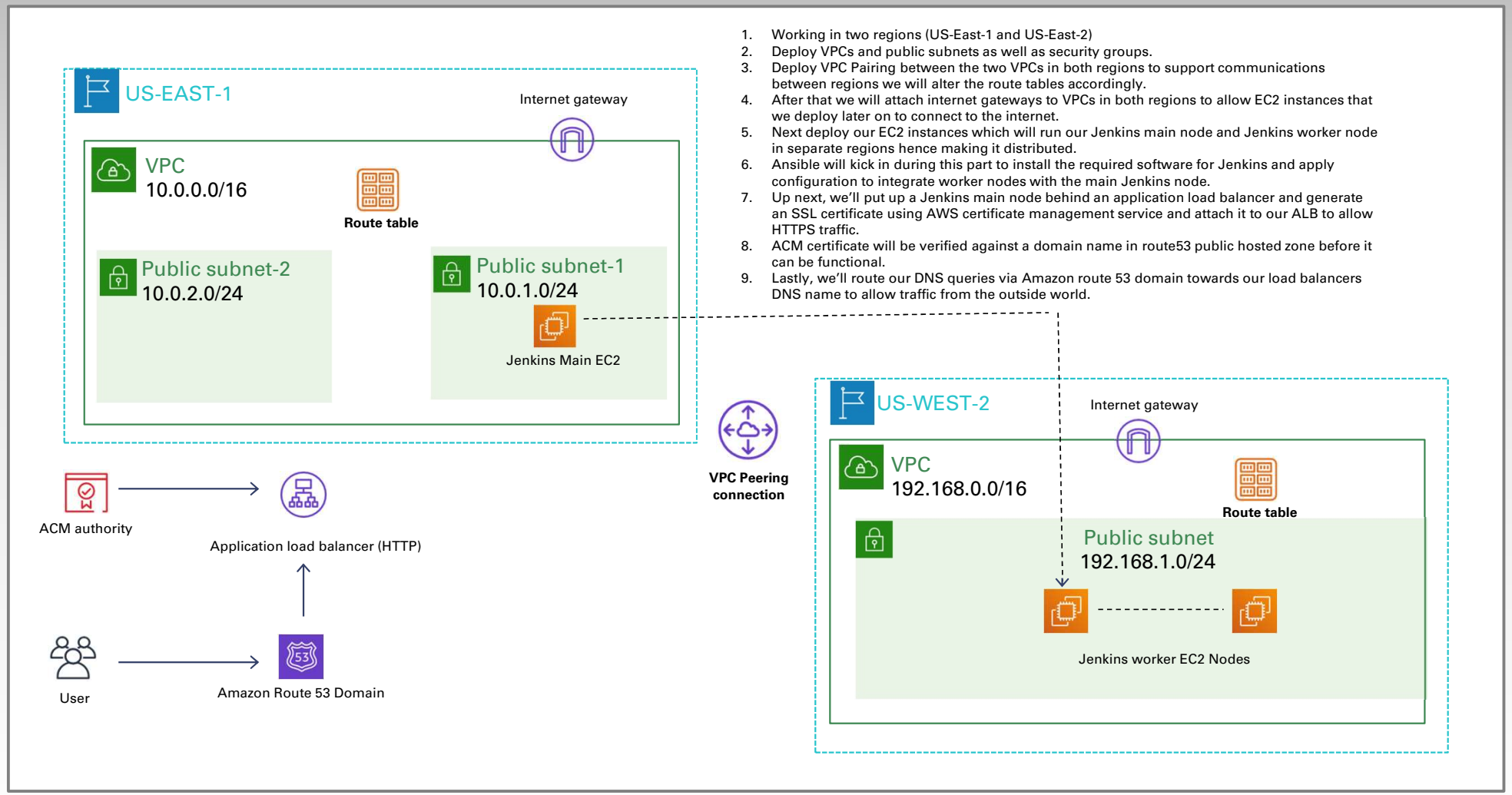
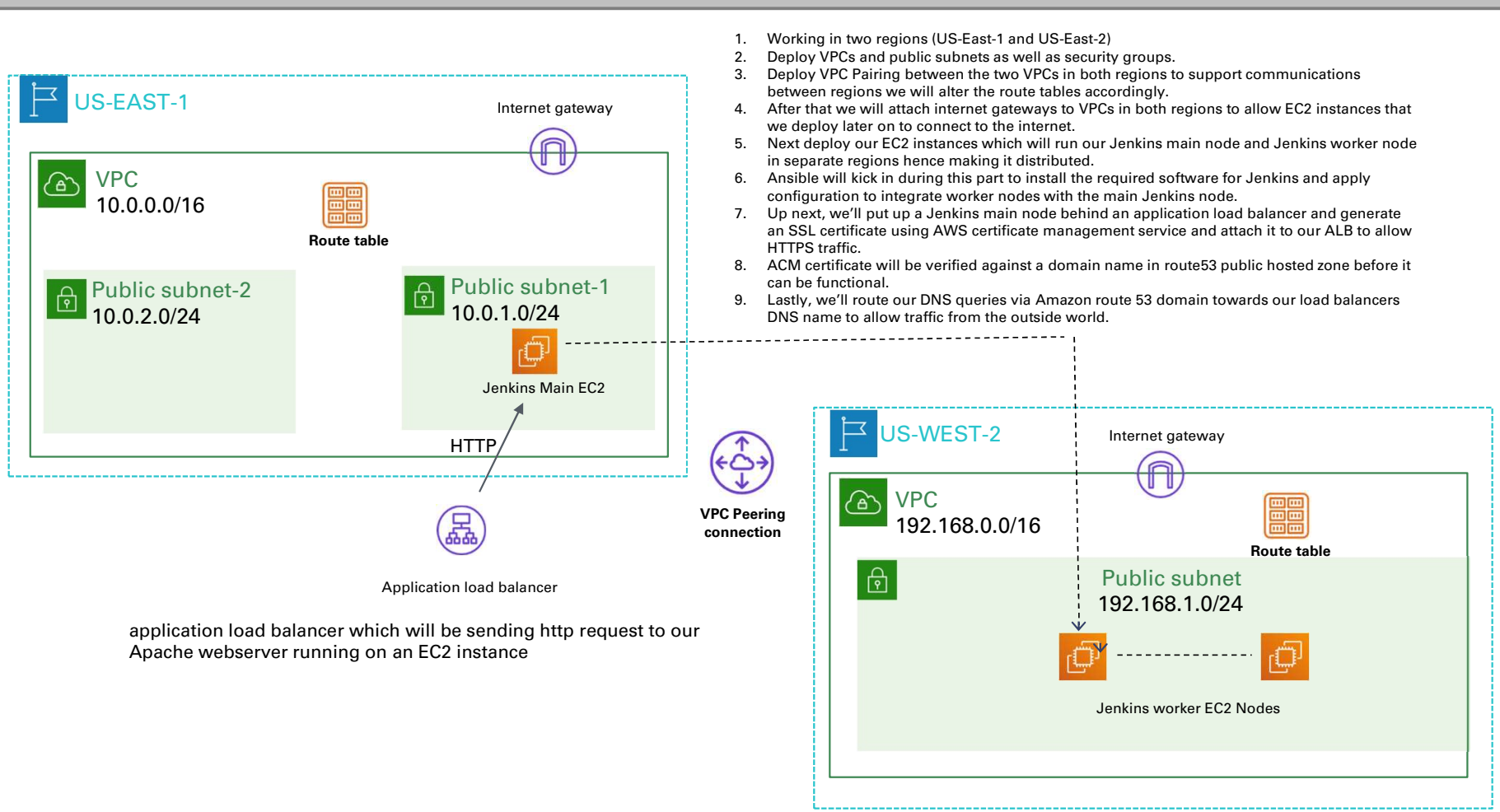

