VPC

Virtual Private Cloud

Dr Peadar Grant

Cloud Architecture

Goals

You will be able to:

1. Explain a VPC as a container for Virtual Machines
2. Design IP address schemes for simple VPCs
3. Construct basic VPC using CLI and console
4. Test your setup using a PowerShell script
5. Create a script to set up VPCs

Agenda

1. VPC
2. Addressing
3. Routing
4. Sample VPCs
5. Demo - simple example
6. This week's lab

VPC

AWS documentation states that:

> *"Virtual Private Cloud (VPC) lets* you provision *a* logically isolated *section of the AWS Cloud where you can launch AWS resources in a* virtual network *that you define. You have complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways."*

Virtual machines exist in a software-defined network you create.

Key VPC components

Virtual Private Cloud your virtual network in AWS

  Subnet within VPC in AZ

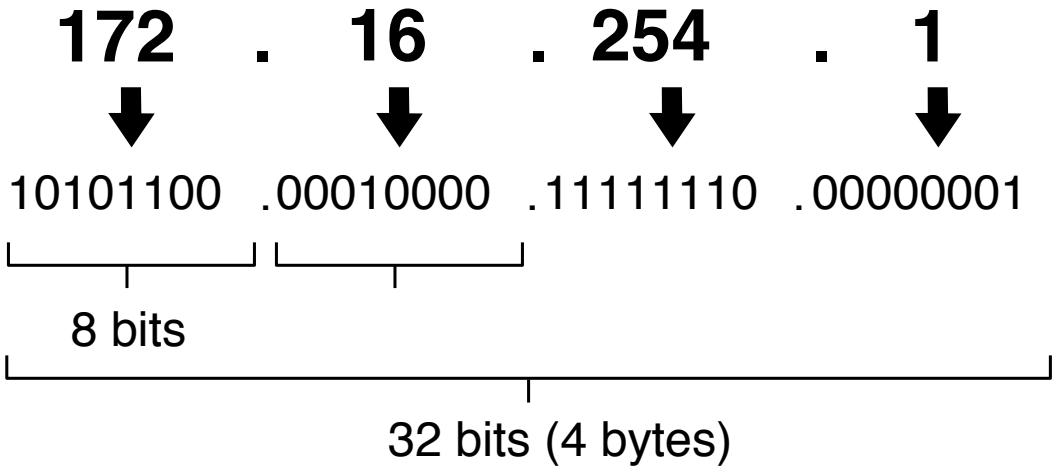
EC2 instances virtual machines

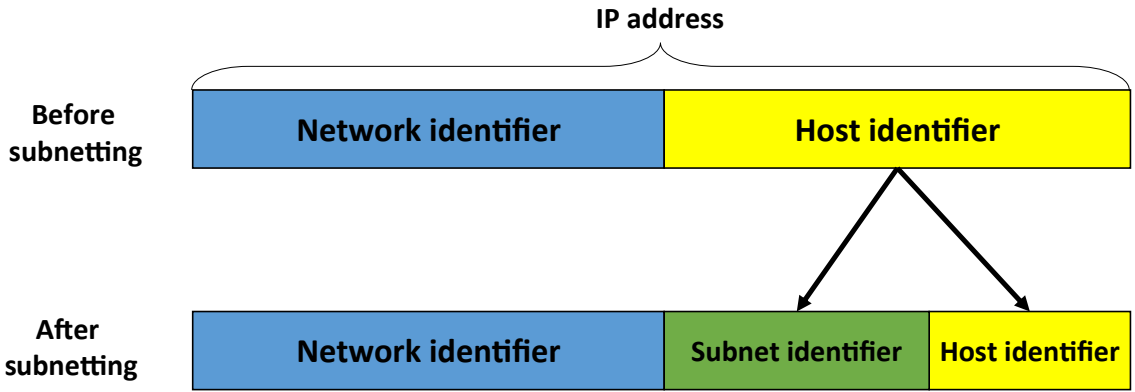
Network ACL inter-subnet firewall

Route tables routing in/outside VPC

Internet gateway connects VPC to internet

IPv4 address in dotted-decimal notation




IP address
Before subnetting
Network identifier
Host identifier
After subnetting
Network identifier
Subnet identifier
Host identifier

