AWS re:INVENT

Another Day, Another Billion Flows

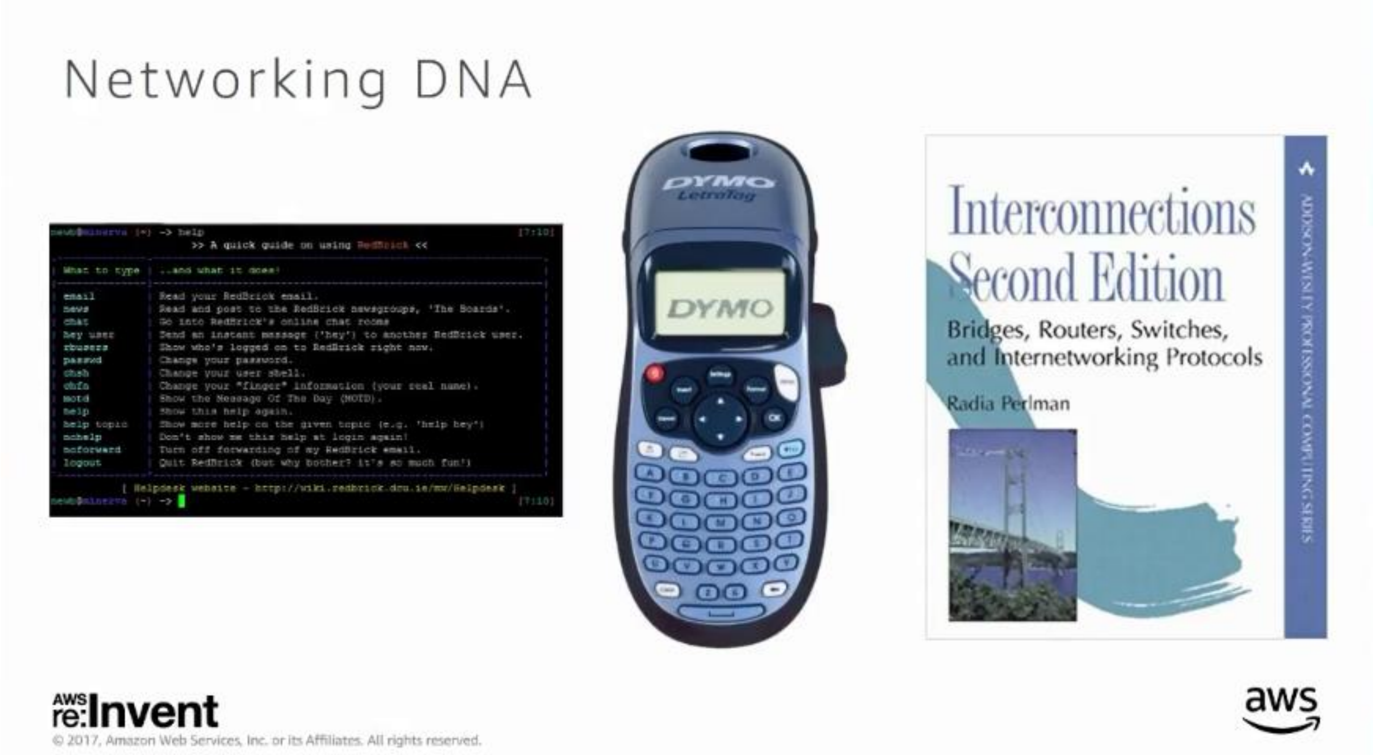
Colm MacCárthaigh

November 29, 2017

AWS re:Invent

aws

In this session, we walk through the ***Amazon VPC network*** and describe the problems we were solving when we created it, and the features we've been adding as we scale it. We cover how these problems and features are traditionally solved, and why those solutions are not scalable, inexpensive, or secure enough for AWS. Finally, we provide an overview of the solution that we've implemented. We discuss some of the unique mechanisms that we use to ensure customer isolation, get packets into and out of the network, and support new features such as ***NAT and VPC endpoints***



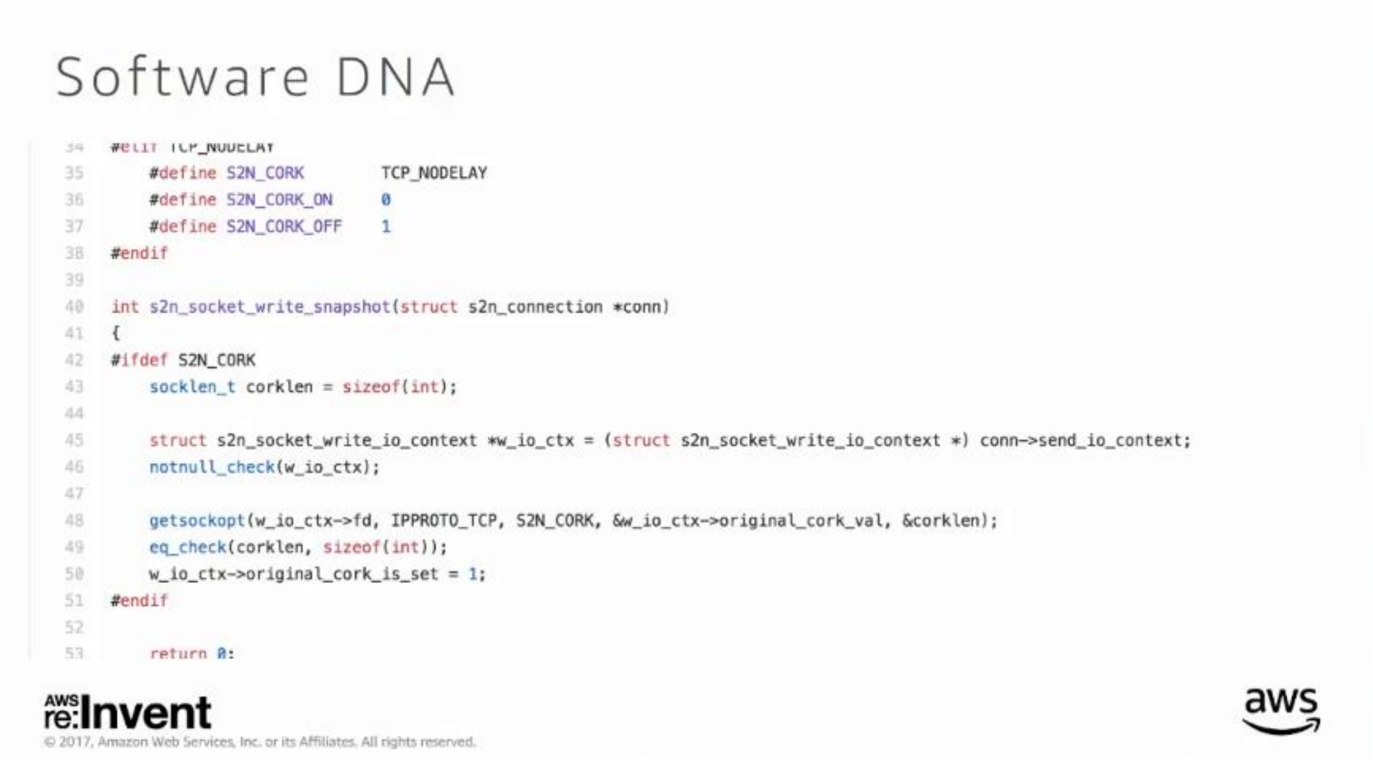
Networking DNA

```
newb@minerva (~) -> help                                                  [7:10]
                    >> A quick guide on using RedBrick <<
 ----------------------------------------------------------------------------
| What to type | ..and what it does!                                          |
 ----------------------------------------------------------------------------
| email        | Read your RedBrick email.                                    |
| news         | Read and post to the RedBrick newsgroups, 'The Boards'.      |
| chat         | Go into RedBrick's online chat rooms                         |
| hey user     | Send an instant message ('hey') to another RedBrick user.    |
| rbusers      | Show who's logged on to RedBrick right now.                  |
| passwd       | Change your password.                                        |
| chsh         | Change your user shell.                                      |
| chfn         | Change your "finger" information (your real name).           |
| motd         | Show the Message Of The Day (MOTD).                          |
| help         | Show this help again.                                        |
| help topic   | Show more help on the given topic (e.g. 'help hey')          |
| nohelp       | Don't show me this help at login again!                      |
| noforward    | Turn off forwarding of my RedBrick email.                    |
| logout       | Quit RedBrick (but why bother? it's so much fun!)            |
 ----------------------------------------------------------------------------
          [ Helpdesk website - http://wiki.redbrick.dcu.ie/mw/Helpdesk ]
newb@minerva (~) ->                                                       [7:10]
```

DYMO
LetraTag
DYMO
Interconnections
Second Edition
Bridges, Routers, Switches, and Internetworking Protocols
Radia Perlman
ADDISON-WESLEY PROFESSIONAL COMPUTING SERIES

