

Project 2

# Networking project



Jason Zhang

Demo Video: <https://broadcast.amazon.com/videos/1176703>

---

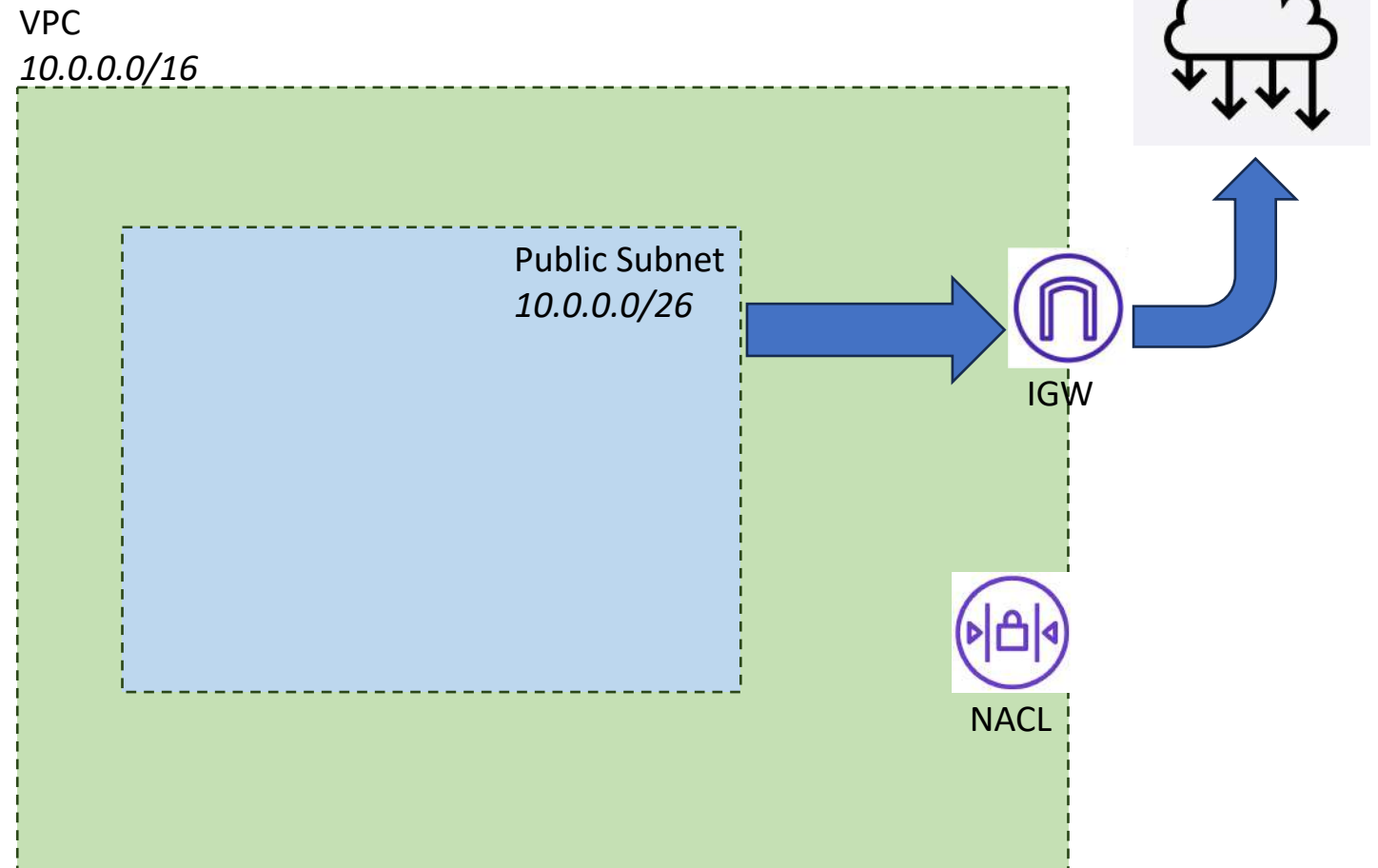
# VPC –Set up

## Route Table:

Destination	Target
0.0.0.0/0	Igw-<igwID>
<VPC IPv4 CIDR>	local

## Default NACL:

Allow **ALL** traffic to flow **in and out** of the subnets with which it is associated



# VPC



VPC

$x.x.x.x/16 \sim /28$



- Max 5 CIDR / VPC
  - Min CIDR / 28 (16)
  - Max CIDR /16 (65536)
- Do not overlap with other VPC while peering
- Private IP only allow
  - **10.0.0.0 – 10.255.255.255**
  - **172.16.0.0 – 172.31.255.255**
  - **192.168.0.0 – 192.168.255.255**
- 5 reserves IPs (F4L1)
  - 10.0.0.0
  - 10.0.0.1
  - 10.0.0.2
  - 10.0.0.3
  - 10.0.0.255

