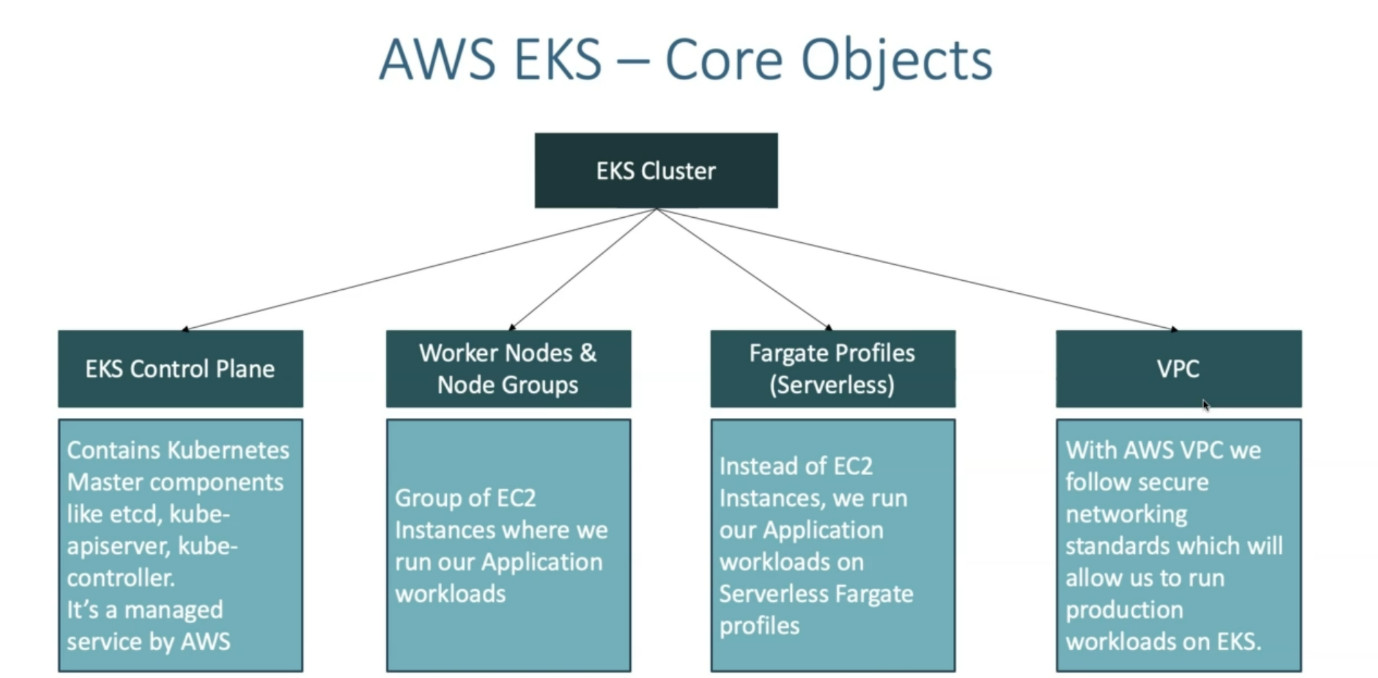
**05: EKS Cluster Introduction**



--- EKS cluster primarily will have 4 core concepts.

**1.EKS control plane**

**2.Worker nodes and node groups**

**3.Fargate profiles**

**4.Vpc**

--- **EKS control plane** – EKS control plane is nothing but master node of the regular architecture. Kubernetes master node is controlled by aws. This is complete managed service by AWS.

