AWS Networking 101

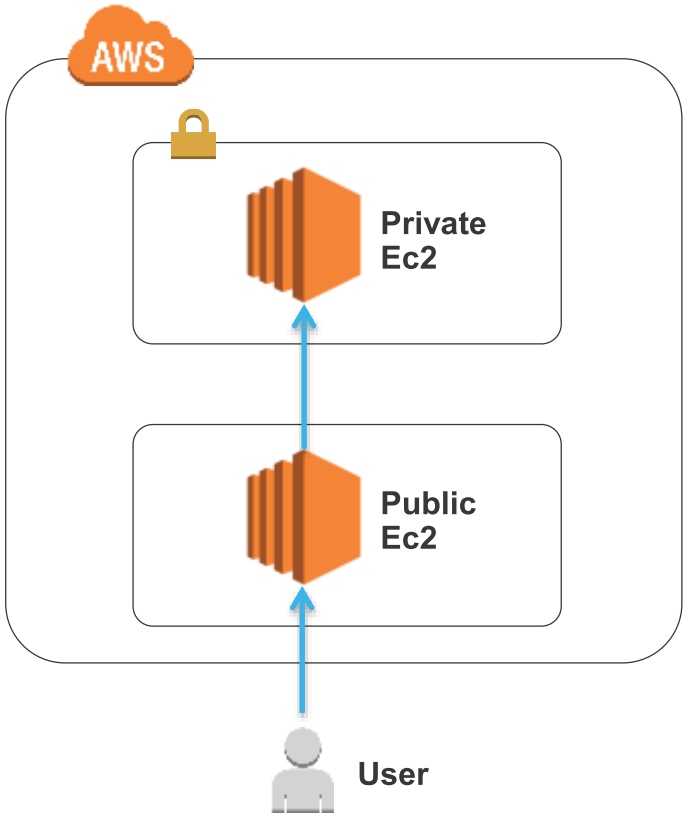
Stenio Ferreira - CrossMarket

June 2017

slalom

Basic Architecture – Private/ Public instances

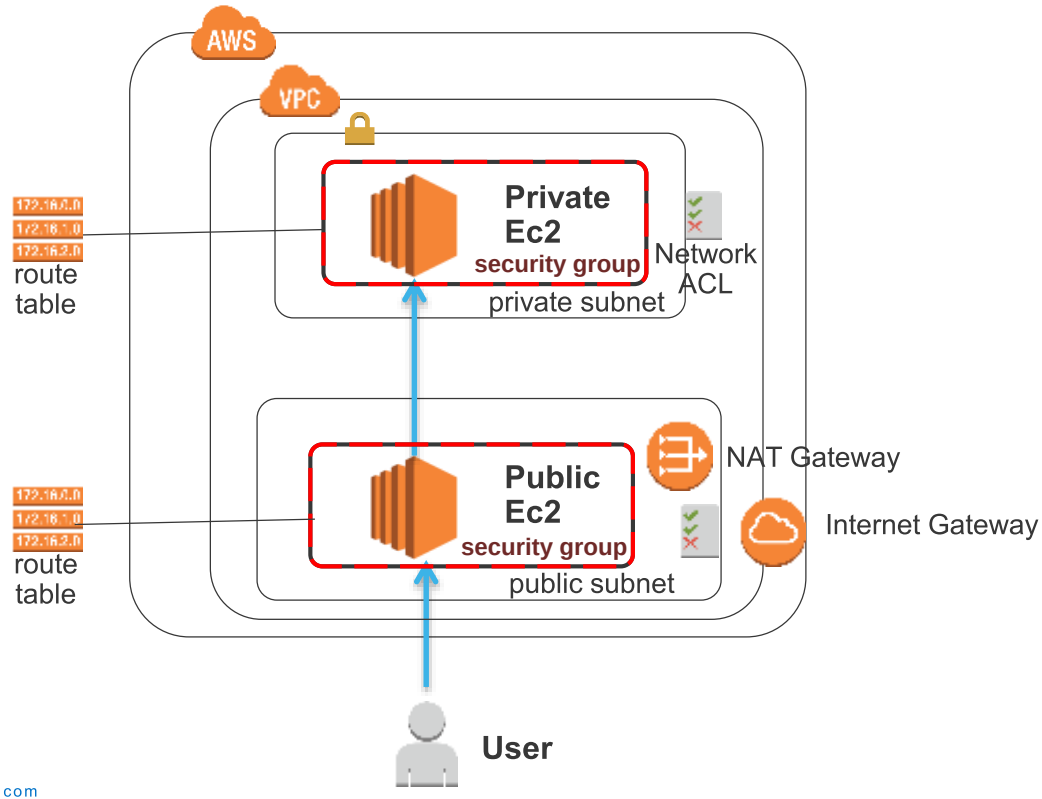
Simple, right?