**Example:** if need 29 IP address,  $/27 = 32$  IPs,  $32 - 5 = 27 < 29$ , thus subnet mask should be  $/26$

# Web Server – Set up

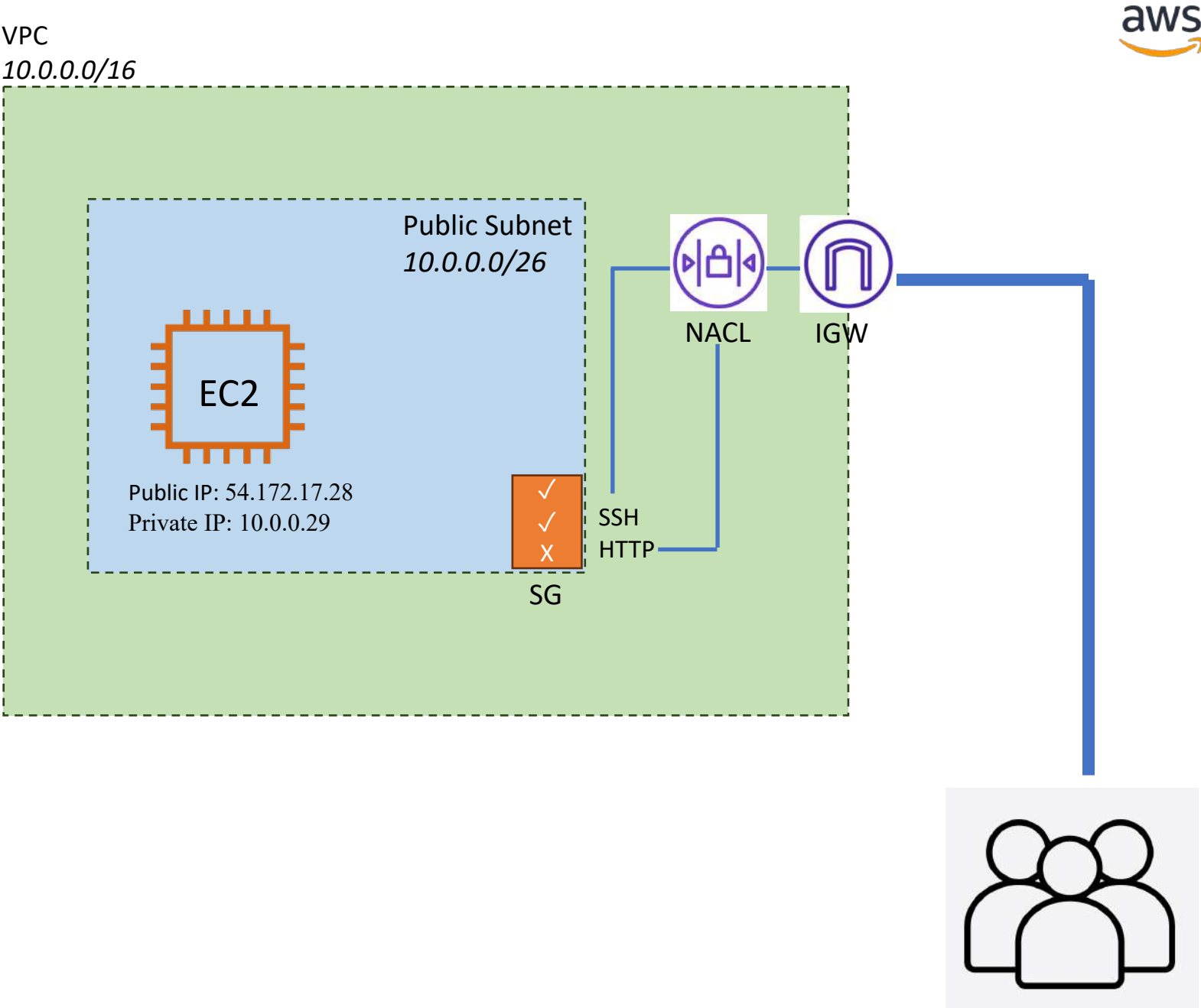
## Security Group:

### Inbound

Type	Port	Source
SSH	22	0.0.0.0/0
HTTP	80	0.0.0.0/0

### outbound

Type	Port	Source
All traffic	All	0.0.0.0/0



# Web Server – Connect

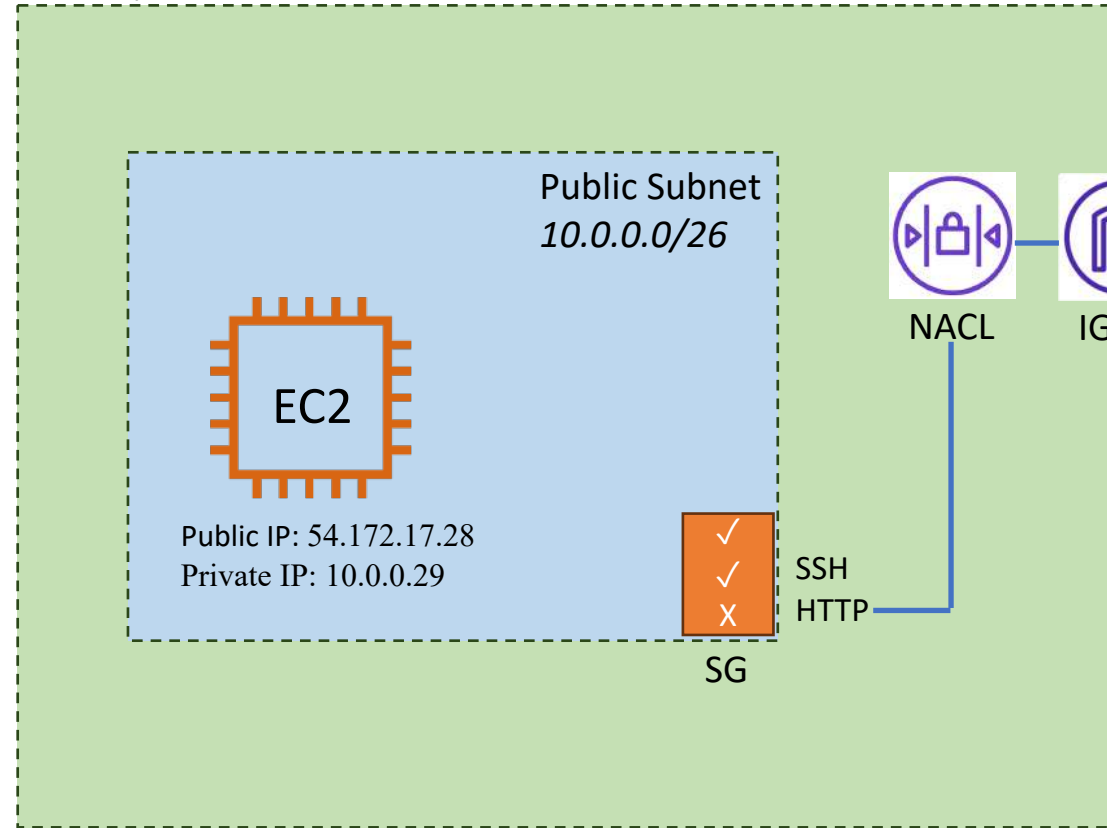
http://54.172.17.28/



Not Secure 54.172.17.28

## Hello World

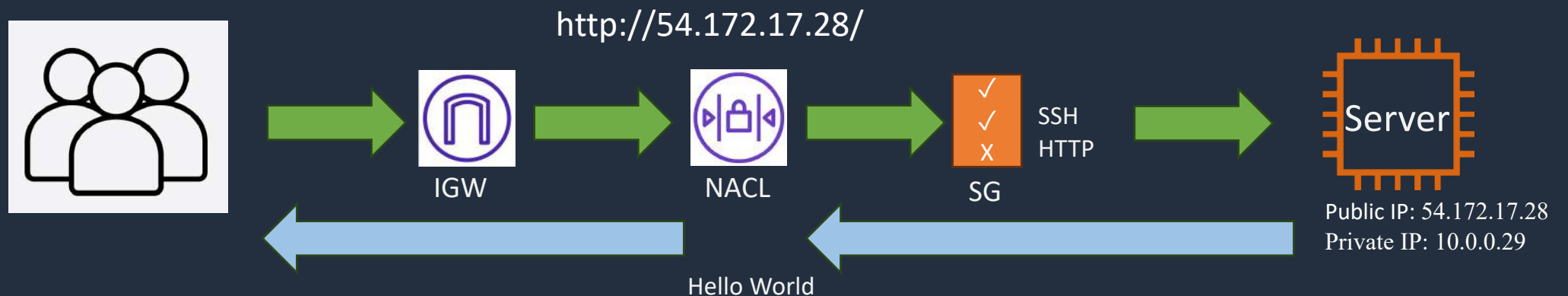
VPC  
10.0.0.0/16



# Q&A

**Q: how does the EC2 instance know its way out to the internet?**

From the VPC section, we need to set up an Internet gateway for the public subnet. Since the EC2 instance is in the public subnet, it can access the Internet. However, the EC2 instance still needs to determine which protocols can be accessed. Therefore, we need to set up inbound rules in the security group to allow SSH or HTTP access.



# Client Side – Set up

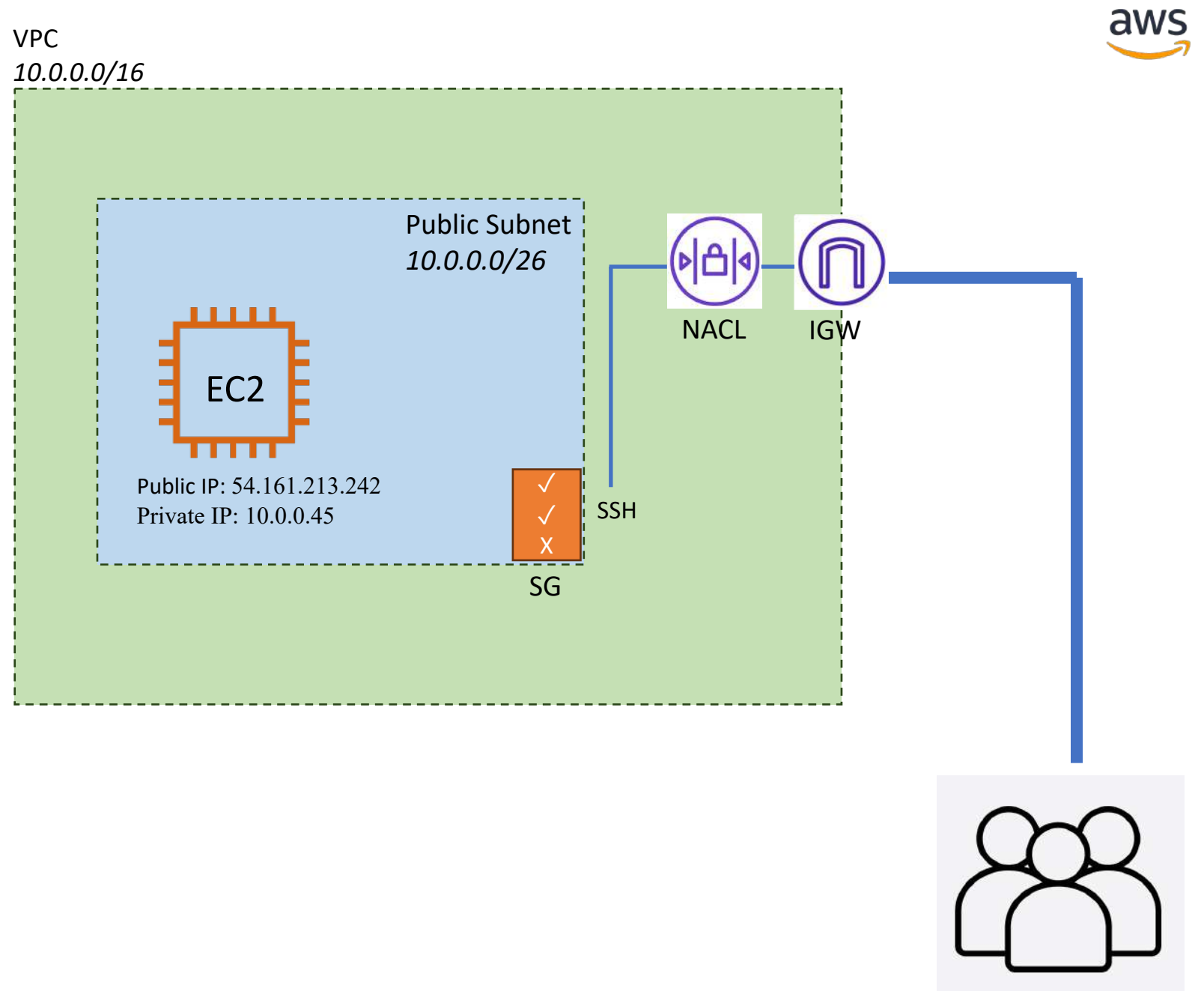
## Security Group:

Inbound

Type	Port	Source
SSH	22	0.0.0.0/0

outbound

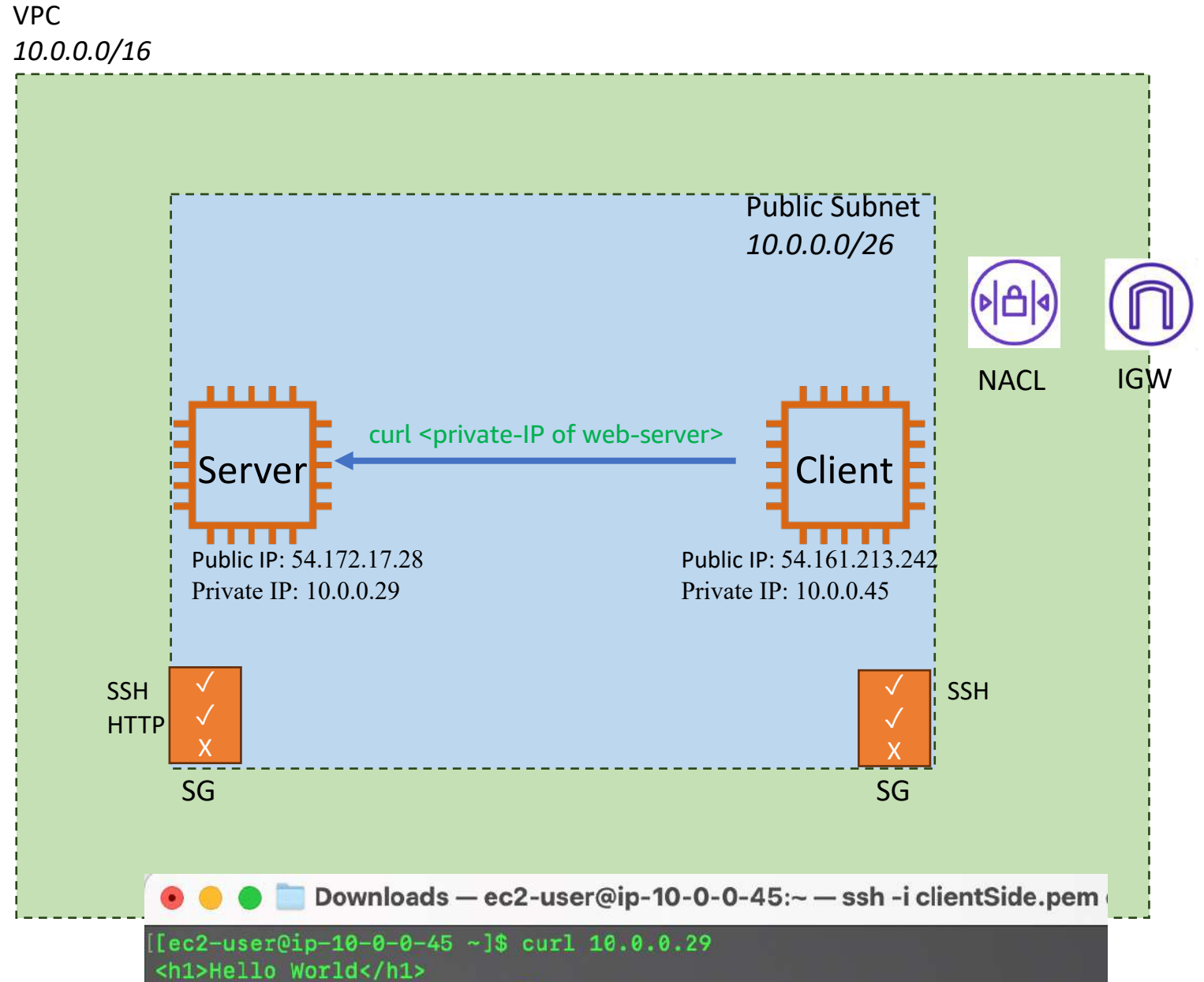
Type	Port	Source
All traffic	All	0.0.0.0/0