AWS re:Invent

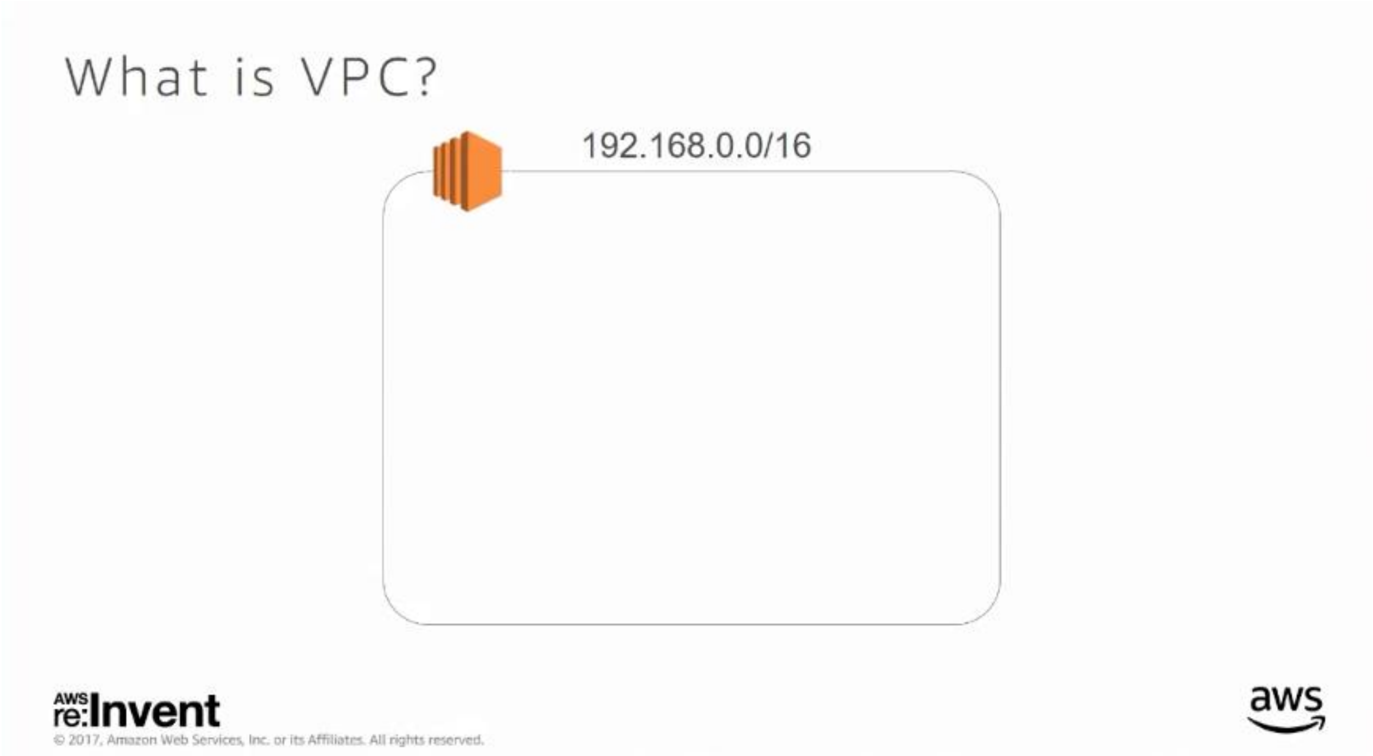
aws



Software DNA

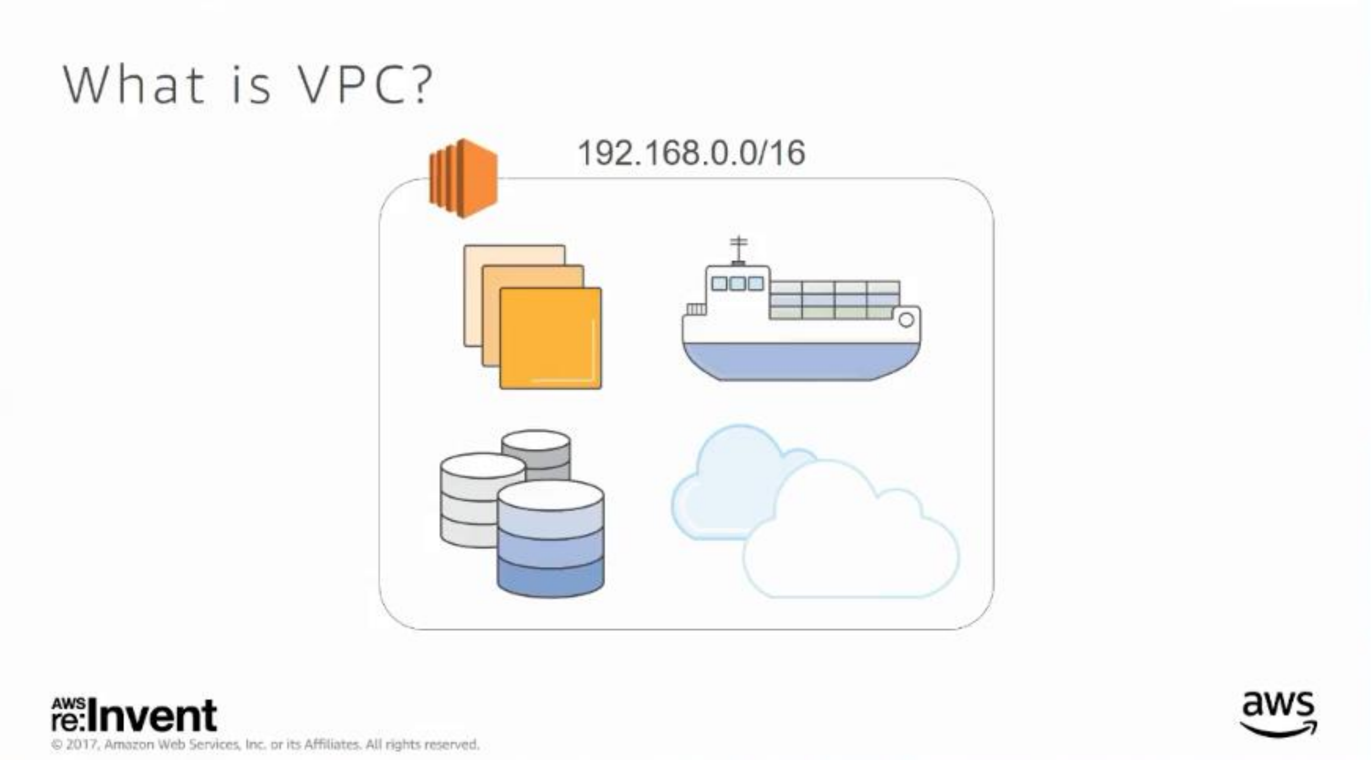
```
#elif TCP_NODELAY
    #define S2N_CORK        TCP_NODELAY
    #define S2N_CORK_ON     0
    #define S2N_CORK_OFF    1
#endif

int s2n_socket_write_snapshot(struct s2n_connection *conn)
{
#ifdef S2N_CORK
    socklen_t corklen = sizeof(int);

    struct s2n_socket_write_io_context *w_io_ctx = (struct s2n_socket_write_io_context *) conn->send_io_context;
    notnull_check(w_io_ctx);

    getsockopt(w_io_ctx->fd, IPPROTO_TCP, S2N_CORK, &w_io_ctx->original_cork_val, &corklen);
    eq_check(corklen, sizeof(int));
    w_io_ctx->original_cork_is_set = 1;
#endif

    return 0;
```

Virtual Private Cloud ***VPC*** is a software defined network that is built entirely by code and supported by a physical network infrastructure.



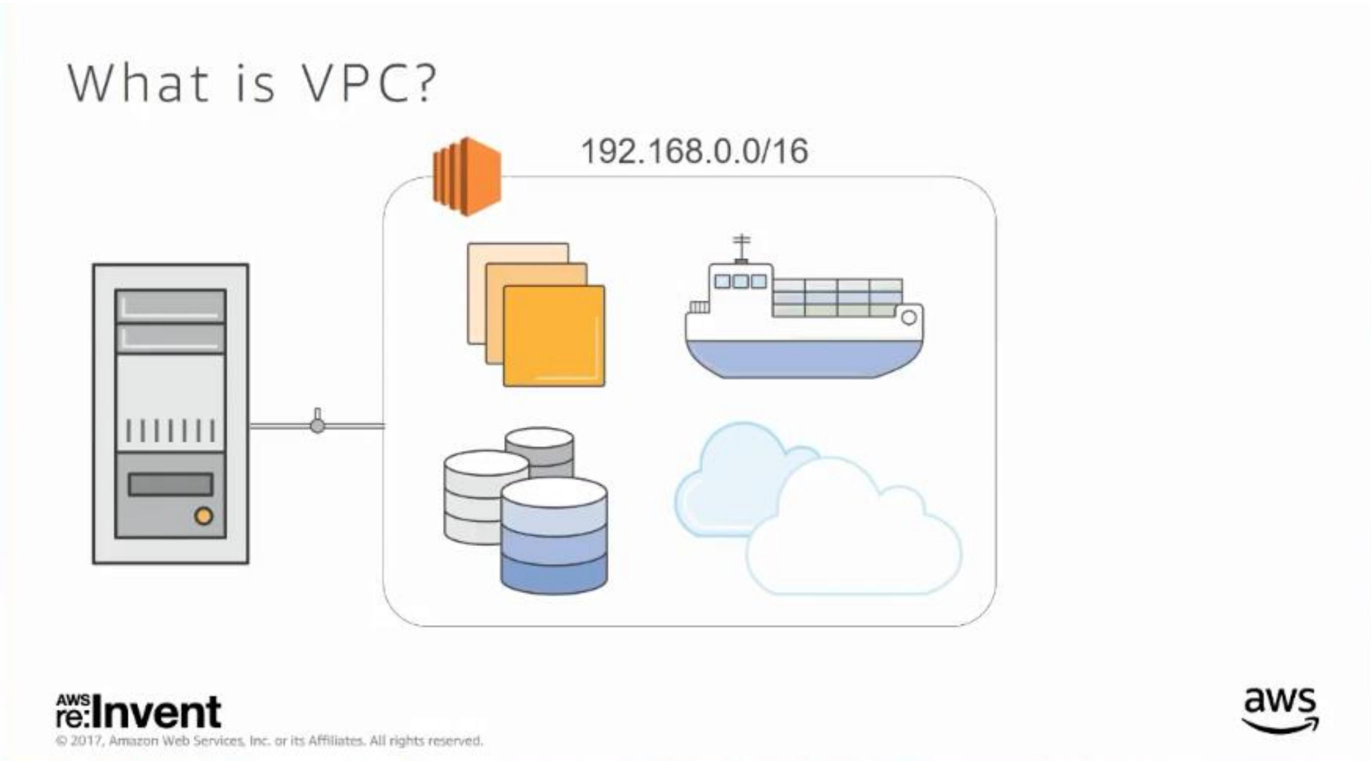
What is VPC?

Customers can create a SDN and assign it a network range they want like 192.168.0.0/16 above



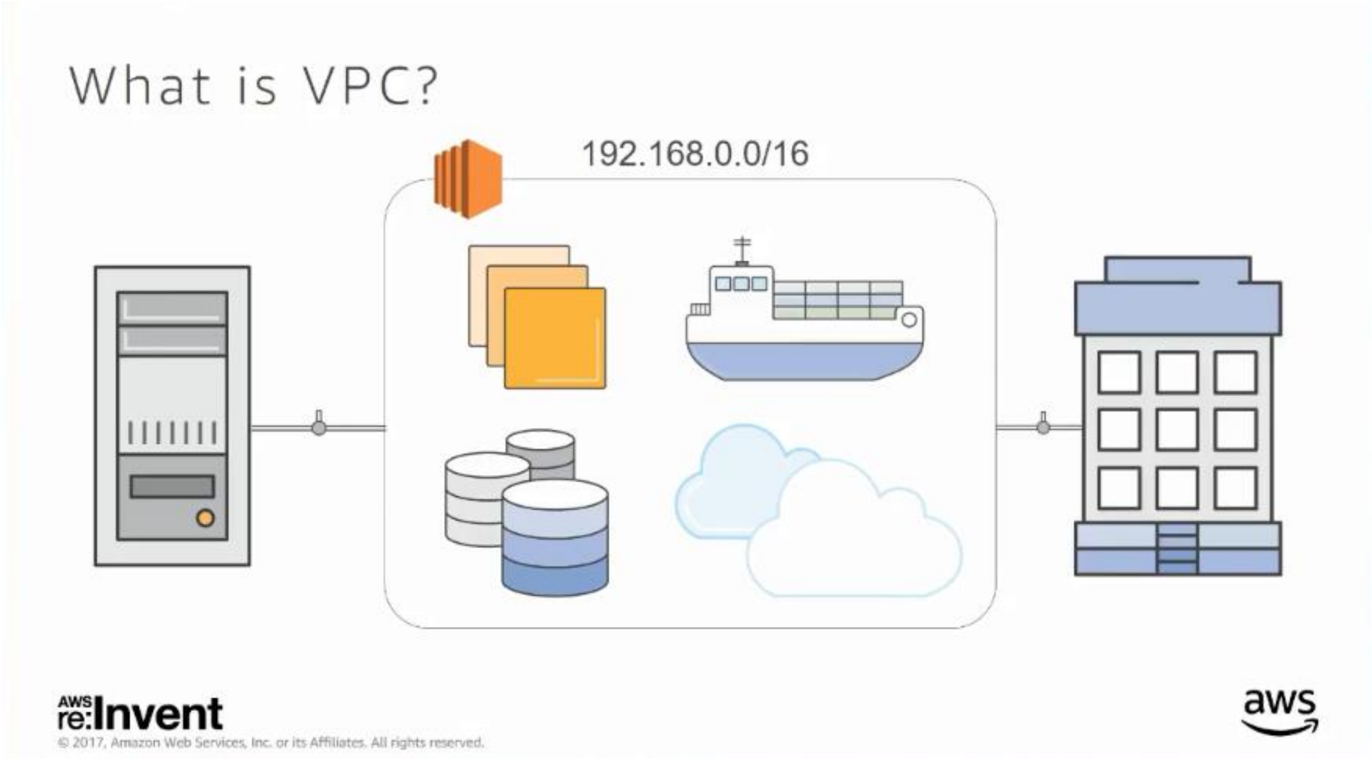
What is VPC?

Inside the VPC we can launch things that can talk to each other like EC2 instances, containers, RDS databases or your own databases, and all sorts of other cloud resources. These can all be launched directly into the VPC, into subnets, etc. all under your control. All those resources launched into the VPC will get IP addresses that makes sense in the context of the VPN and are fully under your control.



What is VPC?

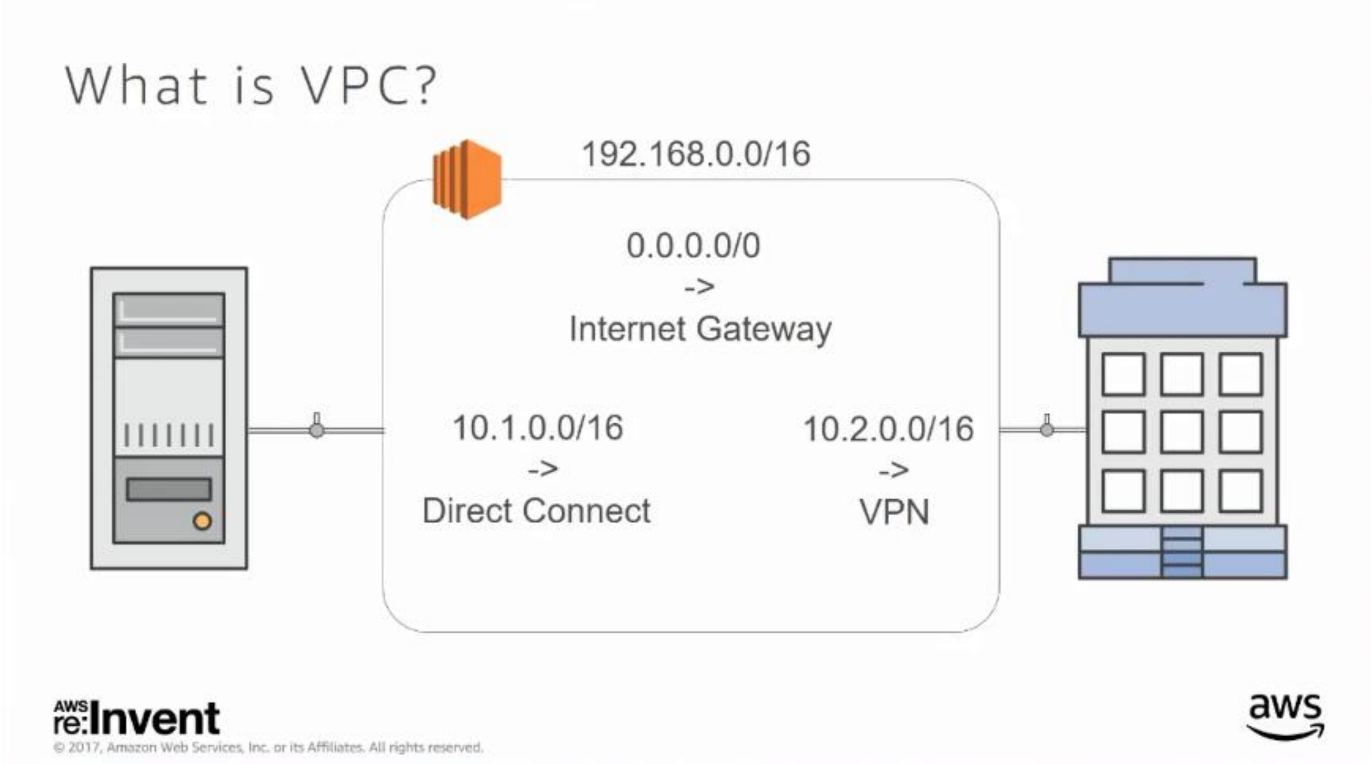
You can connect your VPC network to other networks using the ***AWS DirectConnect*** product, you can now interconnect the VPC to an on-premises data center where you have other machines. This is why it is important so you can pick the VPC IP range so as not to conflict with the IP ranges used in your other networks.



