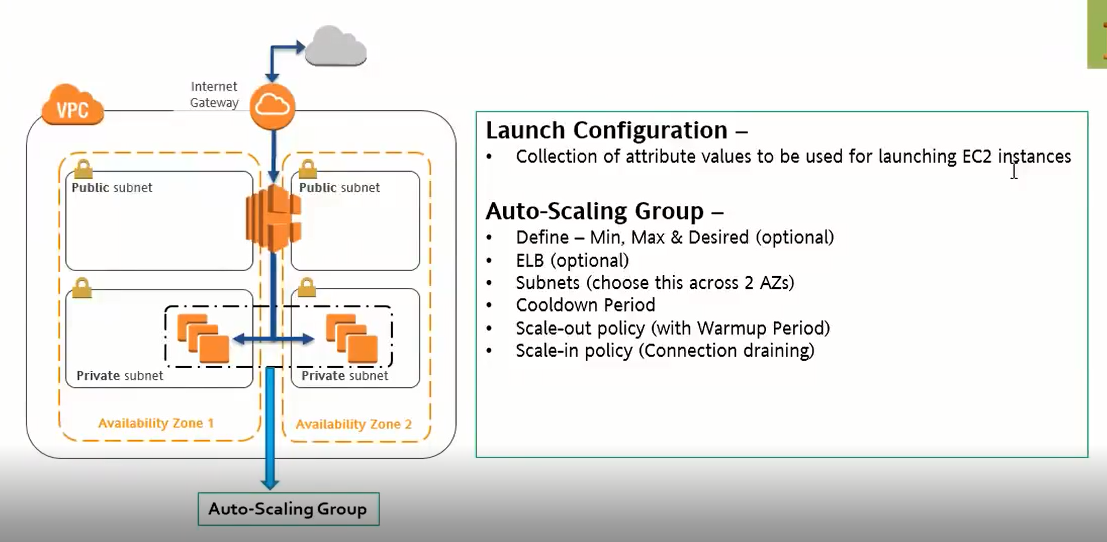
* No overlapping IP ranges between 2 VPCs
* No transitive peering, edge routing, IGW access across VPCs
* No NAT routing between VPCs
* Cannot resolve private DNS values across VPCs
* No cross-referencing of security groups between VPCs

1. Owners of both VPCs need to confirm the peering request.
2. The 2 VPCs could be in same or different AWS accounts.
3. The 2 VPCs could be in same AWS region or different region
4. We need to update route table in both VPCs after peering is done
5. The traffic flow between 2 VPCs will happen via private network

**NAT ELB**

* Equivalent to forward proxy  **.** Equivalent of reverse proxy
* Should be placed always in public subnet **.** Can be placed in public/private network
* Traffic originated from private instances **.** Traffic originated from internet served by going to Internet private or public instance
* Managed service. Lived in one subnet **.** Managed service. Spans across multiple subnets.

**Auto Scaling**



**Amazon EC2 Auto Scaling supports the following custom termination policies**:

* OldestInstance. Terminate the oldest instance in the group. This option is useful when you're upgrading the instances in the Auto Scaling group to a new EC2 instance type. You can gradually replace instances of the old type with instances of the new type.
* NewestInstance. Terminate the newest instance in the group. This policy is useful when you're testing a new launch configuration but don't want to keep it in production.
* OldestLaunchConfiguration. Terminate instances that have the oldest launch configuration. This policy is useful when you're updating a group and phasing out the instances from a previous configuration.
* ClosestToNextInstanceHour. Terminate instances that are closest to the next billing hour. This policy helps you maximize the use of your instances and manage your Amazon EC2 usage costs.
* Default. Terminate instances according to the default termination policy. This policy is useful when you have more than one scaling policy for the group.
* OldestLaunchTemplate. Terminate instances that have the oldest launch template: first the non-current launch template followed by the oldest version of the current launch template. This policy is useful when you're updating a group and phasing out the instances from a previous configuration.
* AllocationStrategy. Terminate instances in the Auto Scaling group to align the instance distribution to the allocation strategies for On-Demand and Spot Instances, your current selection of instance types, and distribution across your N lowest priced Spot pools

**Q. What is Jumpbox / Bastion hpst?**

Ans: Bastion host is nothing but a small machine in your VPC which has the public IP and we are using this machine to access other machines from our private network from public network.

We will connect to the Basion host using it’s public IP and then we can connect any other host inside our private network and do the work.

Q. How to record SSH sessions established to our VPC or instances inside the VPC?  
 Ans: Through enabling VPC flow logs.

Warm-up Time: The time the system will wait for coming to service after lunched through autoscaling.

Q. What is VPC endpoint?

Ans: It is basically used to access the service from outside VPC to inside VPC private network(private).

Use Case: Our S3 won’t be in side VPC and we can access S3 via internet. For accessing our s3 from a private network(inside VPC) without going to public network(internet) we can use VPC endpoint.

Exa. We can access our S3 from VPC private network by using the VPC endpoint. We need to create a VPC endpoint for an S3 bucket and add in route table.

**Note: VPC and S3 should be in same region**.