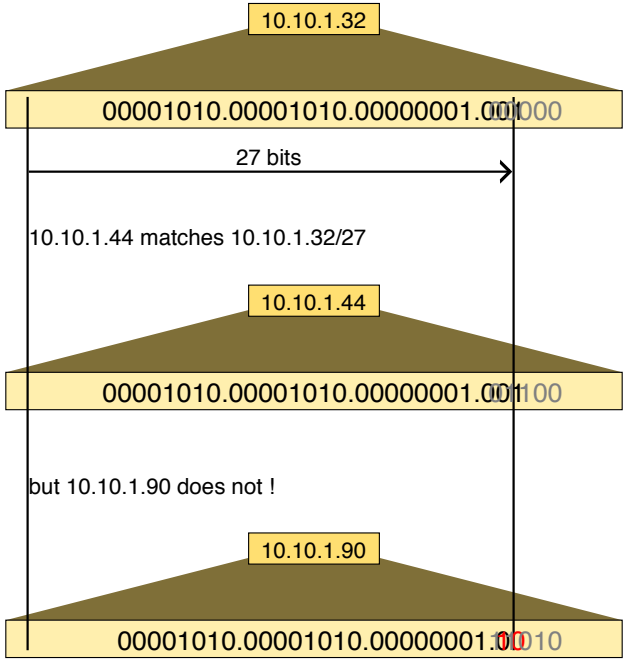
Classless Inter-Domain Routing (CIDR)

Address ranges with (sub)network specified like

$$10.0.0.0/16$$

meaning that:

- Network identifier is 16 bits long ($/16$).
- Host identifier $32 - 16 = 16$ bits long.
- Addresses where first 16 bits different are on different networks.


10.10.1.32
00001010.00001010.00000001.001000
27 bits
10.10.1.44 matches 10.10.1.32/27
10.10.1.44
00001010.00001010.00000001.001100
but 10.10.1.90 does not !
10.10.1.90
00001010.00001010.00000001.001010

Private address ranges (RFC1918)

Private address ranges used freely *inside* private networks.

Block	CIDR block	Range
24-bit	10.0.0.0/8	10.0.0.0-10.255.255.255
20-bit	172.16.0.0/12	172.16.0.0-172.31.255.255
16-bit	192.168.0.0/16	192.168.255.255

- VPCs almost always will use these ranges.
- **Keep it simple:** 10.0.0.0/16 or 192.168.0.0/16

Subnets

VPC is broken up into Subnets:

- VPC has a CIDR block
- Each subnet:
  - Has CIDR block $\leq$ VPC CIDR block
  - Is associated with single AZ
- **Keep it simple** with these patterns:
  - VPC 10.0.0.0/16 with each subnet 10.0.?.0/24
  - VPC 192.168.0.0/16 with each subnet 192.168.?.0/24
  - The /20 address ranges can be confusing!
- Each subnet has some addresses reserved by AWS.
- AWS can auto-assign a public IP to instances in subnet.

Internet gateway

Internet gateway connects VPC to internet:

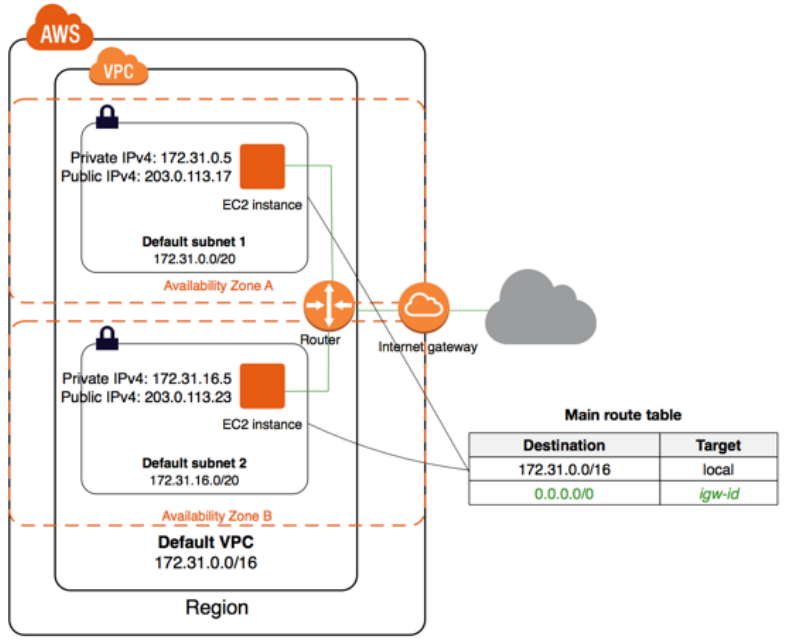
- *Normally* needed for connectivity to VPC
- No configuration required.
- Created *and attached* to VPC.

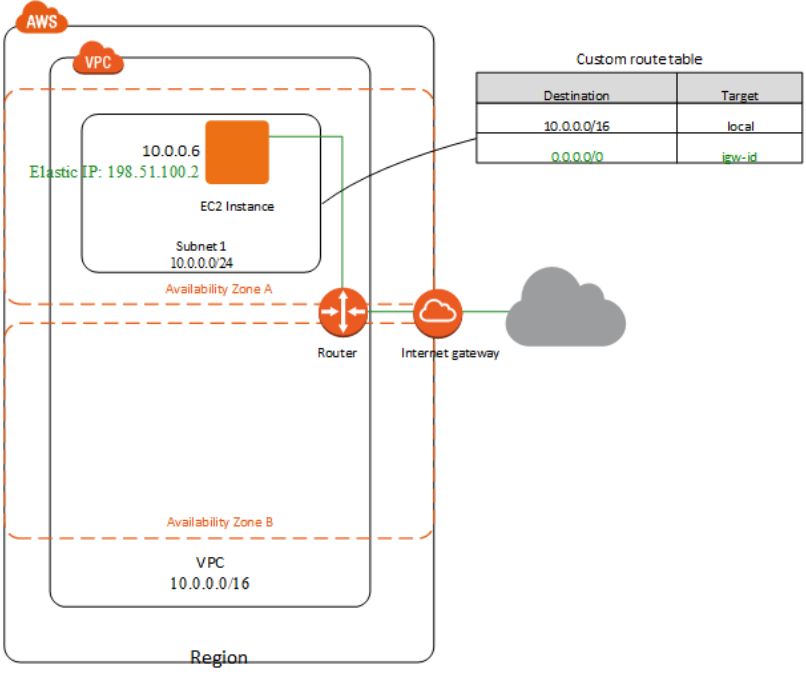
Routing must send traffic to internet gateway

Route table

Sample route table for 10.0.0.0/16 VPC:

Destination	CIDR format	Action
Within VPC	10.0.0.0/16	Local
On internet	0.0.0.0/0	Internet Gateway


AWS
VPC
Private IPv4: 172.31.0.5
Public IPv4: 203.0.113.17
EC2 instance
Default subnet 1
172.31.0.0/20
Availability Zone A
Router
Internet gateway
Private IPv4: 172.31.16.5
Public IPv4: 203.0.113.23
EC2 instance
Default subnet 2
172.31.16.0/20
Availability Zone B
Default VPC
172.31.0.0/16
Region
Main route table
Destination
Target
172.31.0.0/16
local
0.0.0.0/0
igw-id


AWS
VPC
10.0.0.6
Elastic IP: 198.51.100.2
EC2 Instance
Subnet 1
10.0.0.0/24
Availability Zone A
Router
Internet gateway
Availability Zone B
VPC
10.0.0.0/16
Region
Custom route table
Destination
Target
10.0.0.0/16
local
0.0.0.0/0
igw-id

Demo - simple example

- VPC 10.0.0.0/16
- Single subnet of entire VPC 10.0.0.0/16
- Internet gateway
- Route table

This week's lab

Setting up a VPC using the AWS Console and the AWS CLI.