What is VPC?

192.168.0.0/16

We can also connect the VPC to any VPN products in our offices to access the VPC resources over private addresses at work.



What is VPC?

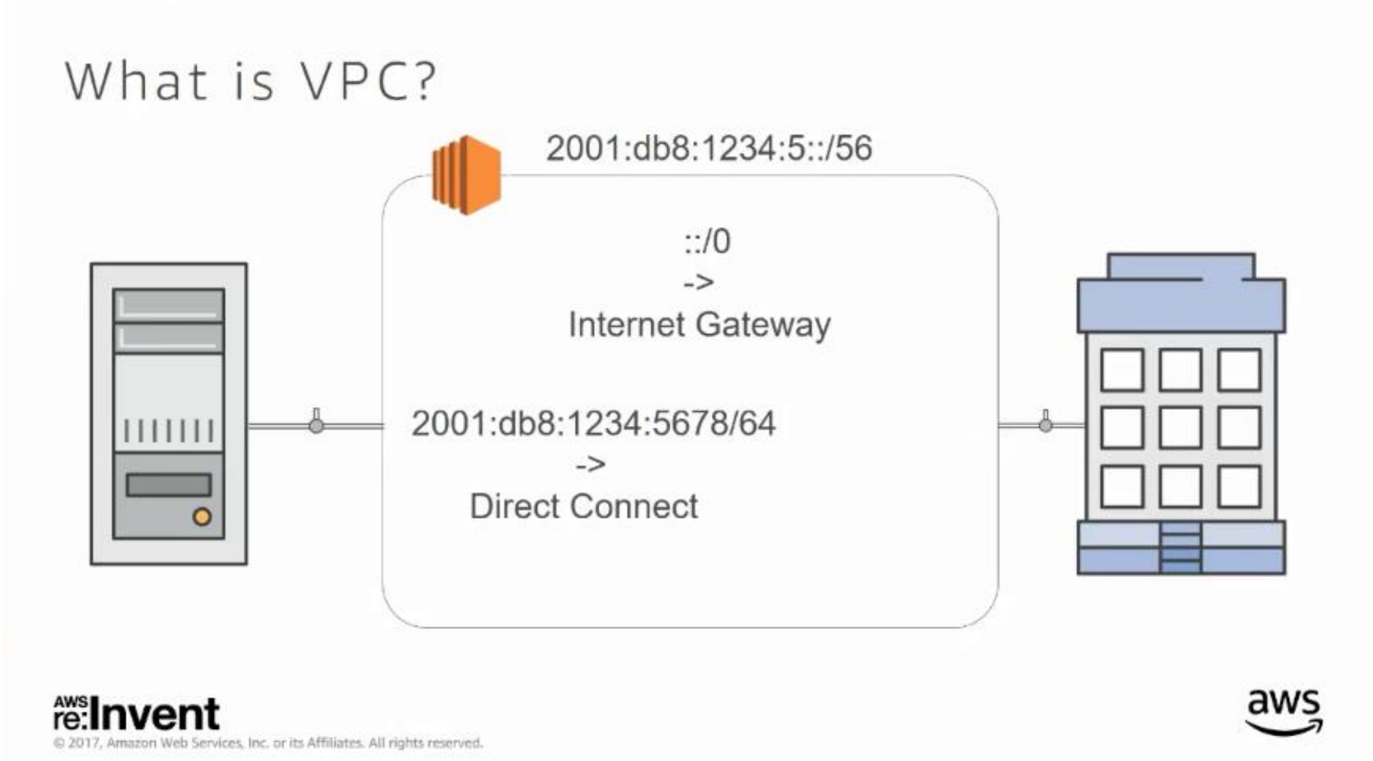
192.168.0.0/16

0.0.0.0/0
->
Internet Gateway

10.1.0.0/16
->
Direct Connect

10.2.0.0/16
->
VPN

VPC also supports ***routing tables*** so that you can ***create routes inside your VPC to say which route prefix goes where*** or there inside your VPC.



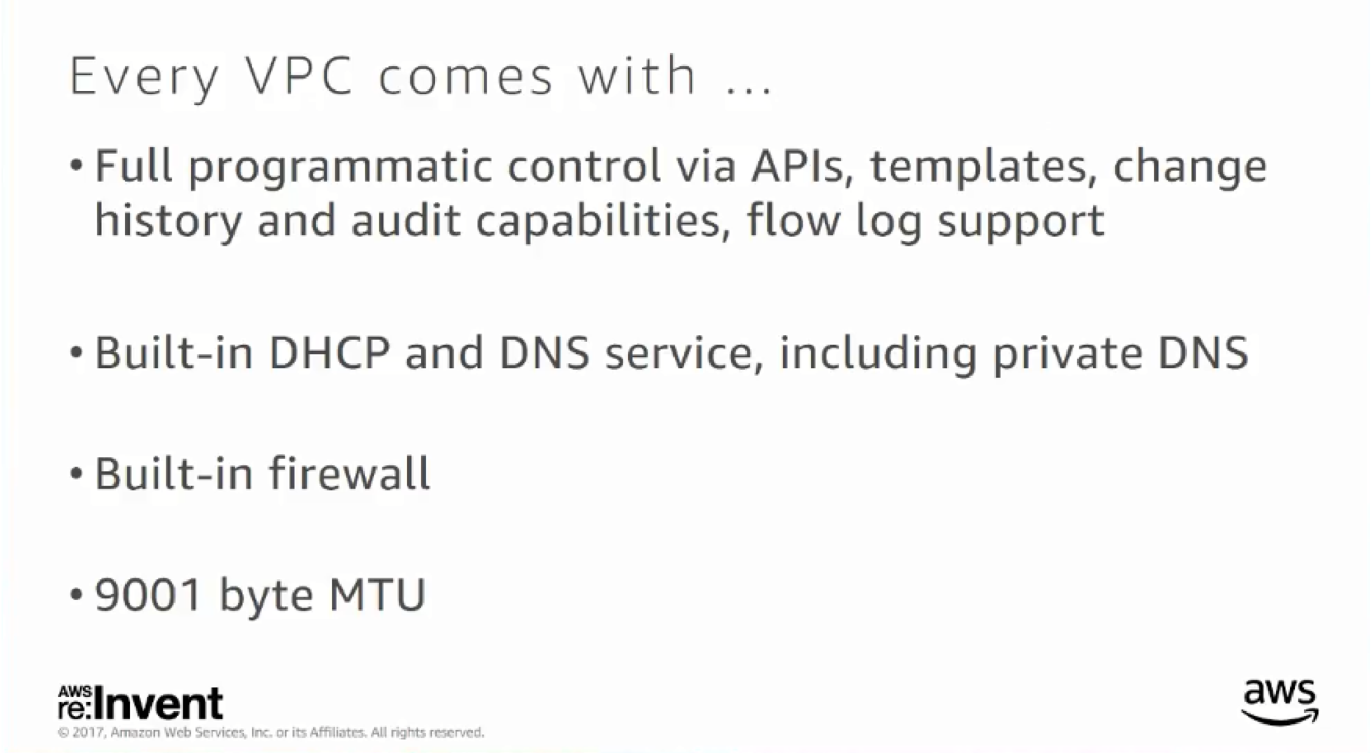
What is VPC?

2001:db8:1234:5::/56

::/0
->
Internet Gateway

2001:db8:1234:5678/64
->
Direct Connect

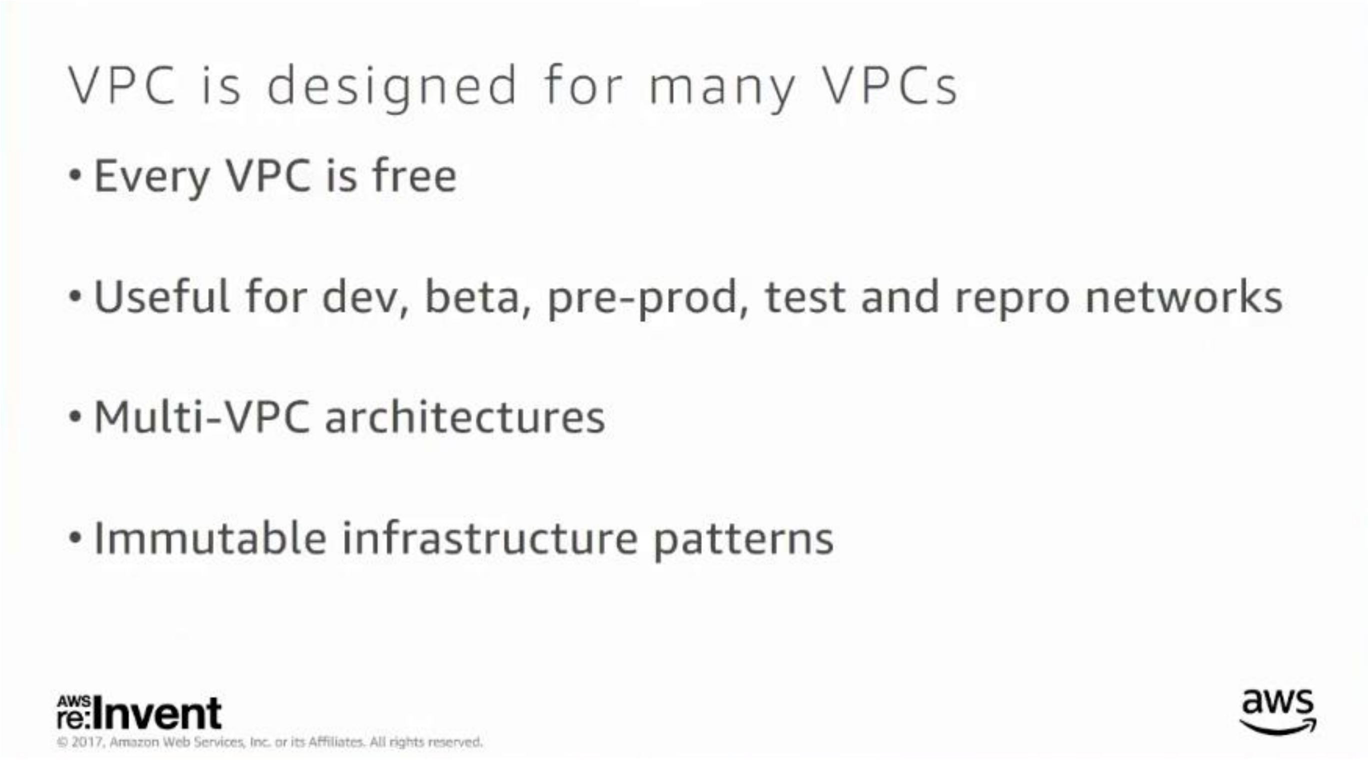
Yu can also use IPv6 in addition to the IPv4 already available in your VPC, it runs the HTTPS IP protocol



Every VPC comes with …

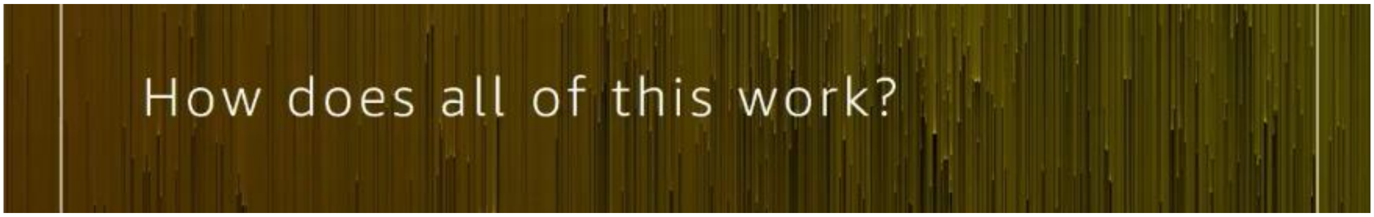
- Full programmatic control via APIs, templates, change history and audit capabilities, flow log support
- Built-in DHCP and DNS service, including private DNS
- Built-in firewall
- 9001 byte MTU

Using software to create the VPC using ***programmable APIs***, we can create routing tables, launch instances, assign IPs, etc. using the APIs. We can template those resources using ***CloudFormation*** or other templating tools to build automation. We also get change history and audit logs about things that happen within our VPC. ***VPC inter-region pairing*** is now possible, you can now pair VPCs in multiple regions. Route53 private DNS also now works across regions, you can now use a private DNS zone in Route53 with VPC's in different regions. ***VPC also has built-in firewalls using Network ACLs and Security Groups***. These are network control methods that allow you to stop or allow traffic on a source-to-destination basis. VPC also supports large packet sizes up to 1500 bytes to 9001 byte MTU packet size.