# In Client Side

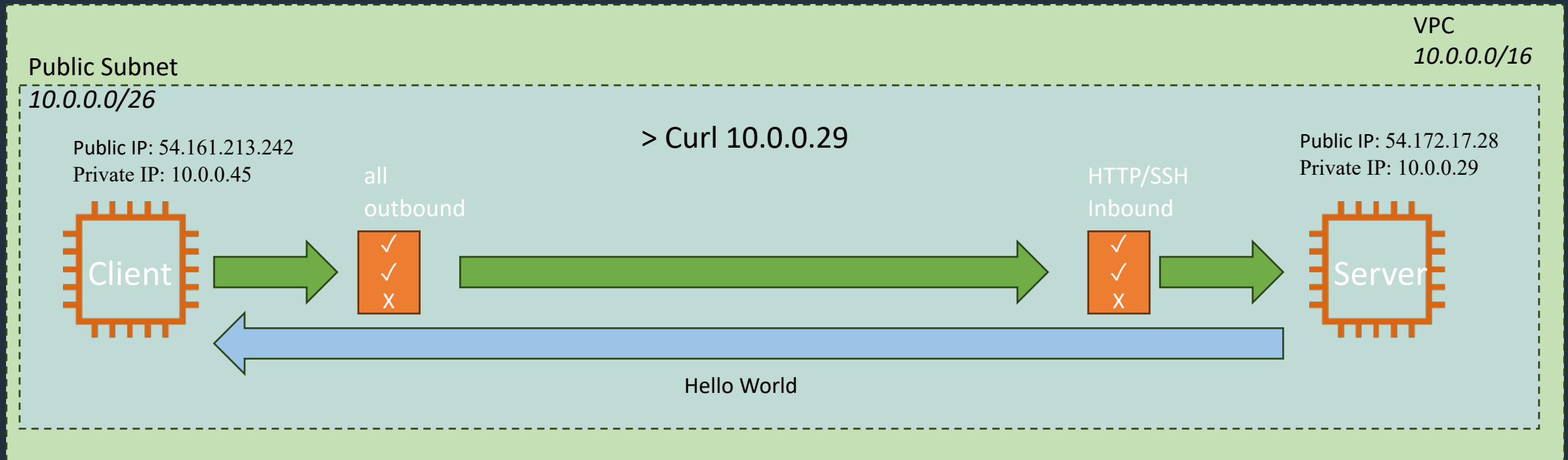
> Curl 10.0.0.29



# Q&A

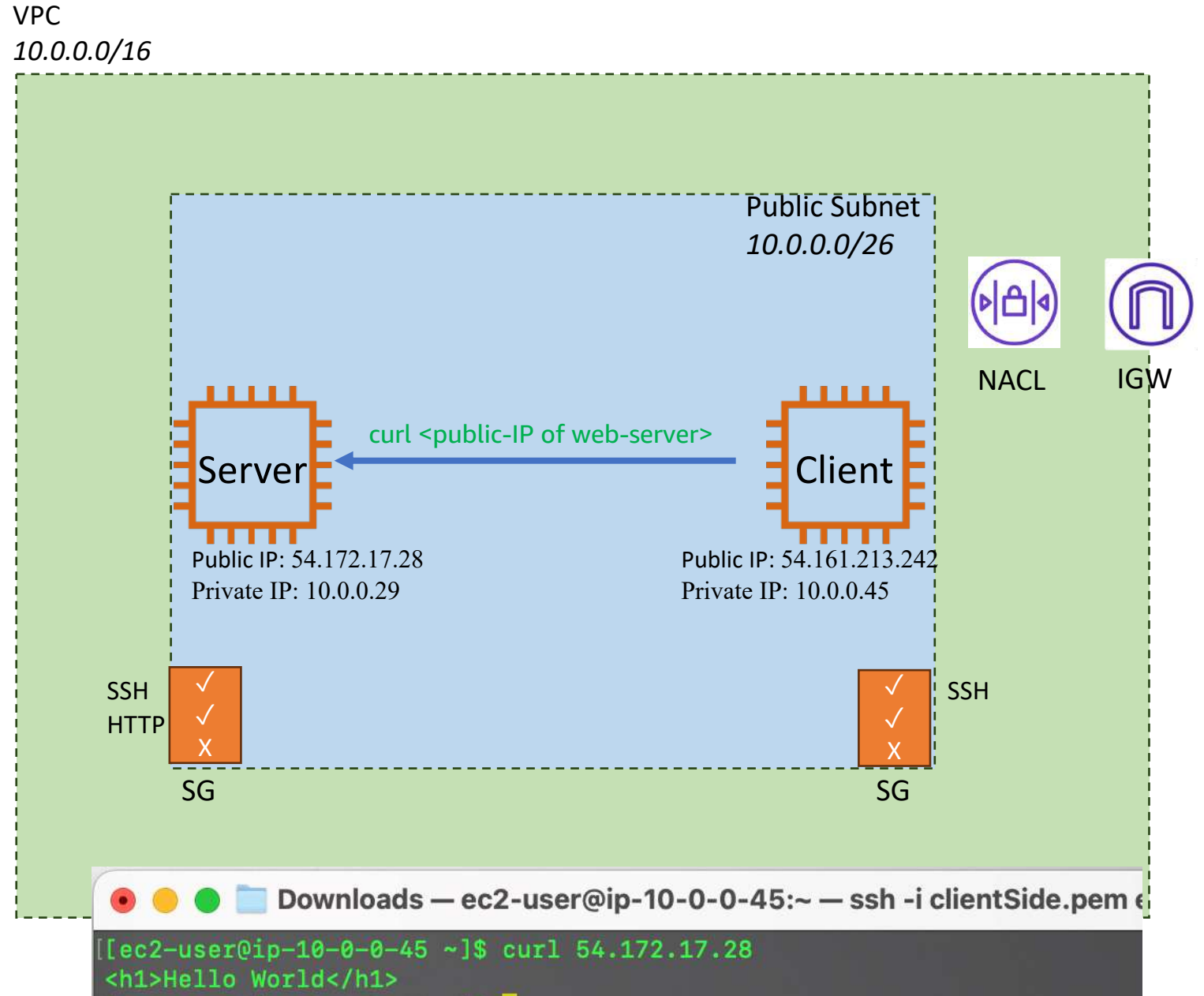
*Q: Does connectivity work? If so, what do you see? Did we traverse the internet while making this request?*

**The connectivity works. The terminal returns “Hello, World.” The traffic does not traverse the internet; instead, it stays within the AWS internal network.**



# In Client Side

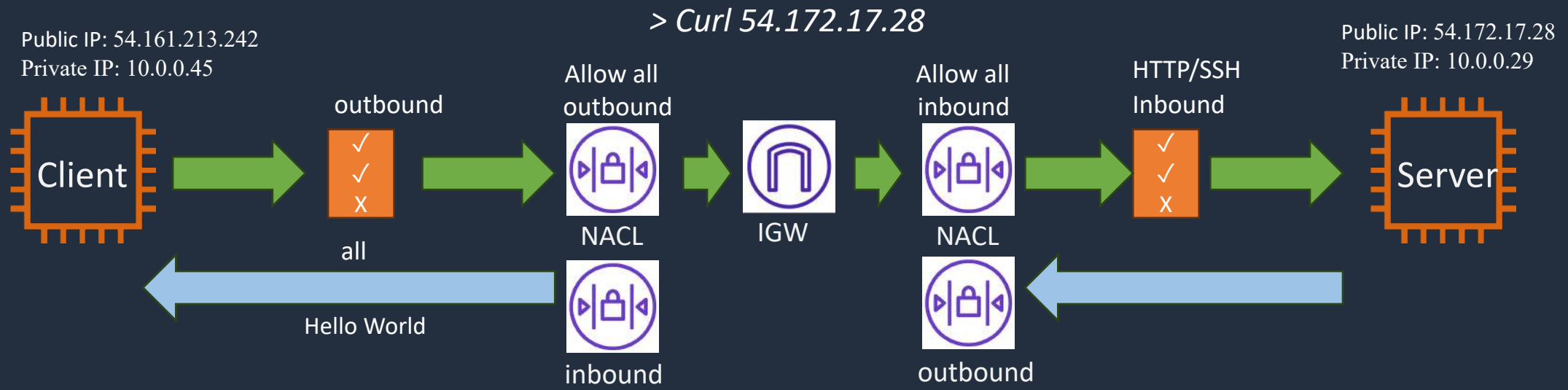
> Curl 54.172.17.28



# Q&A

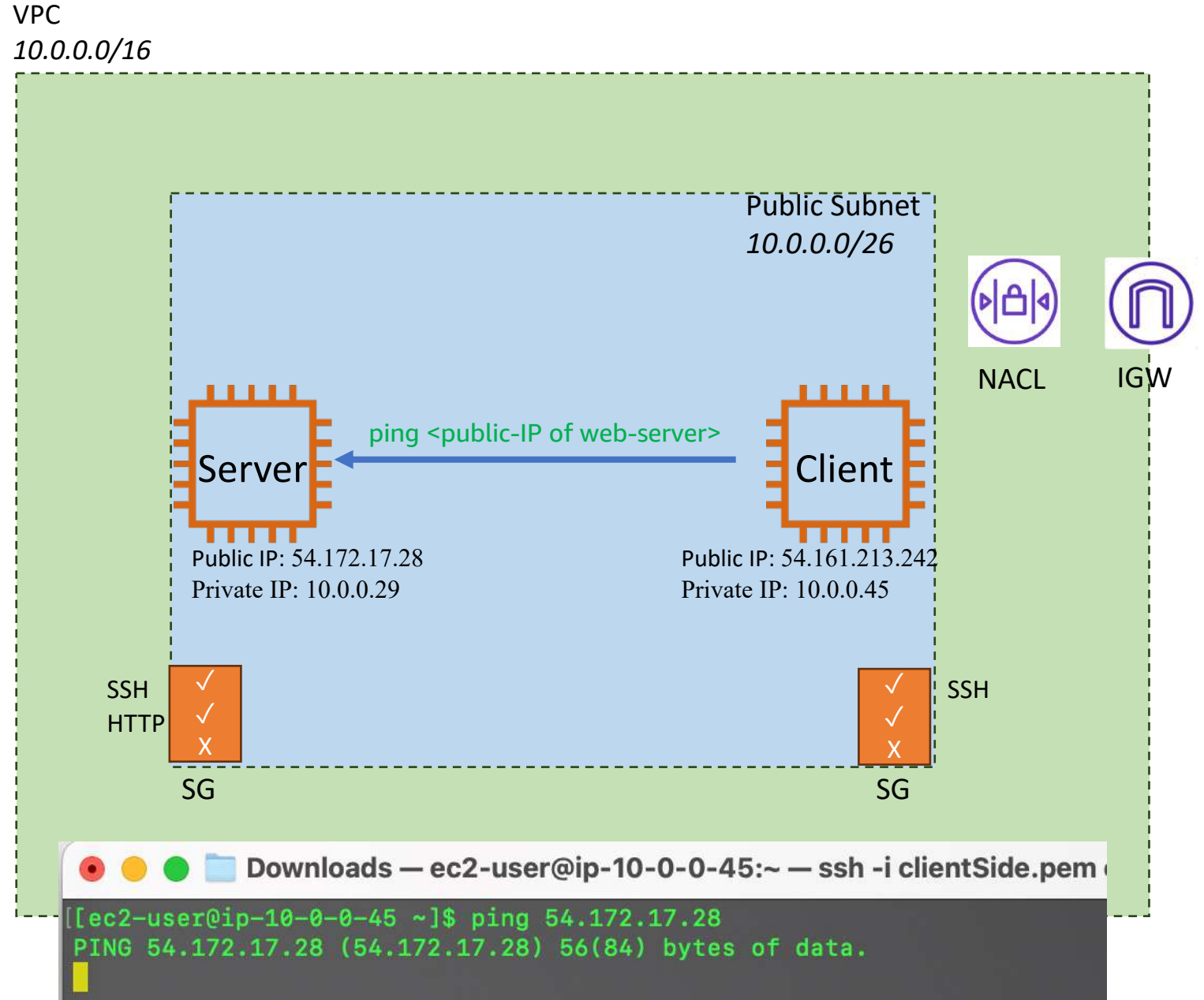
*Q: Is there any difference in response? Did we traverse the internet while making this request?*