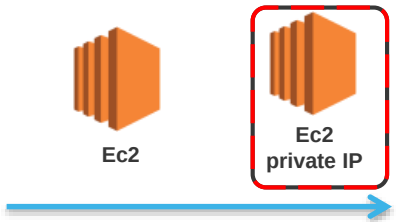
- Private EC2 cannot be accessed directly from outside
- Public EC2 acts as a “bridge”
- User is anything making requests from outside

Detailed Architecture - Private/ Public instances

Well, not quite!

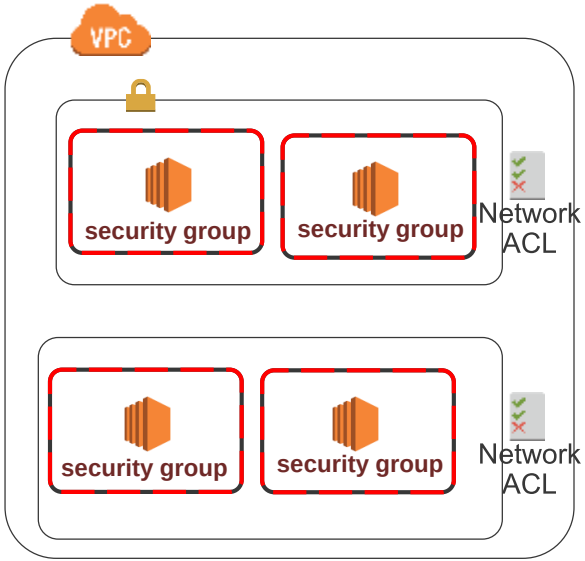


- Security Groups
- Network ACL
- Subnets
  - Private vs Public
- VPC
- Route Tables



Security – Security Groups/ Network ACL

Deny or allow traffic from/to specific sources



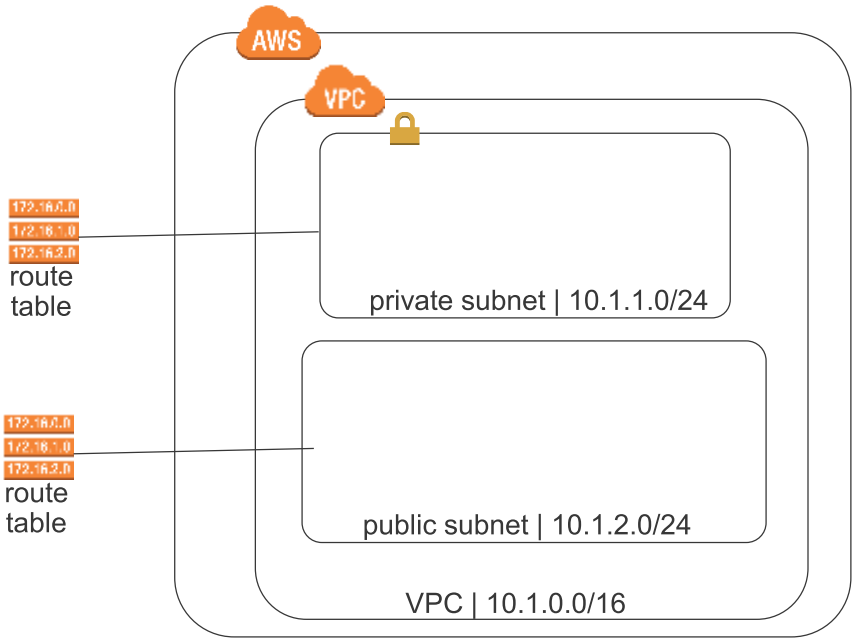
- **Security Groups**
  - Filters traffic entering and exiting an instance
  - Stateful filtering (inbound/outbound automatic)
  - Default deny all traffic