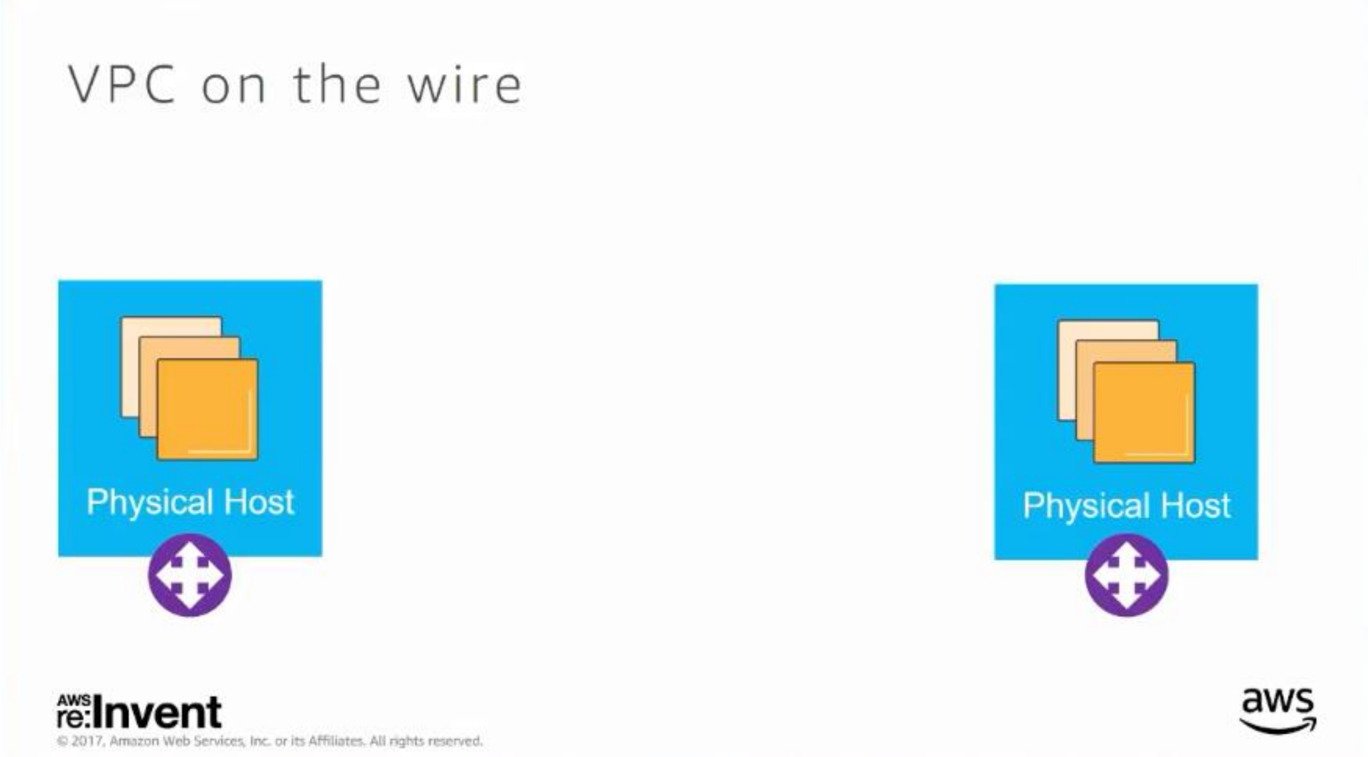


VPC is designed for many VPCs

- Every VPC is free
- Useful for dev, beta, pre-prod, test and repro networks
- Multi-VPC architectures
- Immutable infrastructure patterns

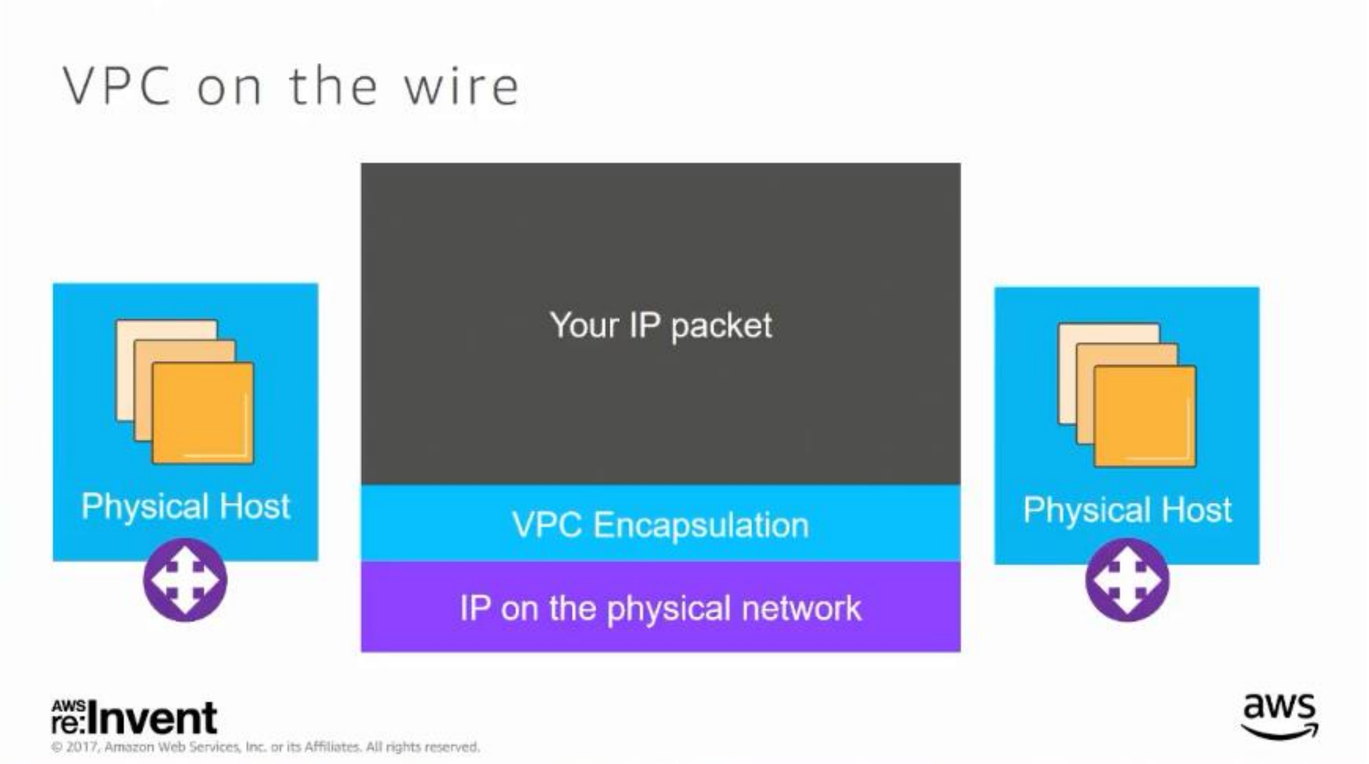


How does all of this work?



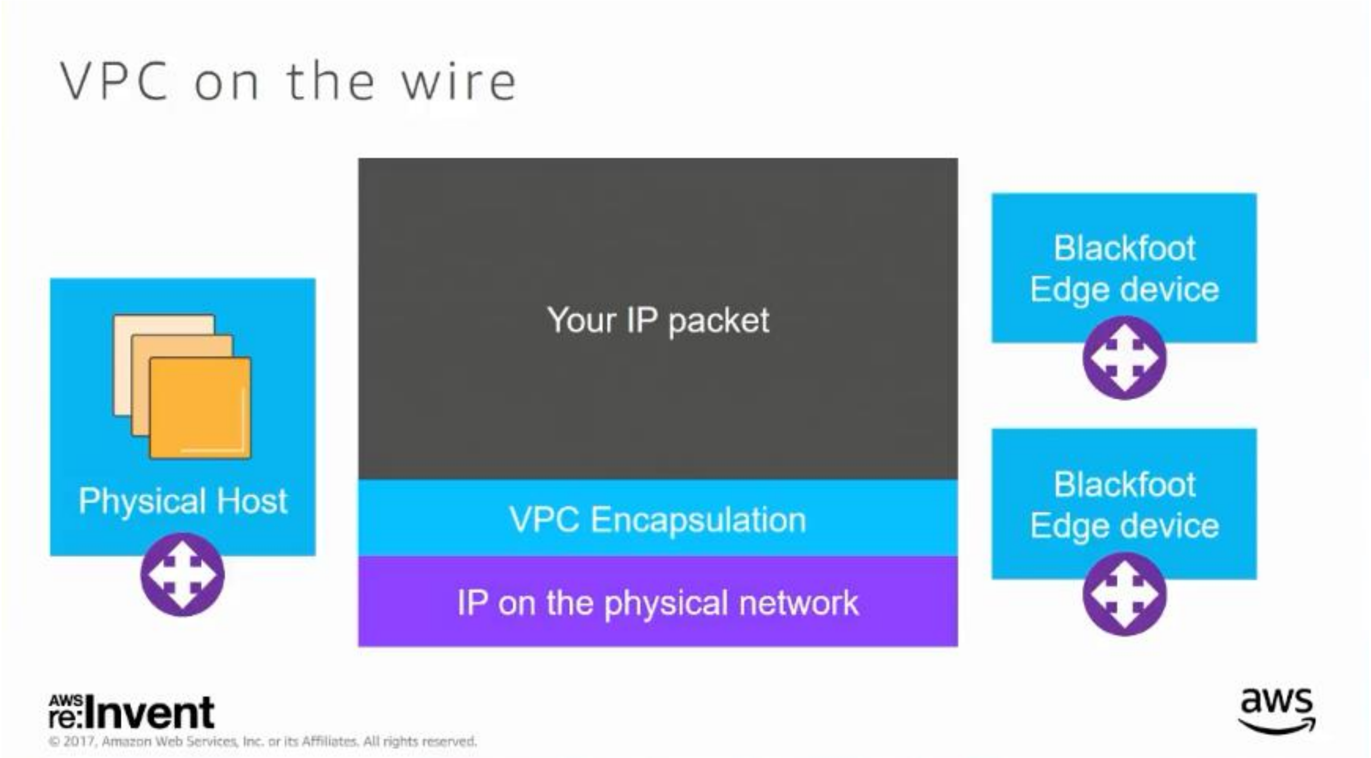
VPC on the wire

Your ***Virtual Machines*** are actually running on some physical box with a ***virtual router*** running on the same level as the ***hypervisor***, the virtual router is responsible for making VPC work as virtual isolated networks as packets move around many VPCs created by customers.