1. Working in two regions (US-East-1 and US-East-2)
2. Deploy VPCs and public subnets as well as security groups.
3. Deploy VPC Pairing between the two VPCs in both regions to support communications between regions we will alter the route tables accordingly.
4. After that we will attach internet gateways to VPCs in both regions to allow EC2 instances that we deploy later on to connect to the internet.
5. Next deploy our EC2 instances which will run our Jenkins main node and Jenkins worker node in separate regions hence making it distributed.
6. Ansible will kick in during this part to install the required software for Jenkins and apply configuration to integrate worker nodes with the main Jenkins node.
7. Up next, we'll put up a Jenkins main node behind an application load balancer and generate an SSL certificate using AWS certificate management service and attach it to our ALB to allow HTTPS traffic.
8. ACM certificate will be verified against a domain name in route53 public hosted zone before it can be functional.
9. Lastly, we'll route our DNS queries via Amazon route 53 domain towards our load balancers DNS name to allow traffic from the outside world.
US-EAST-1
Internet gateway
VPC
10.0.0.0/16
Route table
Public subnet-2
10.0.2.0/24
Public subnet-1
10.0.1.0/24
Jenkins Main EC2
VPC Peering connection
US-WEST-2
Internet gateway
VPC
192.168.0.0/16
Route table
Public subnet
192.168.1.0/24
Jenkins worker EC2 Nodes
ACM authority
Application load balancer (HTTP)
User
Amazon Route 53 Domain



1. Working in two regions (US-East-1 and US-East-2)
2. Deploy VPCs and public subnets as well as security groups.
3. Deploy VPC Pairing between the two VPCs in both regions to support communications between regions we will alter the route tables accordingly.
4. After that we will attach internet gateways to VPCs in both regions to allow EC2 instances that we deploy later on to connect to the internet.
5. Next deploy our EC2 instances which will run our Jenkins main node and Jenkins worker node in separate regions hence making it distributed.
6. Ansible will kick in during this part to install the required software for Jenkins and apply configuration to integrate worker nodes with the main Jenkins node.
7. Up next, we'll put up a Jenkins main node behind an application load balancer and generate an SSL certificate using AWS certificate management service and attach it to our ALB to allow HTTPS traffic.
8. ACM certificate will be verified against a domain name in route53 public hosted zone before it can be functional.
9. Lastly, we'll route our DNS queries via Amazon route 53 domain towards our load balancers DNS name to allow traffic from the outside world.

application load balancer which will be sending http request to our Apache webserver running on an EC2 instance