- **Network ACL**
  - Filters traffic entering and exiting a subnet
  - Stateless filtering (explicit inbound/outbound rules)
  - Network ACLs does not filter traffic between instances in the same subnet
  - Default allow all traffic

Filter by:
Protocol, Port Range, Source (CIDR, Security Group)

Subnets

Segmentation within a VPC CIDR block



Each subnet has:

- **CIDR Range**
  (e.g. 10.1.0.0/16)

  /8 255.0.0.0
  /16 255.255.0.0
  /24 255.255.255.0
  /32 255.255.255.255
  0.0.0.0/0 -> catch all

- **Route Tables**
  Directs traffic

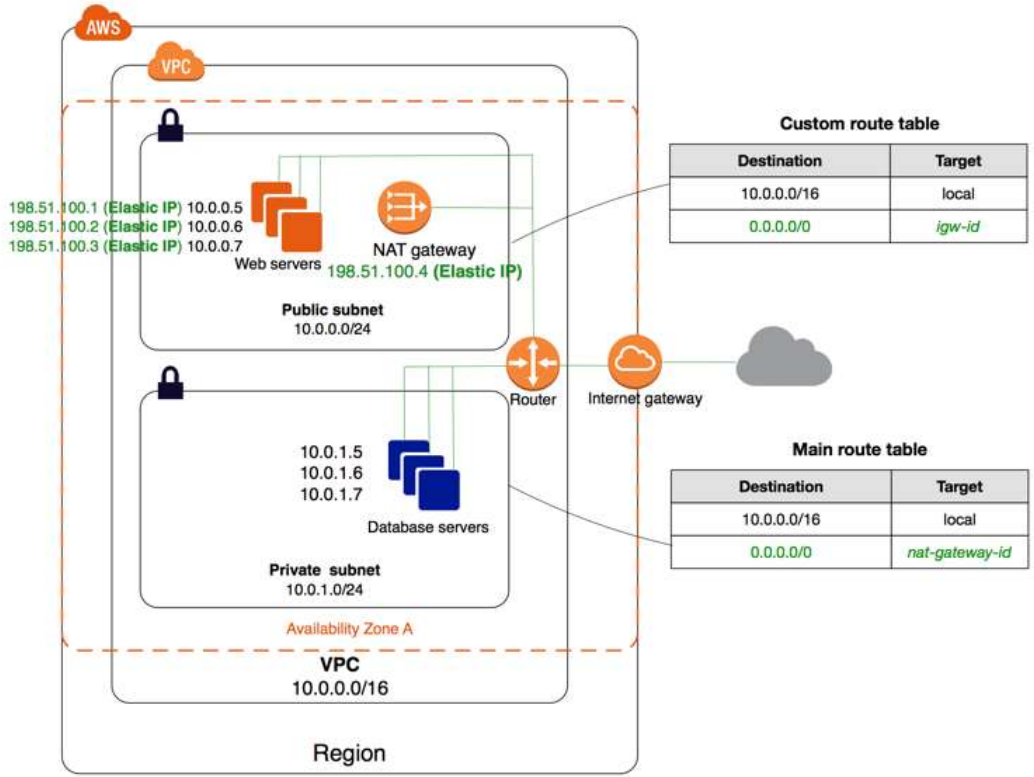
  Destination -> Target
  e.g.
  0.0.0.0/0 -> igw-123
  10.2.0.0/16 -> pcx-123
  10.51.0.0/16 -> eni-123

- **Network ACL**

And is in a single A-Z.