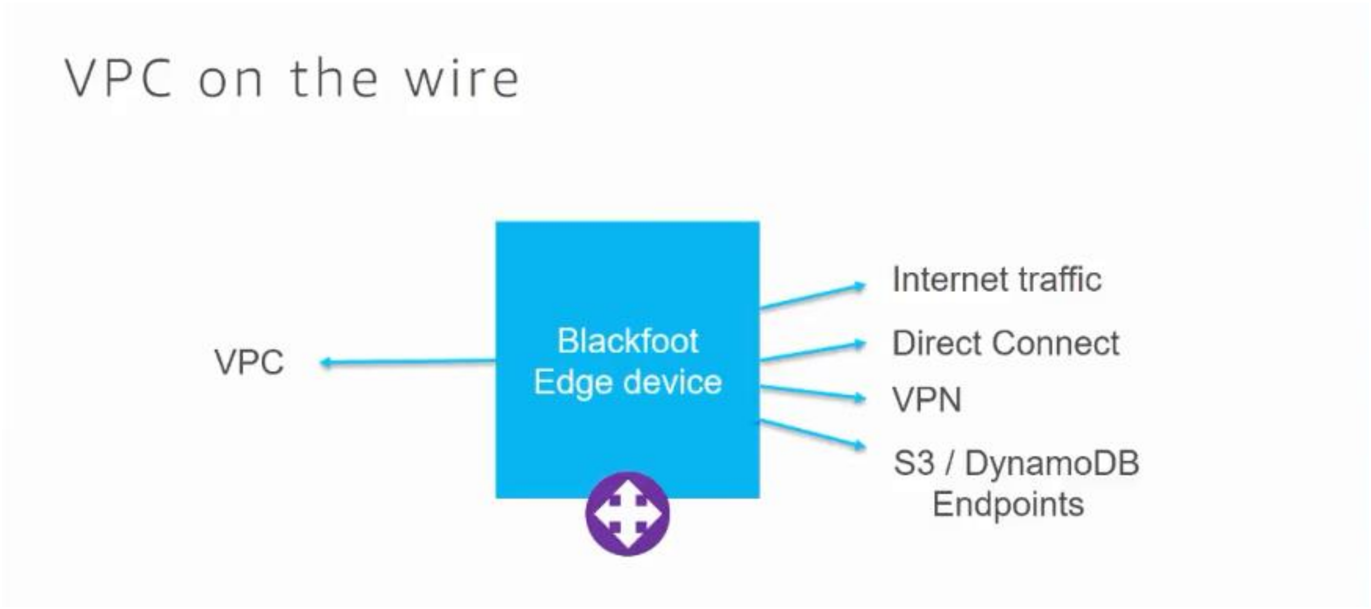


VPC on the wire

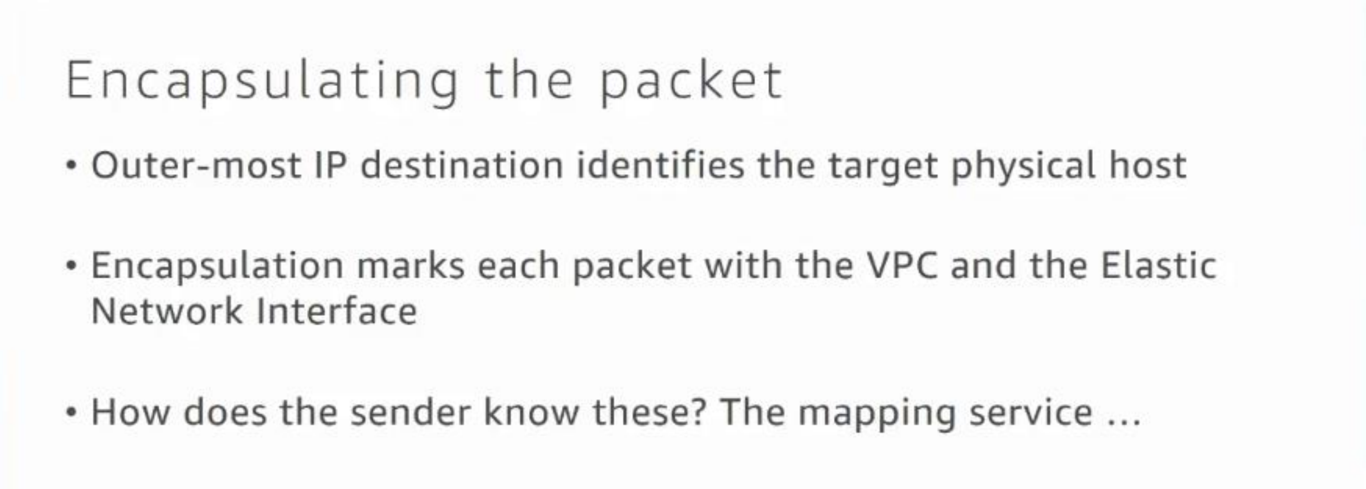
For traffic/communication between 2 instances sitting on different physical hosts, there are many technologies that allow you to do the VPC disambiguation's and identification of packets and their VPCs within the larger physical network, like MPLS, VLANs, etc. but none is cloud scale and AWS uses a proprietary encapsulation for disambiguation and to identify packets on the wire. When you look at packet traffic on the wire, you would see ordinary IP on the physical network at the outermost layer as packets flow between the physical hosts on the physical network infrastructure. Above that physical network sits the ***VPC Encapsulation protocol***, where AWS stamps its own data on the packet for disambiguation, this includes information like which VPC the packet is for, which Elastic Network Interface ENI the packet uses, etc. as the packets flows between physical machines. On top of that sits the normal IP packet that is being transmitted by your instances. Note that this does not cover the fact that we can send traffic elsewhere like to the internet, other physical private networks, etc.



We use the ***Blackfoot Edge Device*** to translates from the inside VPC network to the outside normal IP networking. It uses an embedded router within it that helps it to encapsulate and de-encapsulate packets, unwrap it, and do something with it before sending it on.

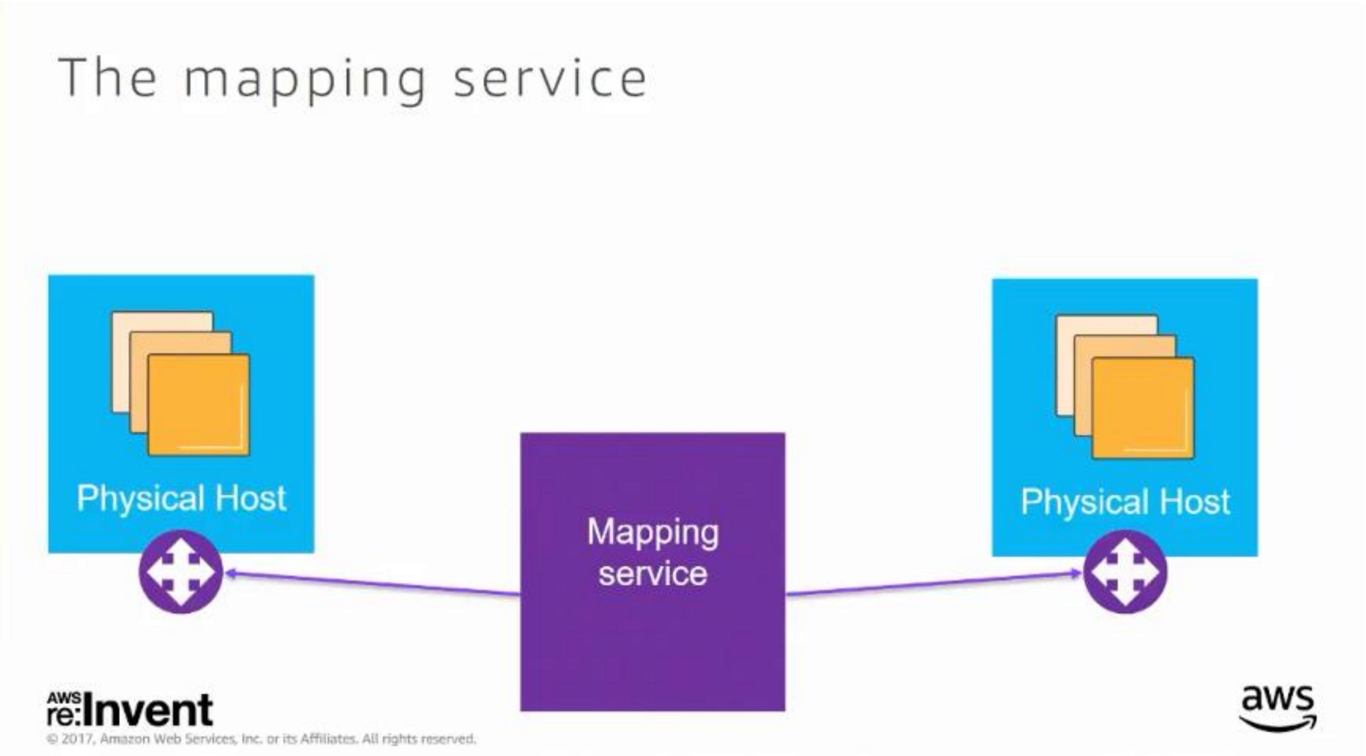


The Blackfoot Edge Device ***BED*** can send traffic to a bunch of destinations from your VPC. For traffic going out, the BED strips the VPC encapsulation, translate the IP addresses from that used from inside VPC to that expected by the outside public internet/private VPCs. Similarly, with using DirectConnect, BED encapsulates the traffic and passes it on to another network that actually implements DirectConnect and starts doing things with VLANs used in DirectConnect. This is how VPNs, S3 and DynamoDB endpoints work.



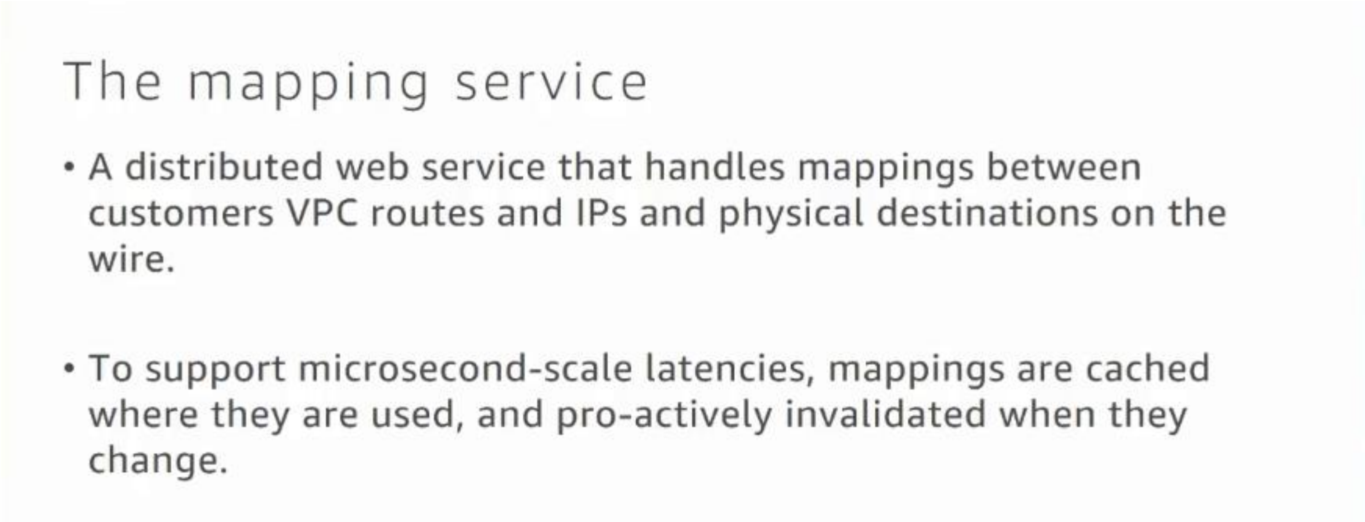
Encapsulating the packet

- Outer-most IP destination identifies the target physical host
- Encapsulation marks each packet with the VPC and the Elastic Network Interface
- How does the sender know these? The mapping service …



The mapping service

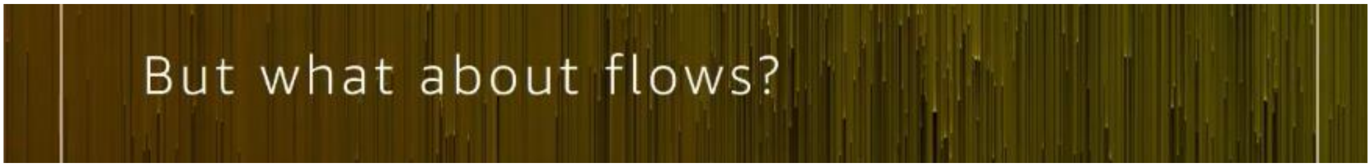
The ***mapping service*** helps us know which physical device to send a packet to. A mapping service is a distributed service that is in charge of mapping what instance and what physical host a particular ENI in any VPC ***currently*** maps to, it is a gigantic table that holds the maps data. the embedded routers in each physical host are talking to the mapping service to know what the real physical destination is for any virtual IP destination they have. It uses a simple network protocol refined for fast lookups.



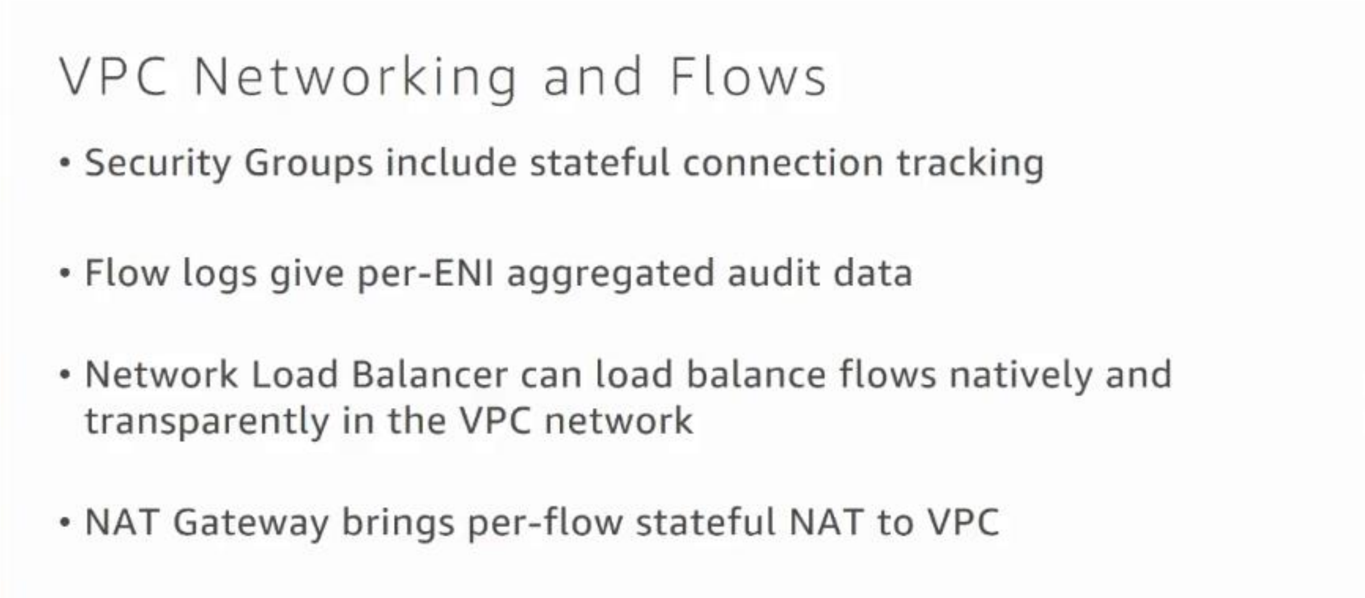
The mapping service

- A distributed web service that handles mappings between customers VPC routes and IPs and physical destinations on the wire.
- To support microsecond-scale latencies, mappings are cached where they are used, and pro-actively invalidated when they change.