Same response, but the traffic does traverse the internet while making this request. Despite both instances being in the same subnet, using the public IP forces the traffic to go outside the internal AWS network, making a round trip over the internet before returning to the second instance. This route involves extra latency and potential security risks compared to using private IPs.



# In Client Side

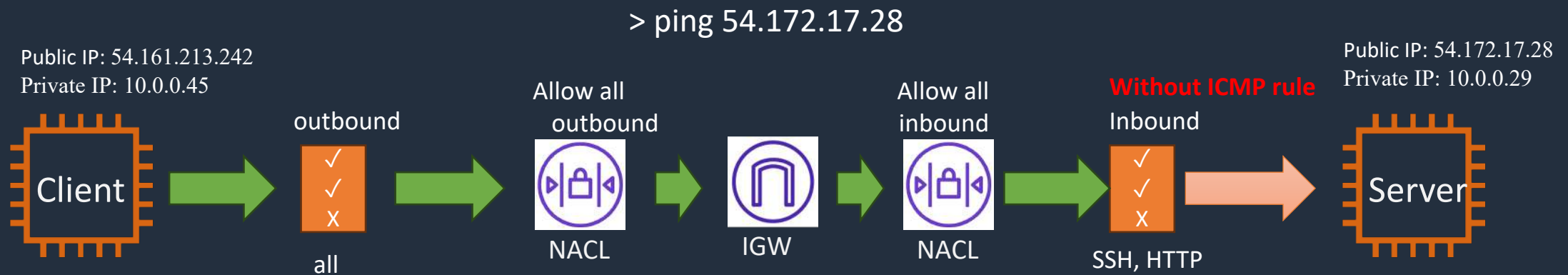
> ping 54.172.17.28



# Q&A

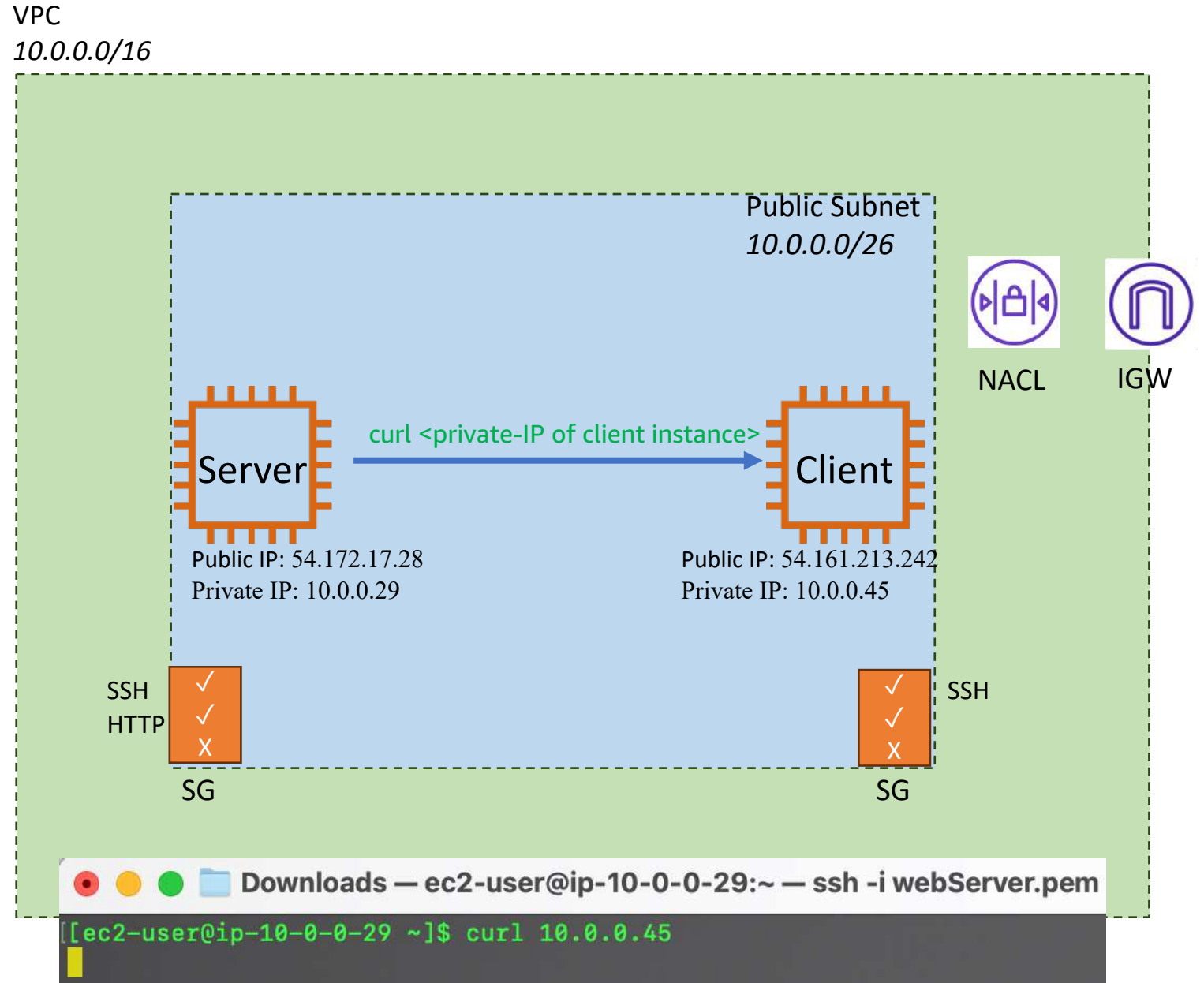
*Q: Did pinging the web-server work? Why? Hint: What protocol does PING use?*