Public vs Private Subnets

Traditional Web Client/ API Server/ Database Server example



Public Subnets:

- Attach an Internet gateway to your VPC.
- Subnet's route table points to the Internet gateway
- Instances in subnet have a globally unique IP address
- Network ACL and security group rules allow the relevant traffic to flow to and from instance

Private Subnets:

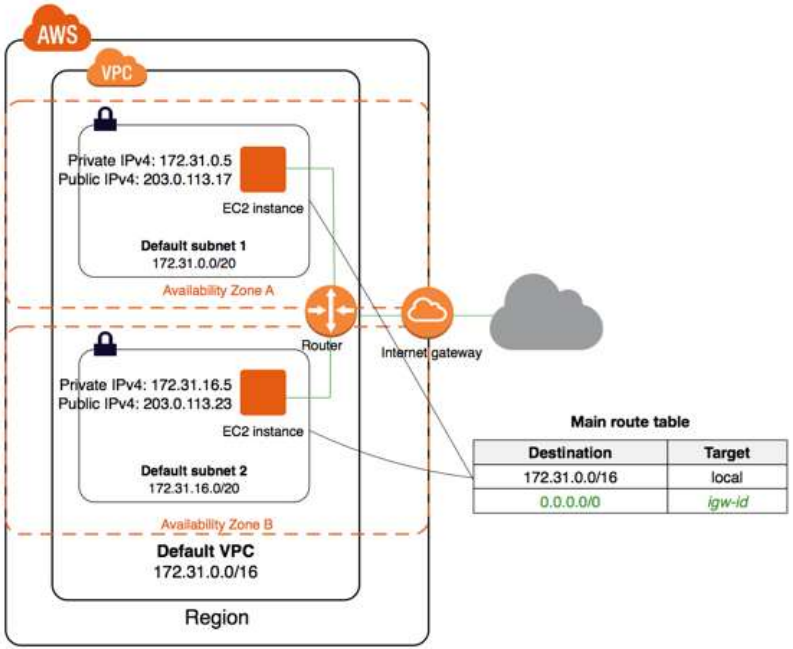
NAT Gateway allows one-way traffic for instances in the private subnet – they can access the internet, but not the other way around.

NAT Gateway

- Located in the public subnet
- Needs elastic ip
- Update route in private subnet to send relevant traffic to NAT Gateway