Mappings are preloaded, pushed-out, and use several mapping patterns to make the lookups easier and faster for a distributed environment when an instance needs to send out packets to outside destinations.



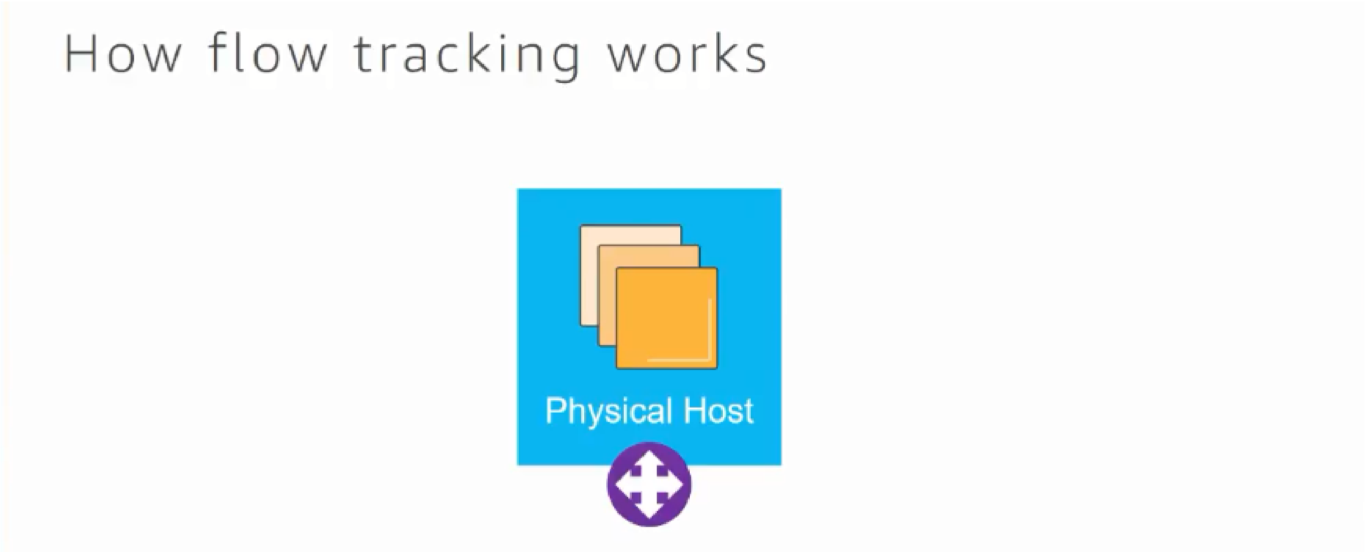
But what about flows?



VPC Networking and Flows

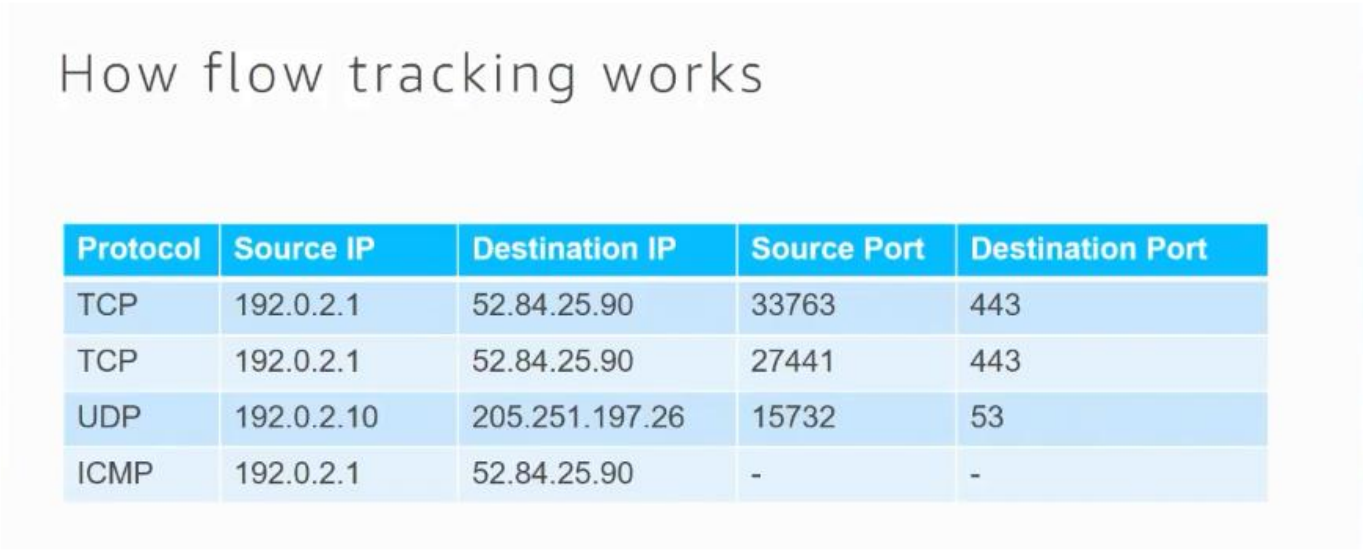
- Security Groups include stateful connection tracking
- Flow logs give per-ENI aggregated audit data
- Network Load Balancer can load balance flows natively and transparently in the VPC network
- NAT Gateway brings per-flow stateful NAT to VPC

Security Groups are really just firewall rules that include stateful connection tracking. There are flow traffic going on within the VPC network as shown above



How flow tracking works

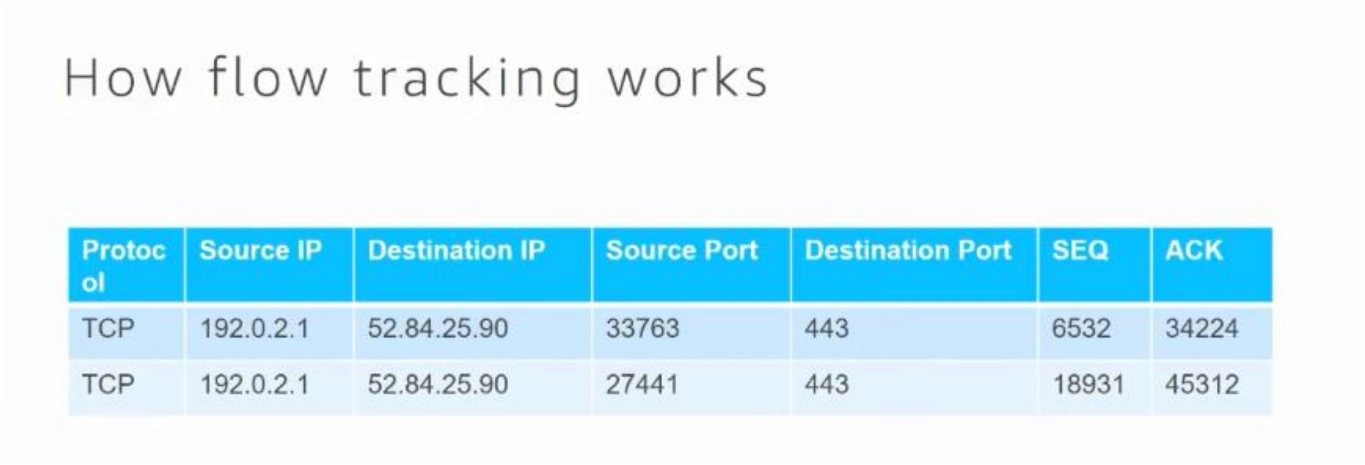
A lot of the flow tracking is actually happening on the embedded router attached to the physical host machine,



How flow tracking works

Protocol	Source IP	Destination IP	Source Port	Destination Port
TCP	192.0.2.1	52.84.25.90	33763	443
TCP	192.0.2.1	52.84.25.90	27441	443
UDP	192.0.2.10	205.251.197.26	15732	53
ICMP	192.0.2.1	52.84.25.90	-	-

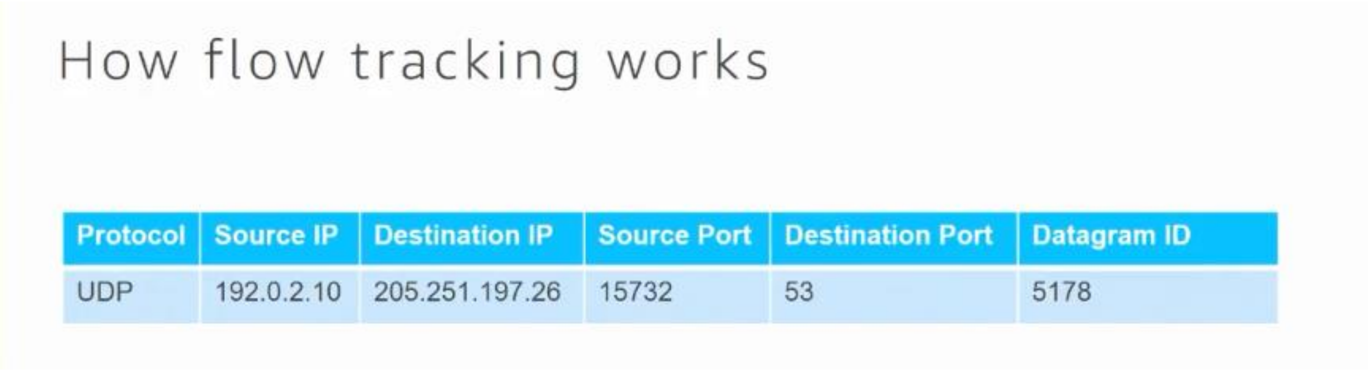
We consider a flow to be a 5-tuple, it is a combination of a Protocol (TCP, UDP, ICMP, etc.), Source IP, Destination IP, Source Port, and Destination Port that are in the packet information.



How flow tracking works

Protocol	Source IP	Destination IP	Source Port	Destination Port	SEQ	ACK
TCP	192.0.2.1	52.84.25.90	33763	443	6532	34224
TCP	192.0.2.1	52.84.25.90	27441	443	18931	45312

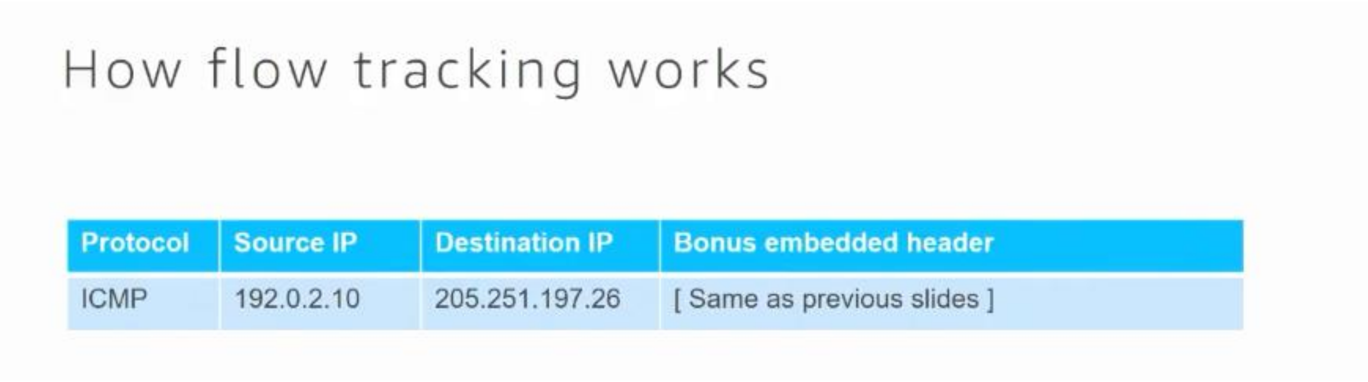
For TCP flow tracking, the 5-tuple has to be completely unique, so that packets cannot be mixed up and confused with each other even on the same instance. We are checking that the SEQ and ACK numbers are within TCP range during communication