**Pinging the web server does not work because ping requires the ICMP protocol. In the security group of the web server, only SSH & HTTP traffic are allowed.**



# In Web Server

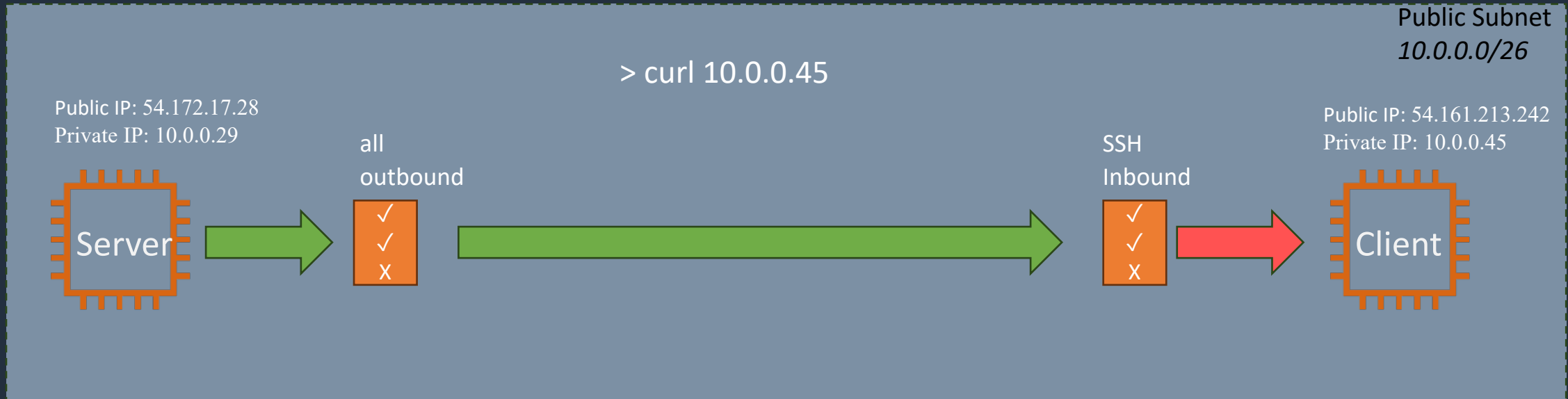
```
> curl 10.0.0.45
```



# Q&A

Q: What was the response? If connectivity failed, explain the possible cause.

There is no response because, in the security group of the client instance, only SSH traffic is allowed, but the curl command requires the HTTP protocol.





# Bonus points

> sudo yum install tcpdump

> sudo tcpdump host 10.0.0.29 and port 80 -w file.pcap

> tcpdump -r file.pcap

> Curl 10.0.0.29

> Curl 10.0.0.29

> Curl 10.0.0.29

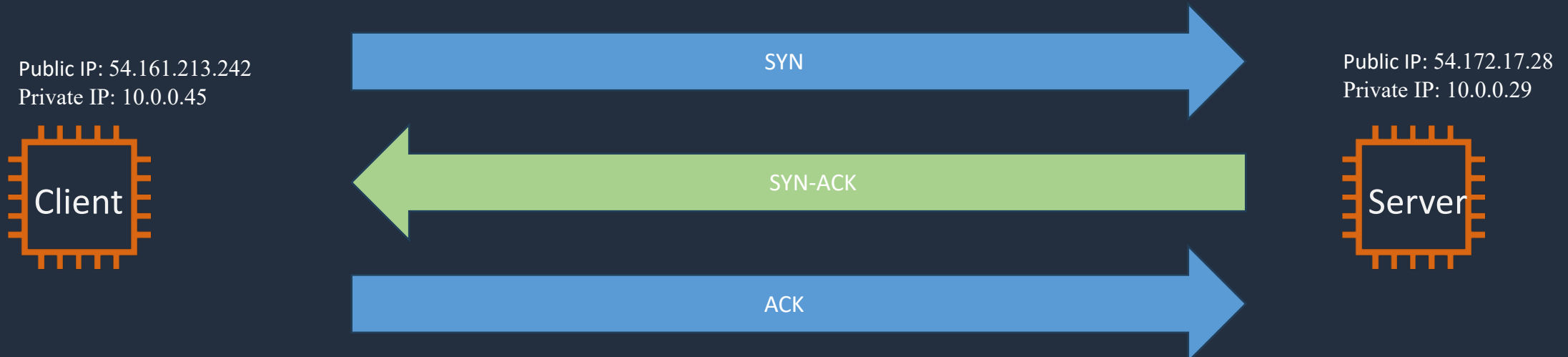
```
Downloads — ec2-user@ip-10-0-0-29:~ — ssh -i webServer.pem ec2-user@54...
[ec2-user@ip-10-0-0-29 ~]$ sudo tcpdump host 10.0.0.45 and port 80 -w file.pcap
dropped privs to tcpdump
tcpdump: listening on enX0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
^C30 packets captured
31 packets received by filter
0 packets dropped by kernel
[ec2-user@ip-10-0-0-29 ~]$ tcpdump -r file.pcap
reading from file file.pcap, link-type EN10MB (Ethernet), snapshot length 262144
22:27:49.963156 IP 10.0.0.45.56178 > ip-10-0-0-29.ec2.internal.http: Flags [S], seq 1
328722462, win 62727, options [mss 8961,sackOK,TS val 3544704136 ecr 0,nop,wscale 7],
length 0
22:27:49.963372 IP ip-10-0-0-29.ec2.internal.http > 10.0.0.45.56178: Flags [S.], seq
3959967355, ack 1328722463, win 62643, options [mss 8961,sackOK,TS val 3473054081 ecr
3544704136,nop,wscale 7], length 0
22:27:49.963757 IP 10.0.0.45.56178 > ip-10-0-0-29.ec2.internal.http: Flags [.], ack 1
, win 491, options [nop,nop,TS val 3544704137 ecr 3473054081], length 0
22:27:49.963757 IP 10.0.0.45.56178 > ip-10-0-0-29.ec2.internal.http: Flags [P.], seq
1:73, ack 1, win 491, options [nop,nop,TS val 3544704137 ecr 3473054081], length 72:
HTTP: GET / HTTP/1.1
22:27:49.963781 IP ip-10-0-0-29.ec2.internal.http > 10.0.0.45.56178: Flags [.], ack 7
3, win 489, options [nop,nop,TS val 3473054082 ecr 3544704137], length 0
22:27:49.964123 IP ip-10-0-0-29.ec2.internal.http > 10.0.0.45.56178: Flags [P.], seq
1:270, ack 73, win 489, options [nop,nop,TS val 3473054082 ecr 3544704137], length 26
9: HTTP: HTTP/1.1 200 OK

Downloads — ec2-user@ip-10-0-0-45:~ — ssh -i clientSide.pem ec2-use...
[ec2-user@ip-10-0-0-45 ~]$ curl 10.0.0.29
<h1>Hello World</h1>
[ec2-user@ip-10-0-0-45 ~]$ curl 10.0.0.29
<h1>Hello World</h1>
[ec2-user@ip-10-0-0-45 ~]$ curl 10.0.0.29
<h1>Hello World</h1>
[ec2-user@ip-10-0-0-45 ~]$
```