VPCs

Logically isolated section of AWS where resources can be launched

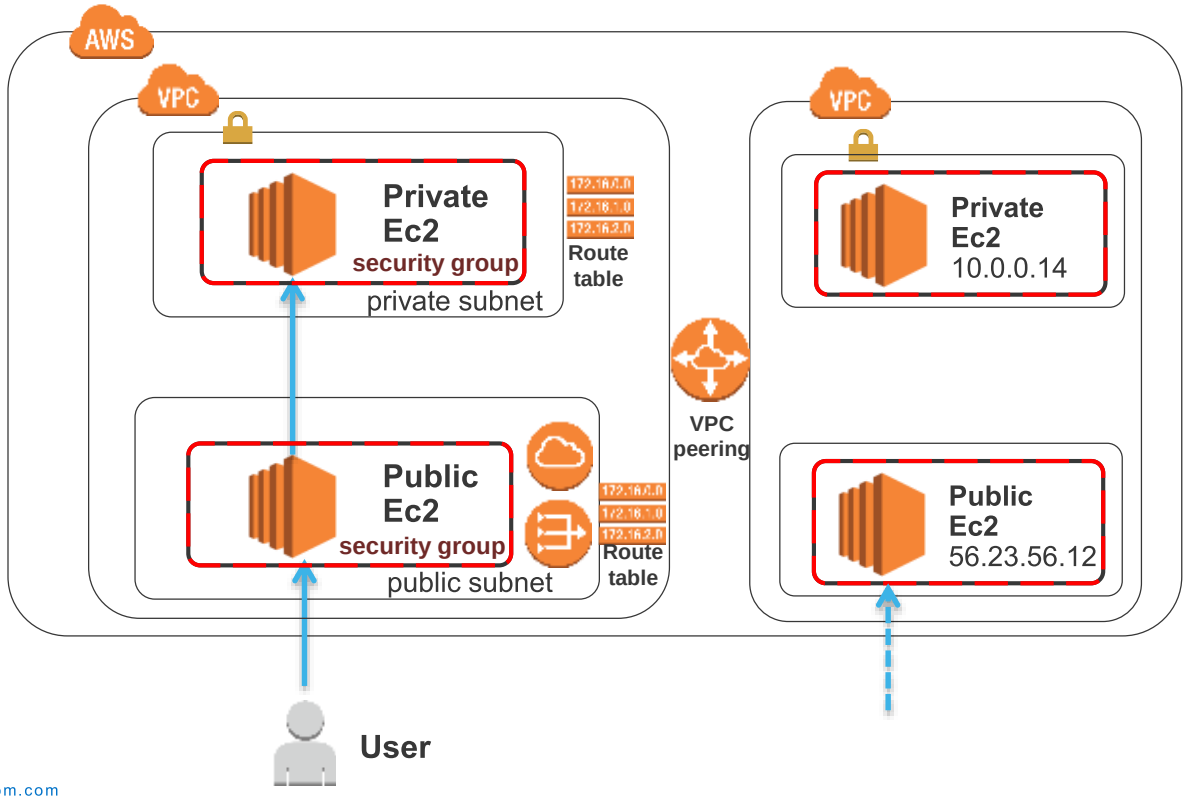


Default VPC:

- Create a VPC with a size /16 IPv4 CIDR block.
- Create a default subnet in each Availability Zone.
- Create an Internet gateway and connect it to your default VPC.
- Create a main route table for your default VPC with a rule that sends all IPv4 traffic destined for the Internet to the Internet gateway.
- Create a default security group and associate it with your default VPC.
- Create a default network access control list (ACL) and associate it with your default VPC.
- Associate the default DHCP options set for your AWS account with your default VPC.

