**Interface Endpoints**

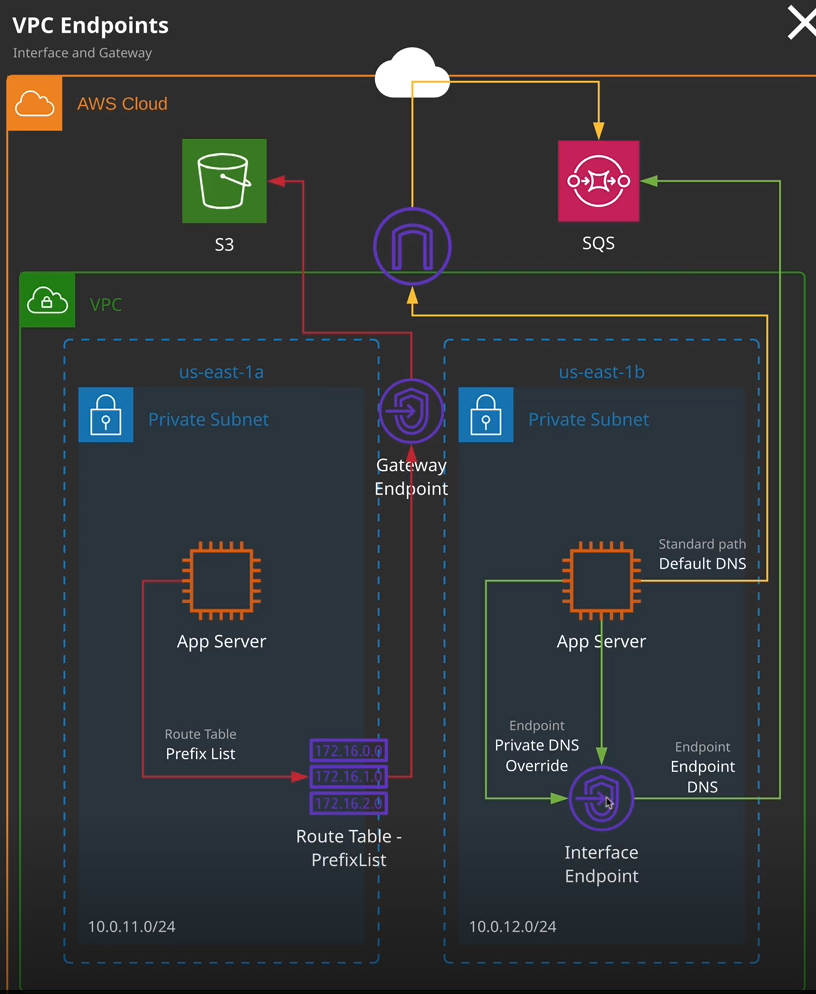
So interface endpoints are created from the same place. If I go to endpoints and I create an endpoint and this time we're going to create one for SNS which is a service that uses interface endpoints, some of your head and slight. This so com.amazonaws.us-east-1.sns which is the region an SNS which is the service. I'll still need to pick the VPC that this goes into so VPC1 and then I'll need to select subnets.

Now **it's important to understand the differences between these interface endpoints and gateway endpoints. For interface endpoints routing is not used. With interface endpoints, what you're actually doing is creating a networking object an ENI that lives in a specific subnet. So where gateway endpoints were highly available across all availability zones inside a specific region. With interface endpoints, you have to explicitly set which subnets it goes into.** So I could deselect this and have it only going into one availability zone. So one subnet in one availability zone. I could also select all of the availability zones inside a region, or at least all of the ones which a VPC is using. So in this case, I've only got two 1a and 1b. You can only create these interface endpoints. You create them once in a specific availability zone and you have to set a subnet. So in this case, I want to set the application a subnet and the application b subnet and these will be accessible from any of the other subnets inside this VPC but by having multiple availability zones, you making sure that it's highly available. There's also an option called private DNS and I want to uncheck this for a second because it's worth understanding exactly how this works, and I want to demonstrate it, so I'm going to untick that for now.