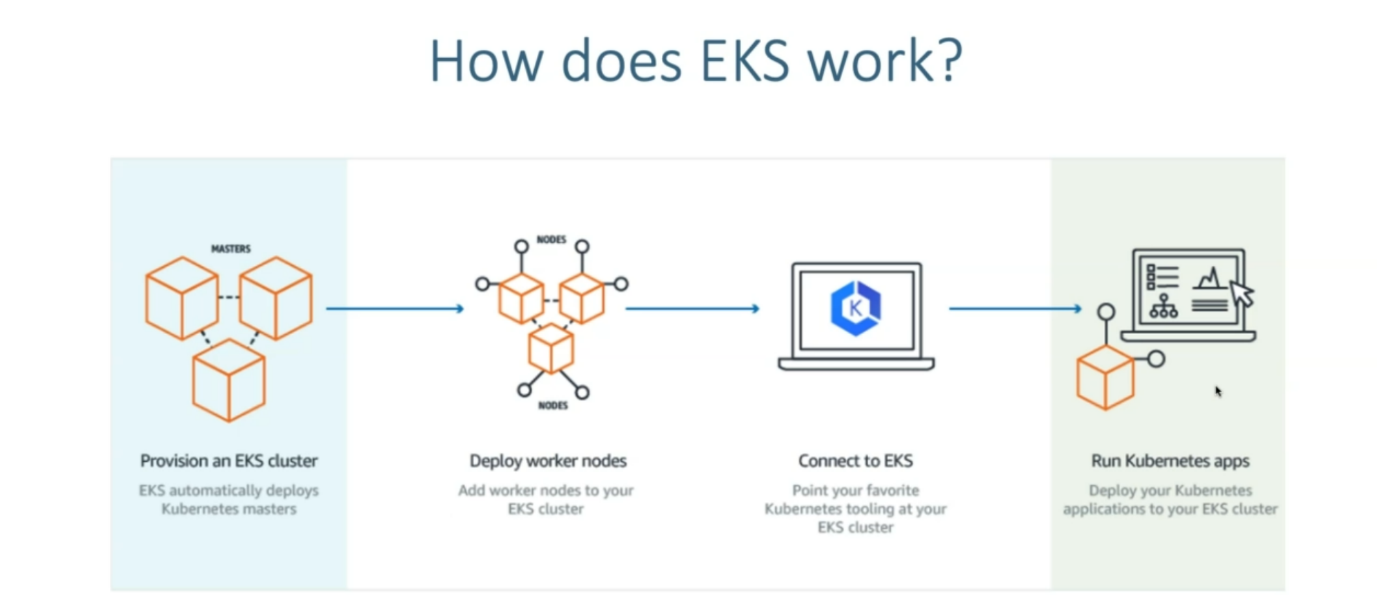
--- **Worker nodes and node groups** - Worker nodes and node groups are nothing but group of EC2 instance. We will provision clusters and run our applications on these clusters.

--- **vpc** – security comes from VPC.

--- **note** - you can deploy your application workloads directly on the public subnet in the EKS cluster, or you can create a node grouping the private subnet and deploy there. You can even deploy fargate profiles.

--- another important thing here is that the fargate profiles can be created only if you’re VPC is having minimum one private subnet, which means forget profiles only run-on private subnets, which means we should be aware of VPC concepts and also, we should be aware of public and private subnet and security groups.