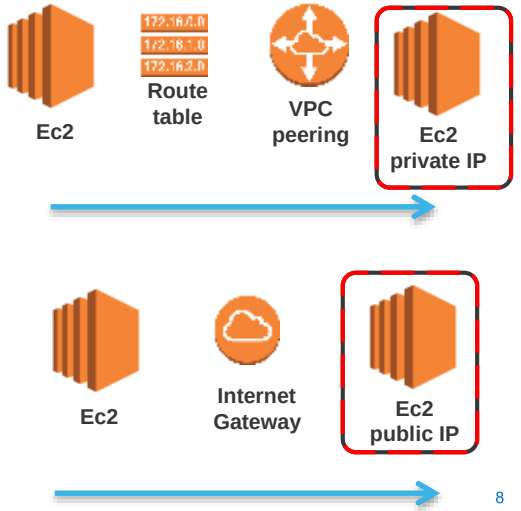
Multiple VPCs

VPC peering is possible only in same AWS Region



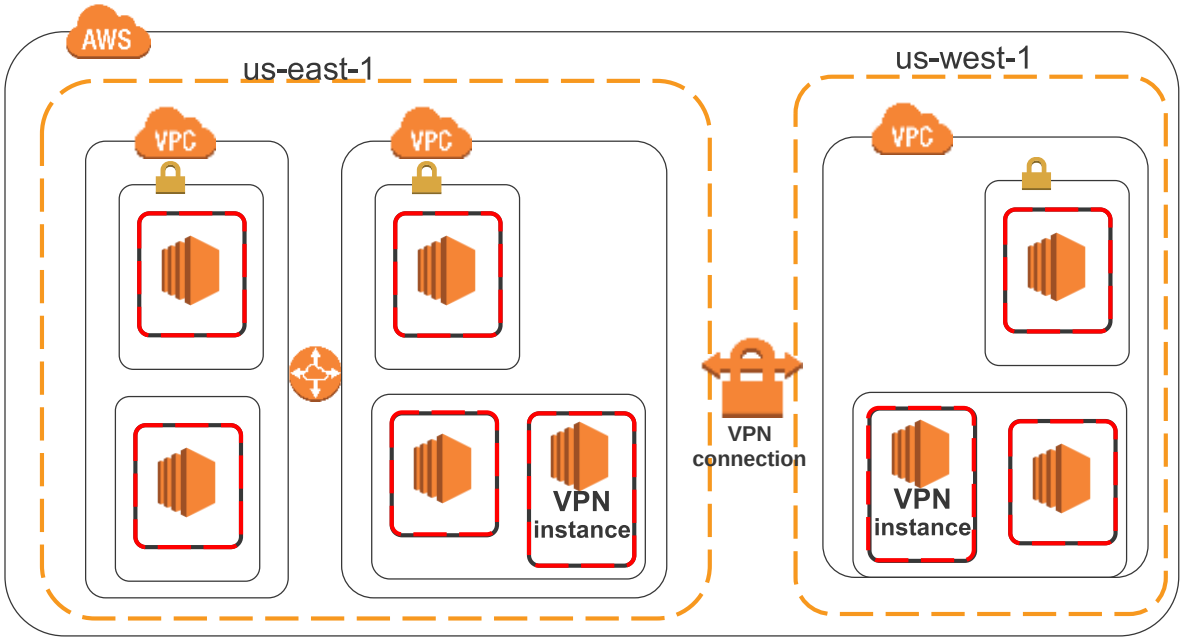
Route Table configuration

- VPC Peering for private IPs
- Internet Gateway/
  Nat Gateway for public Ips



Multiple AWS Regions

Traditional Web Client/ API Server/ Database Server example

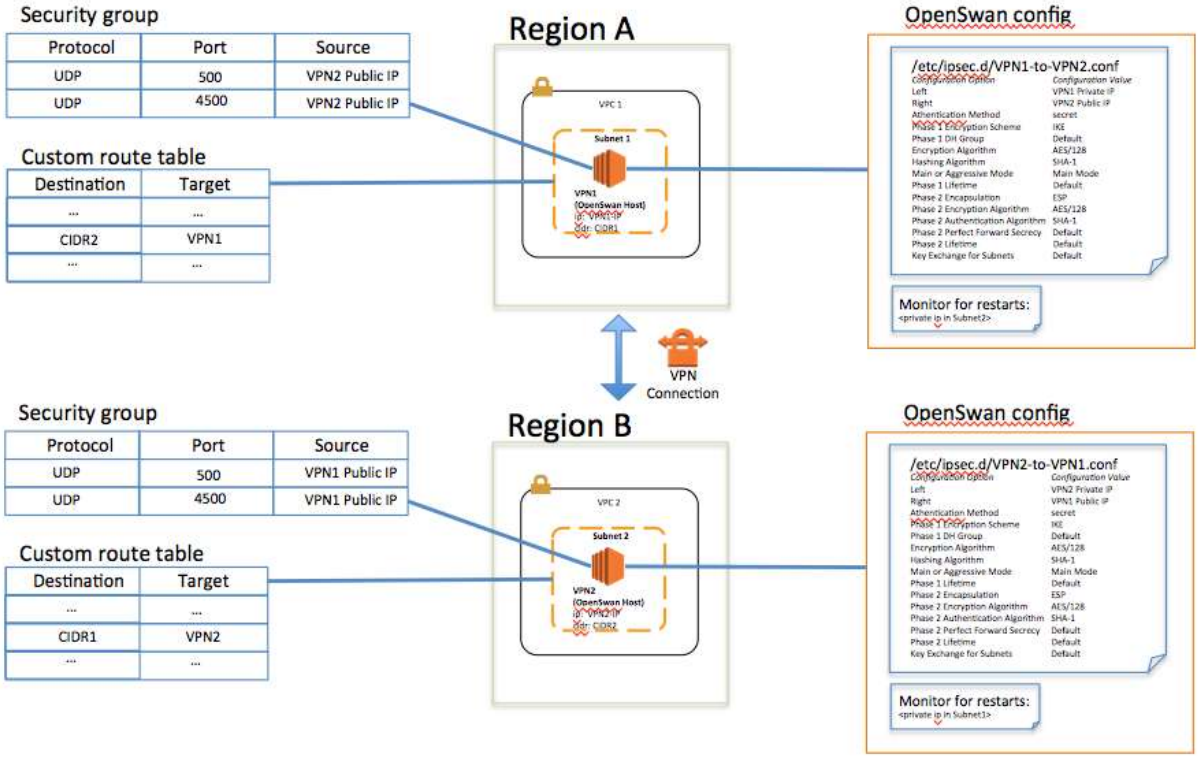


User

- VPN Tunnels
- Route Tables

IPSec VPN

IPSec is a Protocol suite for authenticating and encrypting data transfer across a network