How flow tracking works

Protocol	Source IP	Destination IP	Source Port	Destination Port	Datagram ID
UDP	192.0.2.10	205.251.197.26	15732	53	5178

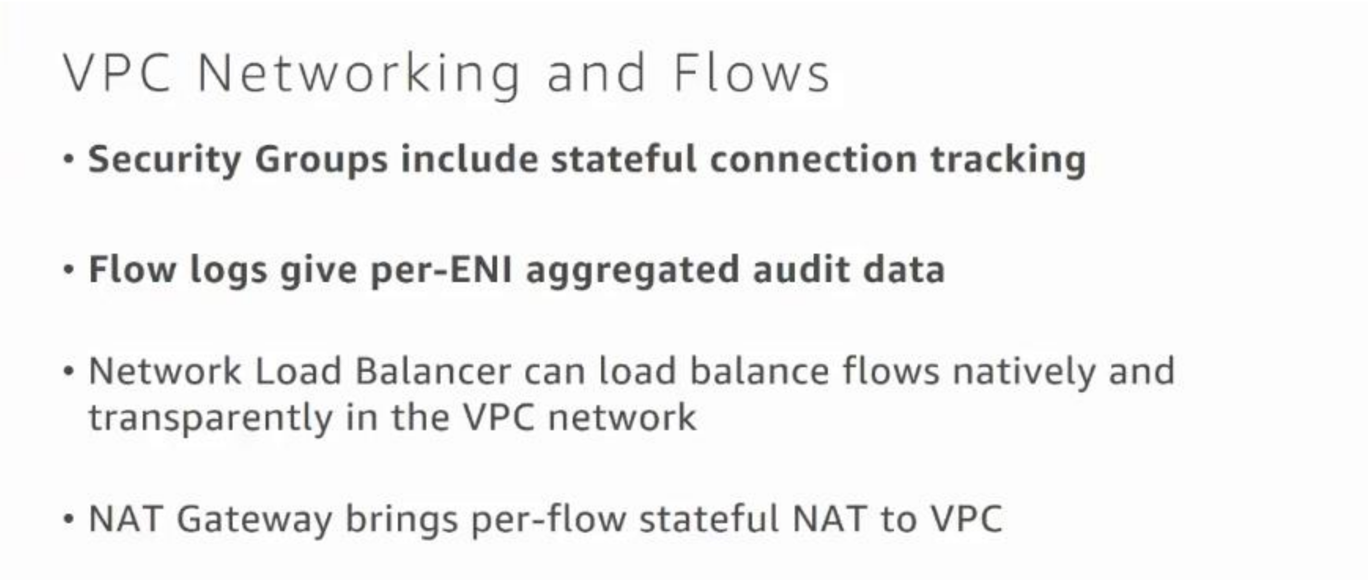
UDP flow tracking uses Datagram IDs for packets that are being too large and splitted for reassembling back from the multiple fragments and filtering out fake fragments



How flow tracking works

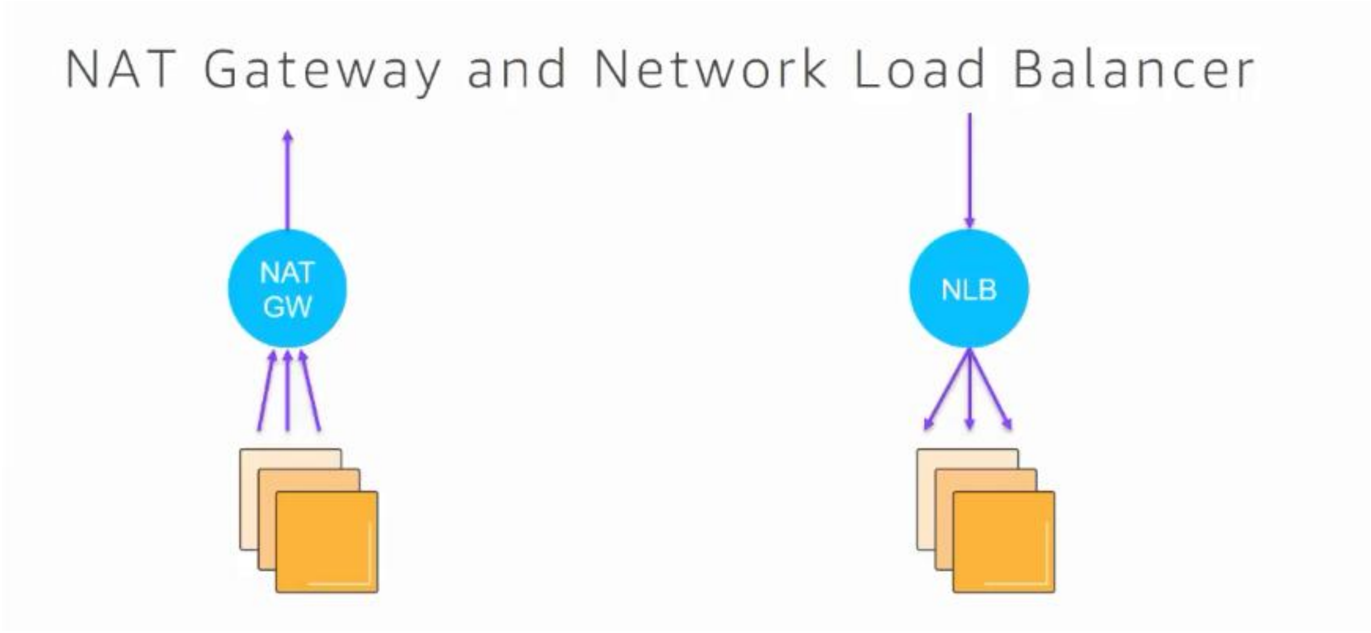
Protocol	Source IP	Destination IP	Bonus embedded header
ICMP	192.0.2.10	205.251.197.26	[Same as previous slides]

ICMP (Internet Control Messaging Protocol) is mostly used for signaling errors related to connections and for sending pings as ICMP echo for getting back an echo reply.

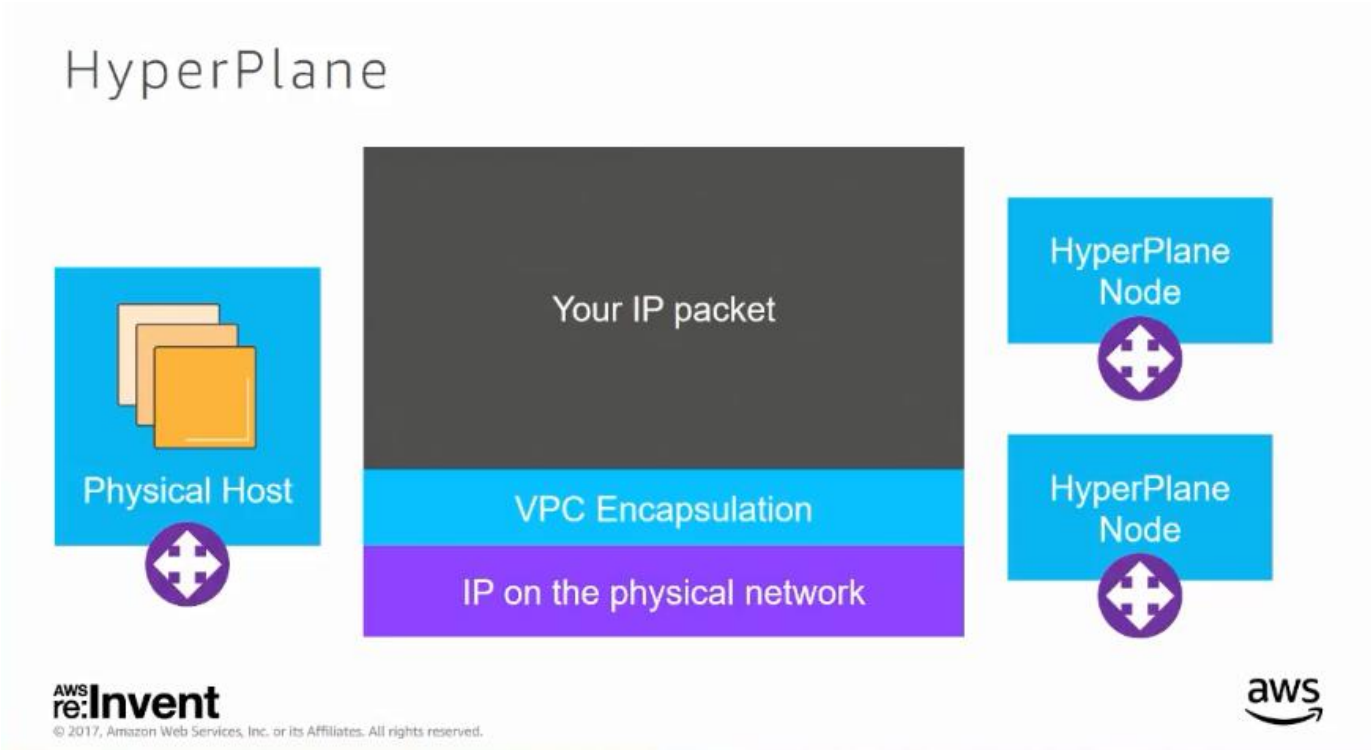


VPC Networking and Flows

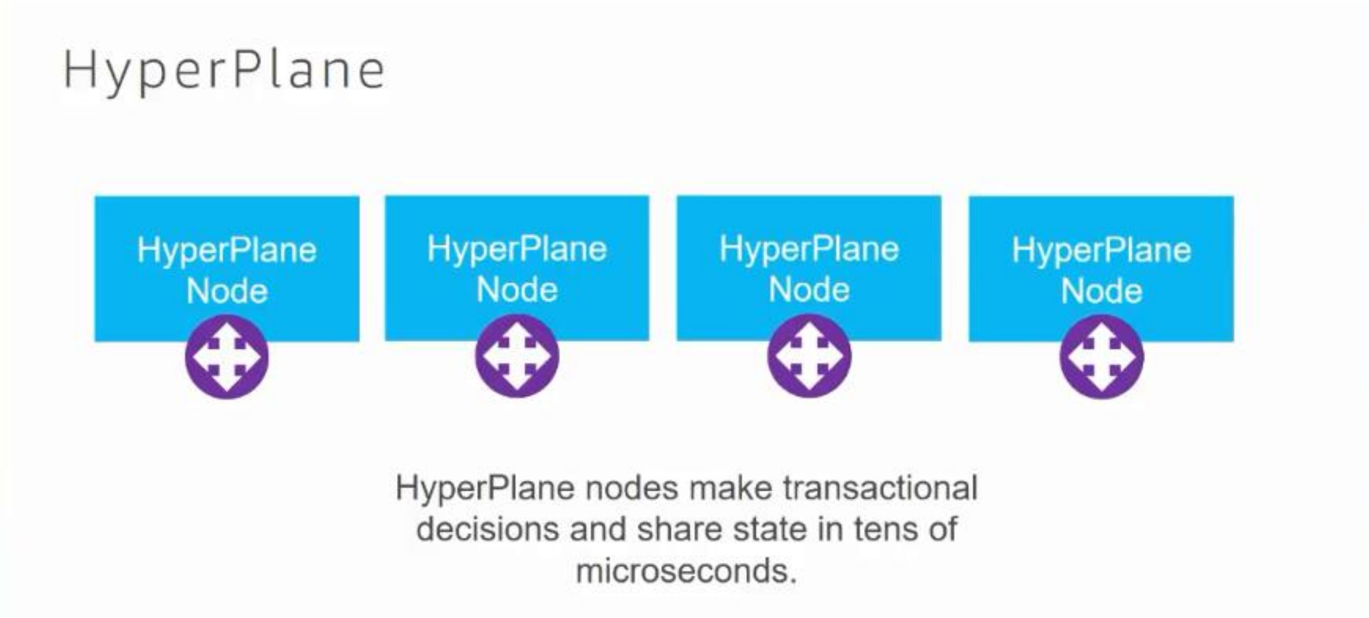
- **Security Groups include stateful connection tracking**
- **Flow logs give per-ENI aggregated audit data**
- Network Load Balancer can load balance flows natively and transparently in the VPC network
- NAT Gateway brings per-flow stateful NAT to VPC



NAT Gateway and NLB (Network Load Balancer) are very similar, they are really about letting many instances share an IP address. They are really the same problem and we built a system called Hyperplane for this as seen below



Hyperplane Nodes are normal EC2 instances that are also on the VPC network along with the VPC’s physical hosts. The Hyperplane nodes are accepting traffic and essentially doing state tracking, packet routing, etc. while still be on the EC2 network too.