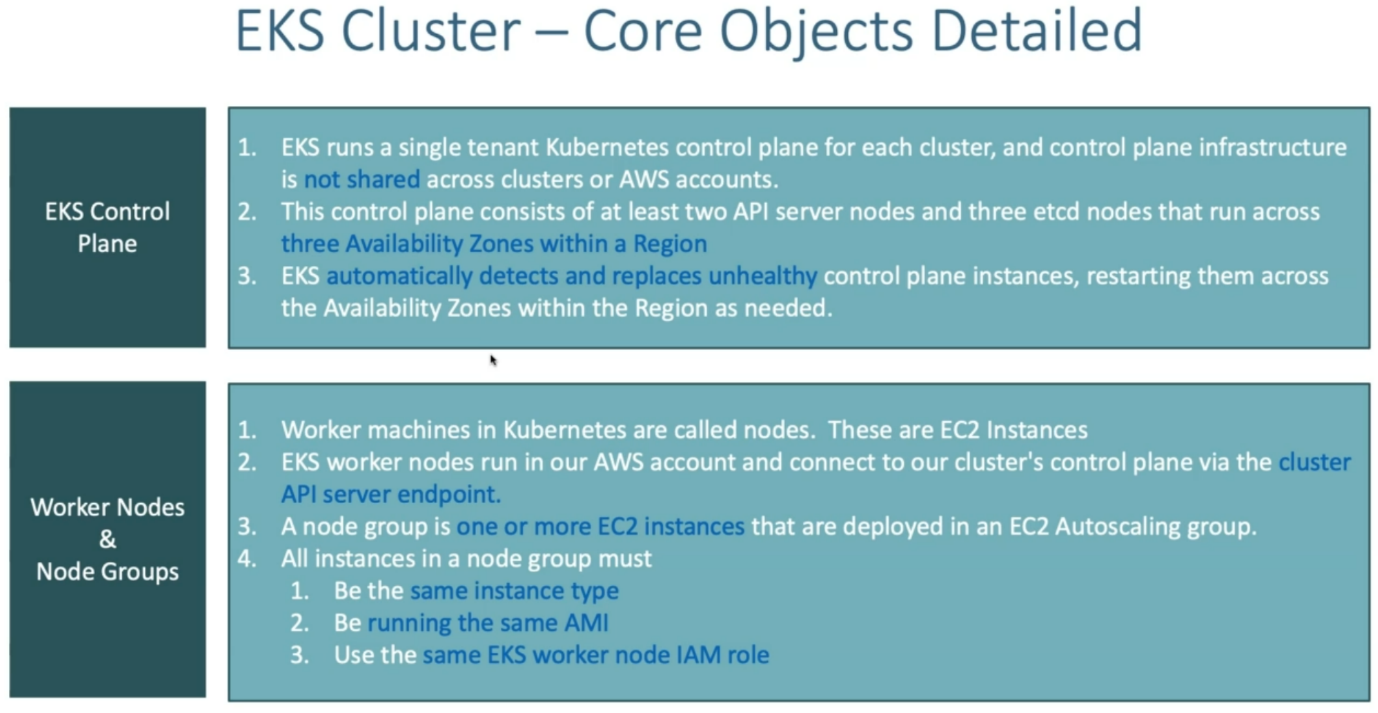
**How does EKS works**

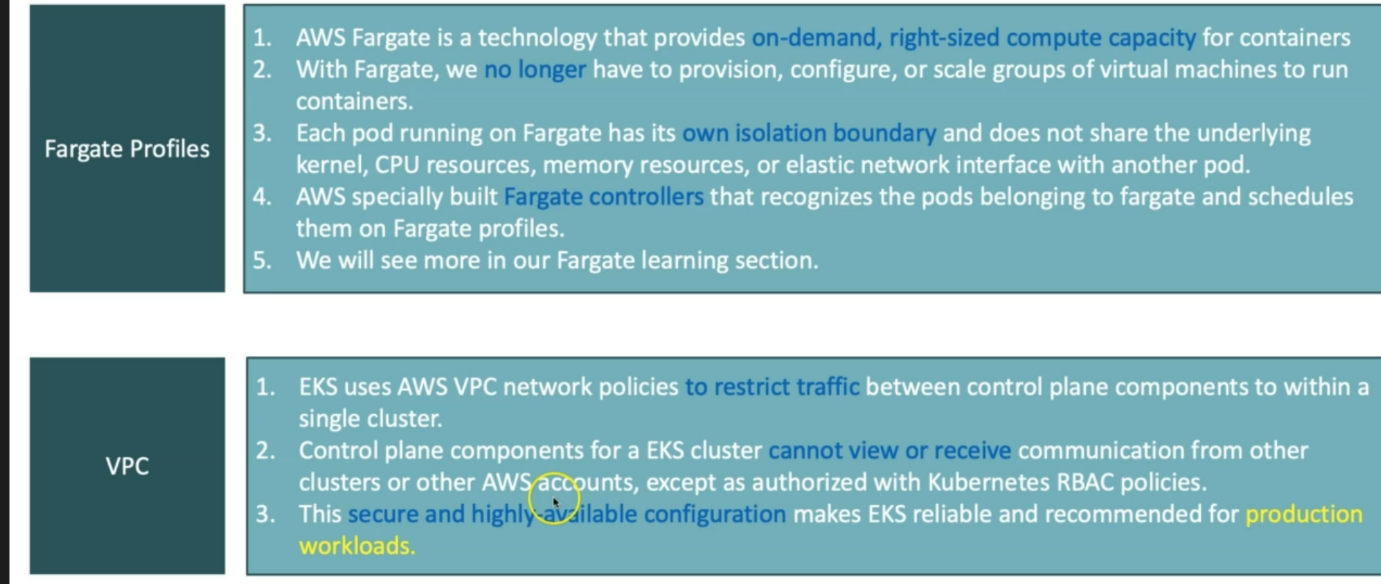


--- So first we are going to provision our EKS cluster and then which is nothing but our control plane and once we provision our EKS cluster, we are going to create the node groups nothing but deploying our worker nodes.

--- what we are going to do now is we are going to deploy our applications. So, to deploy our application workloads, we need to connect to our cluster. So, we are going to do that using our kubectl cli.

**EKS cluster – core objects details**





---