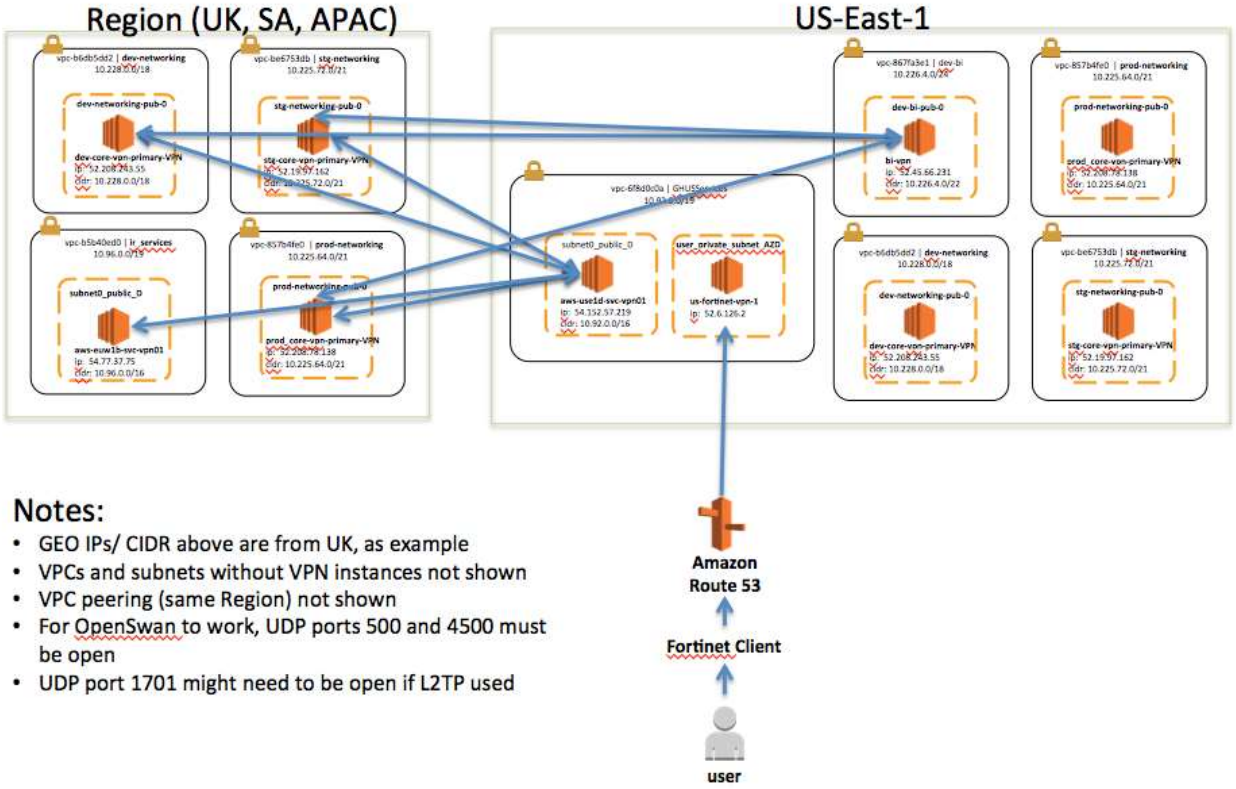


IPSect Stages:

1. **Interesting traffic initiates the IPSec process**
2. **IKE phase one**: authenticates, sets secure channel for phase two.
3. **IKE phase two**: IKE negotiates IPSec SA parameters and sets up matching IPSec SAs in the peers.
4. **Data transfer**
5. **IPSec tunnel termination**: through deletion or by timing out