**Unlike gateway endpoints, interface endpoints can be controlled using security groups,** and you're actually associating a specific security group with the interface endpoint. The default is to use the default security group for the VPC, but you can select other security groups and if you select the security group at this point, you need to make sure that it has rules in it allowing any other resources in that VPC to communicate with this specific interface endpoint. So now that I've set all these options I'm going to scroll down to the bottom and hit create endpoint. I'm going to select the VPC endpoint because I want to draw your attention to a number of really important points that you'll need for the exam. **So the way that interface endpoint works is that it actually provides another unique endpoint for that specific service. So what do I mean by that? Well, when you're interacting with the service from the command line, the command line is actually interacting with what are known as service endpoints. So every public space service has its own public endpoint**. So this is the endpoint that's used for example, if you install the AWS command line tools on your local machine or inside any Linux Academy environment, they're communicating with the public endpoints for that specific AWS service. So S3 has one. SNS has one. SQS has one. Kinesis has one. Every public AWS service has a public endpoint in every single region. Now I'll make sure this link is in the lesson description but this is an example of the public endpoints that you can use for the various different AWS services. So if I scroll down to SNS which is the service I'm demonstrating, this is actually the public endpoint that's used for SNS. **So it's sns.us-east-1.amazonaws.com and if I move back to the terminal and I attempt to do a ping of this public endpoint what I'll get is a public IP address, and that's logical. It's a public endpoint.** It has a public IP address.

Now, **when you create an interface endpoint what you also get our endpoint specific DNS names**. So in this particular case I have created an interface endpoint in two availability zones and so I've got two different DNS names. I've got one DNS name for U.S. East 1b and I've got one DNS name for U.S. East 1a. I've also got a DNS name, which is not availability zone specific. So one of these names will always point at the interface in 1a. One of these will always point at the interface in 1b and one of them will automatically point you at the best interface to use in either 1a or 1b. **So by using this specific interface endpoint, you will always have access as long as you've got one interface that's running in any of the availability zones.** So let me demonstrate that if I move back over to my terminal and this time I ping this endpoint this time it's a private IP address that's inside this VPC. **So if you had any applications running inside a VPC and you wanted them to be able to access these public services without using an internet gateway or a NAT gateway, then you could access them using this new DNS name but that would require adjusting the code of the application. Interface endpoints come with a really cool feature though, if I go to actions and then I go to modify private DNS names and I enable private DNS name and click on modify what that does is it replaces or overrides the default public endpoint DNS so remember, if I go back over to my terminal this first address that I pinged, which is sns.us-east-1.amazonaws.com. This originally pointed at a public IP address. By enabling private DNS on the interface endpoint, that means that this is now overridden. Now, after this private DNS option has taken effect now we're pinging this public endpoint and we're getting this private IP addresses.** So after a couple of minutes, the change is effective and now we could access SNS without requiring any code changes.

**So interface endpoints are controlled via security groups. NACLs can also impact traffics.** Remember, network ACLs apply to a subnet and if you got any situations where any EC2 instances are accessing the endpoints from other subnets, remember these interface endpoints occupy a specific subnet, NACLs only apply for traffic crossing that subnet. So NACLs can impact the traffic if it's in a different subnet. **Interface endpoints and will replace the DNS for the service that you're attempting to use**. **It will need code changes unless you enable private DNS.**



So this is essentially the architecture that's used it for using gateway endpoints issues in this prefix list the communication path then goes from the instance it uses these route table prefix lifts. It goes to the gateway and point and then it hops through to the public endpoint running inside AWS without needing to go via public addressing. Using an interface endpoint the process is somewhat similar. You resolving a public endpoint DNS or a private endpoint DNS that directs at this interface endpoint and then that directs out using private addressing through to these public endpoints. So the purpose of both the gateway endpoint and the interface endpoint is to essentially allow you to connect to public services without needing a public gateway or public IP address. Okay, that's everything I wanted to cover in this lesson. For the solutions architect associate exam really, all you need to be aware of is exactly how these interface endpoints function and exactly when you'd use them. **So you'd use interface endpoints when you're trying to provide access to public AWS services without making the instance or the entire VPC publicly accessible. So maybe you're running an application, which has really stringent security requirements. You're not able to make those instances have a NAT gateway or be otherwise publicly accessible but you need to access S3 maybe for software updates or to access customer data and by using these gateway endpoints you're provided with that functionality.** Please do remember, for the exam **that S3 and DynamoDB use gateway endpoints which use prefix lists. Every other service uses interface end points which make use of DNS.**