# Q&A

Q: Who initiates the TCP three-way handshake?

**Client instance initiates the three-way handshake by sending the first SYN packet. This is typical for HTTP requests, where the client starts the process to establish a TCP connection.**



## Q: What is the HTTP request method?

**“In its simplest form, HTTP is a client/server, one-request/one-response protocol.”**

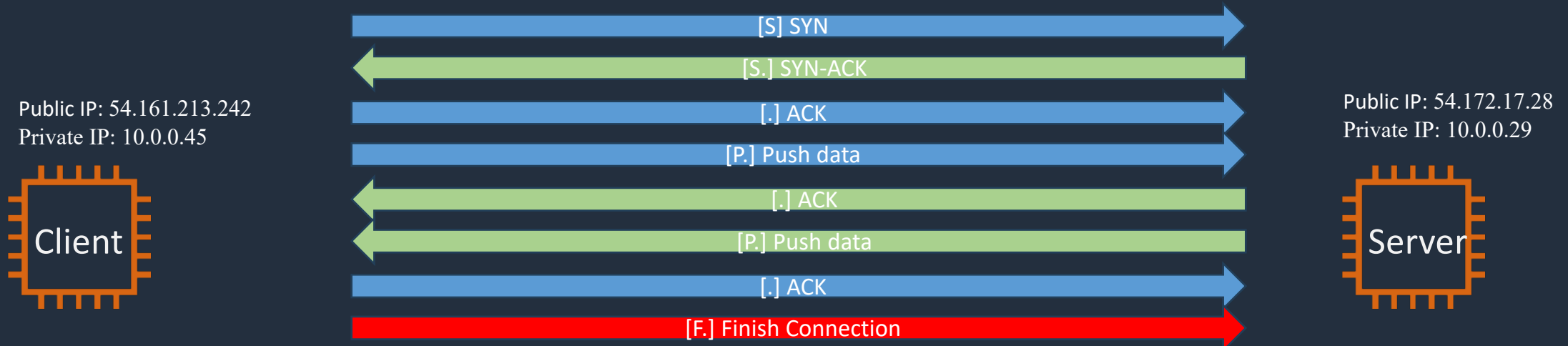
*- 2017 Nemeth Evi et al - UNIX and Linux System Administration Handbook[5thED]\_Rell, P675*

Verb	Purpose
GET	Retrieves the specified resource
HEAD	Like GET, but requests no payload; retrieves metadata only
DELETE	Deletes the specified resource
POST	Applies request data to the given resource
PUT	Similar to POST, but implies replacement of existing contents
OPTIONS	Shows what methods the server supports for the specified path


# Q&A

Q: How is the connection closed between the two peers? (What TCP flags do you see?)

```
[ec2-user@ip-10-0-0-29 ~]$ tcpdump -r file.pcap
reading from file file.pcap, link-type EN10MB (Ethernet), snapshot length 262144
22:27:49.963156 IP 10.0.0.45.56178 > ip-10-0-0-29.ec2.internal.http: Flags [S], seq 1328722462, win 62727, options [mss 8961,sackOK,TS val 3544704136 ecr 0,nop,wscale 7], length 0
22:27:49.963372 IP ip-10-0-0-29.ec2.internal.http > 10.0.0.45.56178: Flags [S.], seq 3959967355, ack 1328722463, win 62643, options [mss 8961,sackOK,TS val 3473054081 ecr 3544704136,nop,wscale 7], length 0
22:27:49.963757 IP 10.0.0.45.56178 > ip-10-0-0-29.ec2.internal.http: Flags [.], ack 1, win 491, options [nop,nop,TS val 3544704137 ecr 3473054081], length 0
22:27:49.963757 IP 10.0.0.45.56178 > ip-10-0-0-29.ec2.internal.http: Flags [P.], seq 1:73, ack 1, win 491, options [nop,nop,TS val 3544704137 ecr 3473054081], length 72: HTTP: GET / HTTP/1.1
22:27:49.963781 IP ip-10-0-0-29.ec2.internal.http > 10.0.0.45.56178: Flags [.], ack 73, win 489, options [nop,nop,TS val 3473054082 ecr 3544704137], length 0
22:27:49.964123 IP ip-10-0-0-29.ec2.internal.http > 10.0.0.45.56178: Flags [P.], seq 1:270, ack 73, win 489, options [nop,nop,TS val 3473054082 ecr 3544704137], length 269: HTTP: HTTP/1.1 200 OK
22:27:49.964541 IP 10.0.0.45.56178 > ip-10-0-0-29.ec2.internal.http: Flags [.], ack 270, win 489, options [nop,nop,TS val 3544704138 ecr 3473054082], length 0
22:27:49.964707 IP 10.0.0.45.56178 > ip-10-0-0-29.ec2.internal.http: Flags [F.], seq 73, ack 270, win 489, options [nop,nop,TS val 3544704138 ecr 3473054082], length 0
```



# Thank You!

A solid orange horizontal bar.

Networking Project  
Jason Zhang