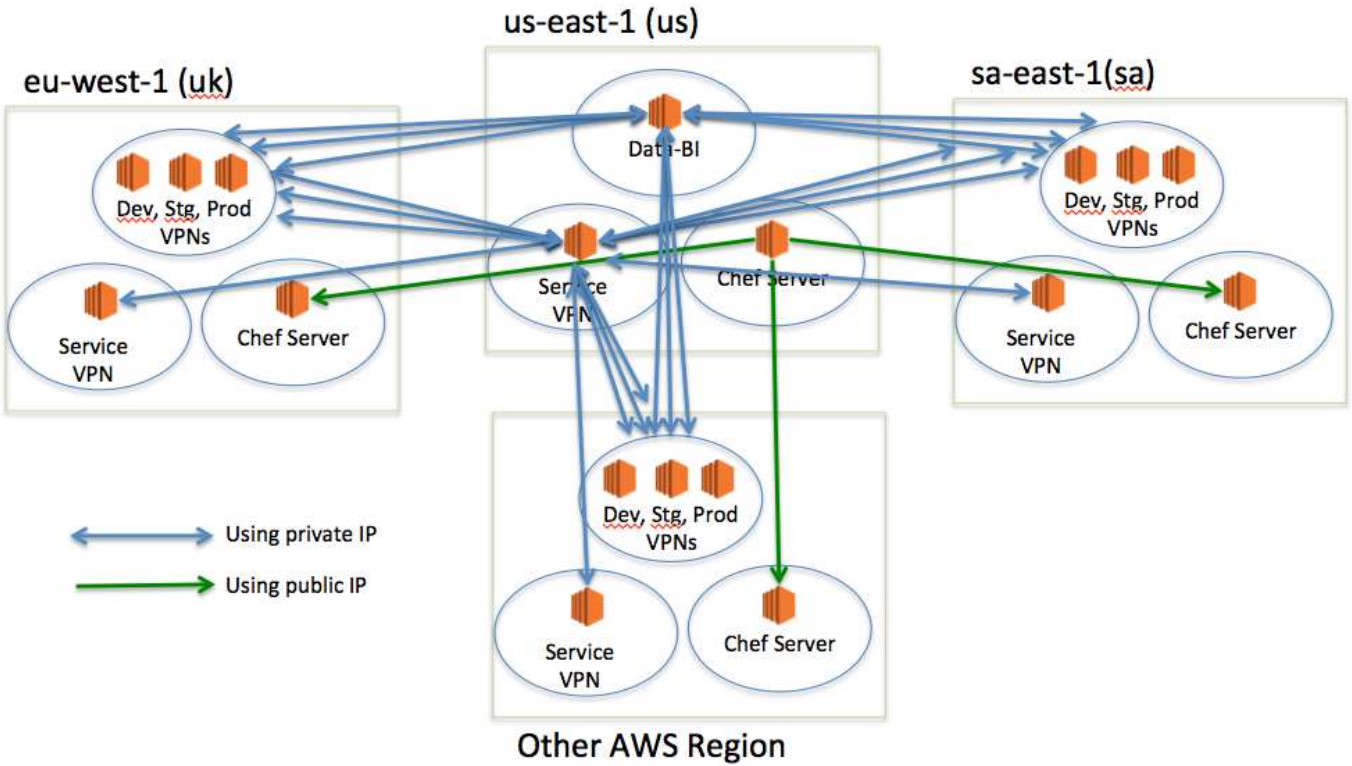
Example Architecture 1/2

Outside access is allowed by Fortinet Gateway



Example Architecture – 2/2

Chef Server hub-spoke




Thank you!
Stenio.ferreira@slalom.com

Note on separating environments/departments

- Multiple AWS Accounts
- Multiple IAM roles same account
- Hybrid: production/non-production

IPSec Commands

# Bring a tunnel up or down:
$ ipsec auto --down <tunnel_name>
# Check status:
# If you do not see "erouted", it signifies that there might be an issue
# Anything 4500 is success, 500 might indicate a failure in one of the sides

$ ip xfrm state

$ ip xfrm monitor

TCPDump Commands

TCPDump can also be used to evaluate traffic received

Shows traffic received in port 22
$ tcpdump -i eth0 port 22
Shows traffic originating from specified IP
$ tcpdump -i eth0 src 192.168.0.2

On the VPN EC2 instances, ensure that Source/ Dest. check is **false**.