HyperPlane

HyperPlane Node

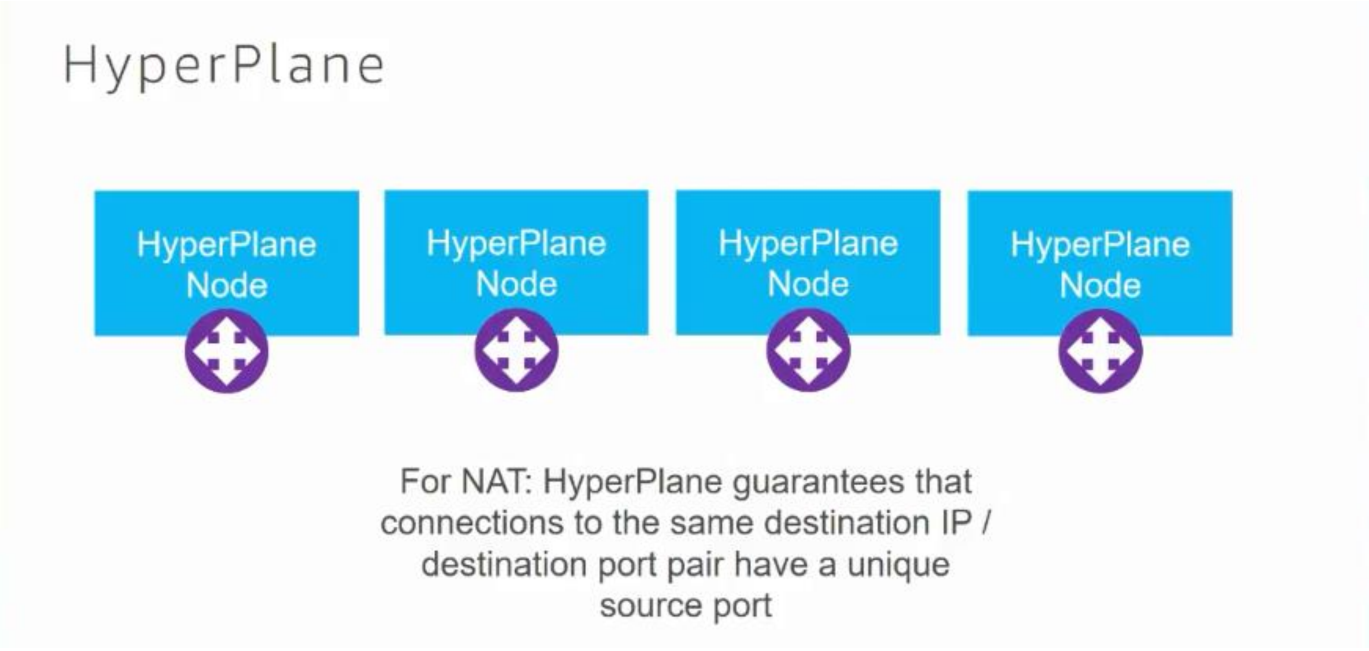
HyperPlane Node

HyperPlane Node

HyperPlane Node

HyperPlane nodes make transactional decisions and share state in tens of microseconds.

Hyperplane is a distributed system. When traffic comes to a Hyperplane node, the nodes make transactional decisions about how to handle any incoming or outgoing packets for flows.



HyperPlane

HyperPlane Node

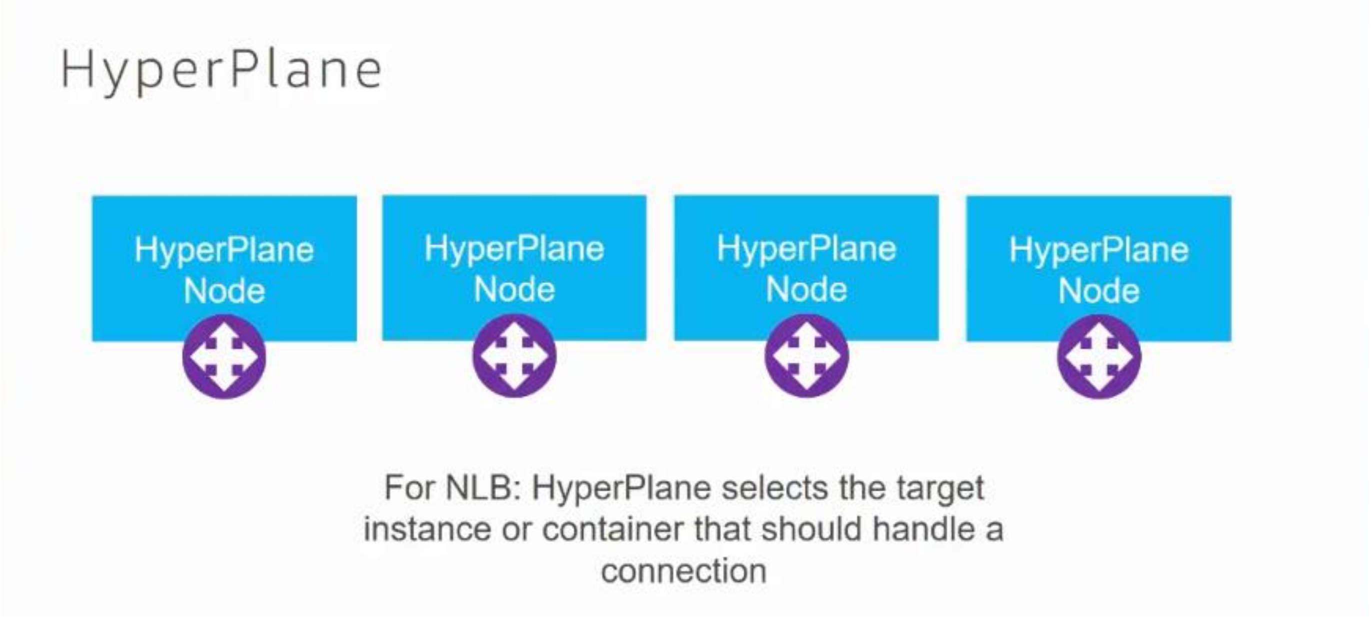
HyperPlane Node

HyperPlane Node

HyperPlane Node

For NAT: HyperPlane guarantees that connections to the same destination IP / destination port pair have a unique source port

This is simply a ***Distributed Consensus Problem*** that needs to guarantee uniqueness using the source or destination ports on a physical host in a very short period of time.



HyperPlane

HyperPlane Node

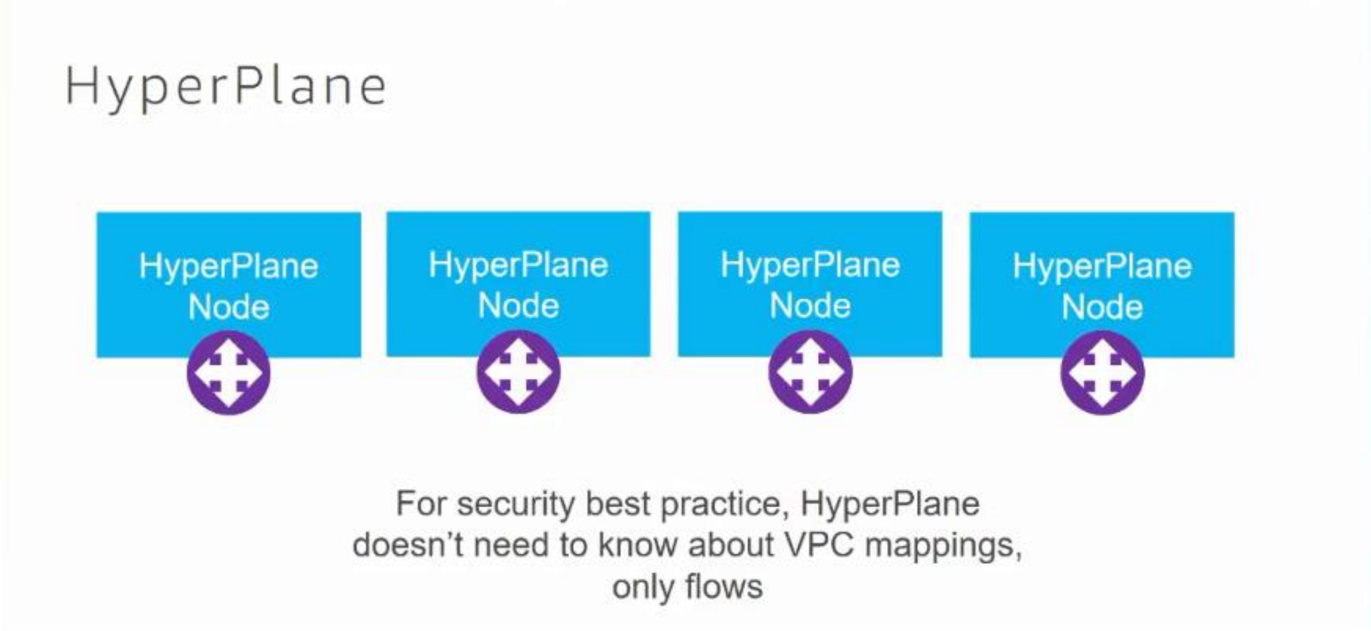
HyperPlane Node

HyperPlane Node

HyperPlane Node

For NLB: HyperPlane selects the target instance or container that should handle a connection

Hyperplane does this by keeping little state using memory about what ports certain packets were directed to



HyperPlane

HyperPlane Node

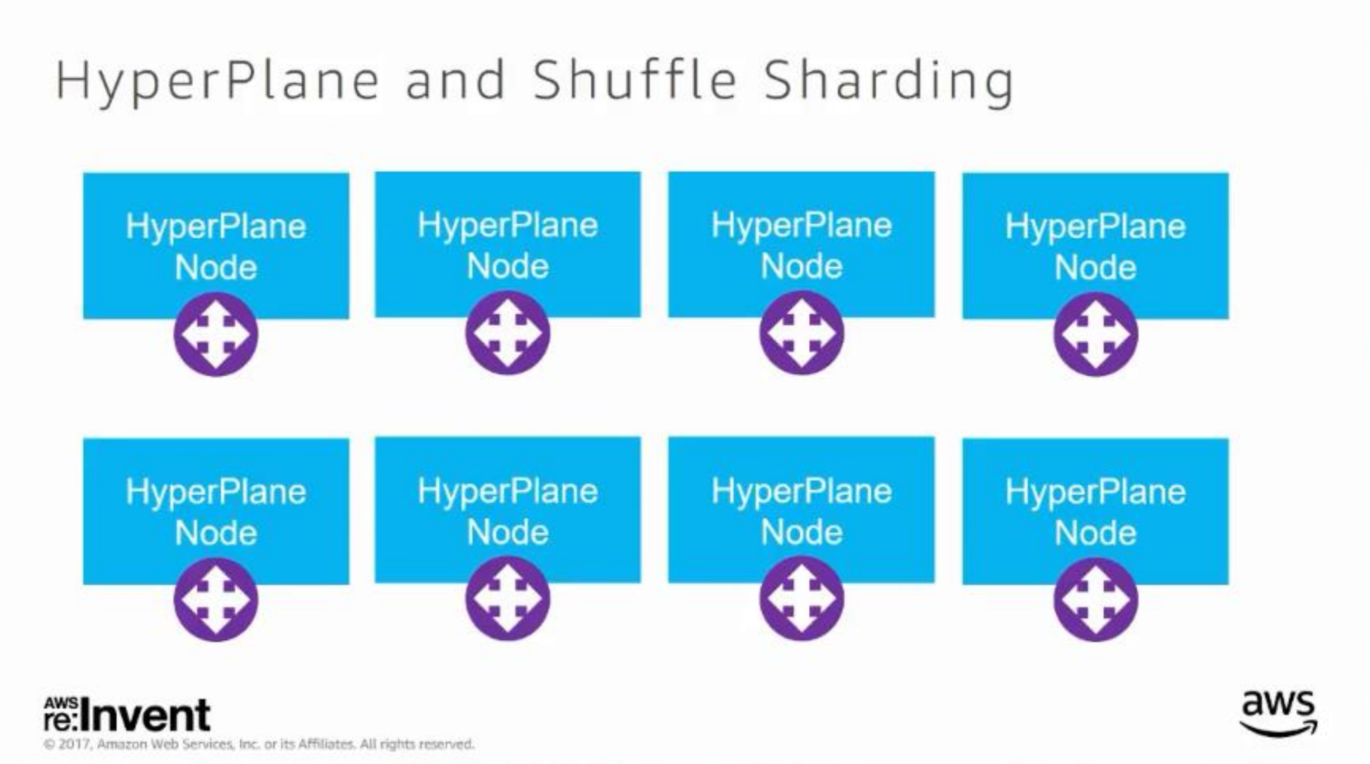
HyperPlane Node

HyperPlane Node

HyperPlane Node

For security best practice, HyperPlane doesn't need to know about VPC mappings, only flows

Hyperplane only know about the Hyperplane resources and not mappings data. Hyperplane nodes are embedded in the network and the network is multi-tenant and built for very large networks. We need to take care of the noisy neighbor problem in multi-tenancy



HyperPlane and Shuffle Sharding

HyperPlane Node

HyperPlane Node

HyperPlane Node

HyperPlane Node

HyperPlane Node

HyperPlane Node

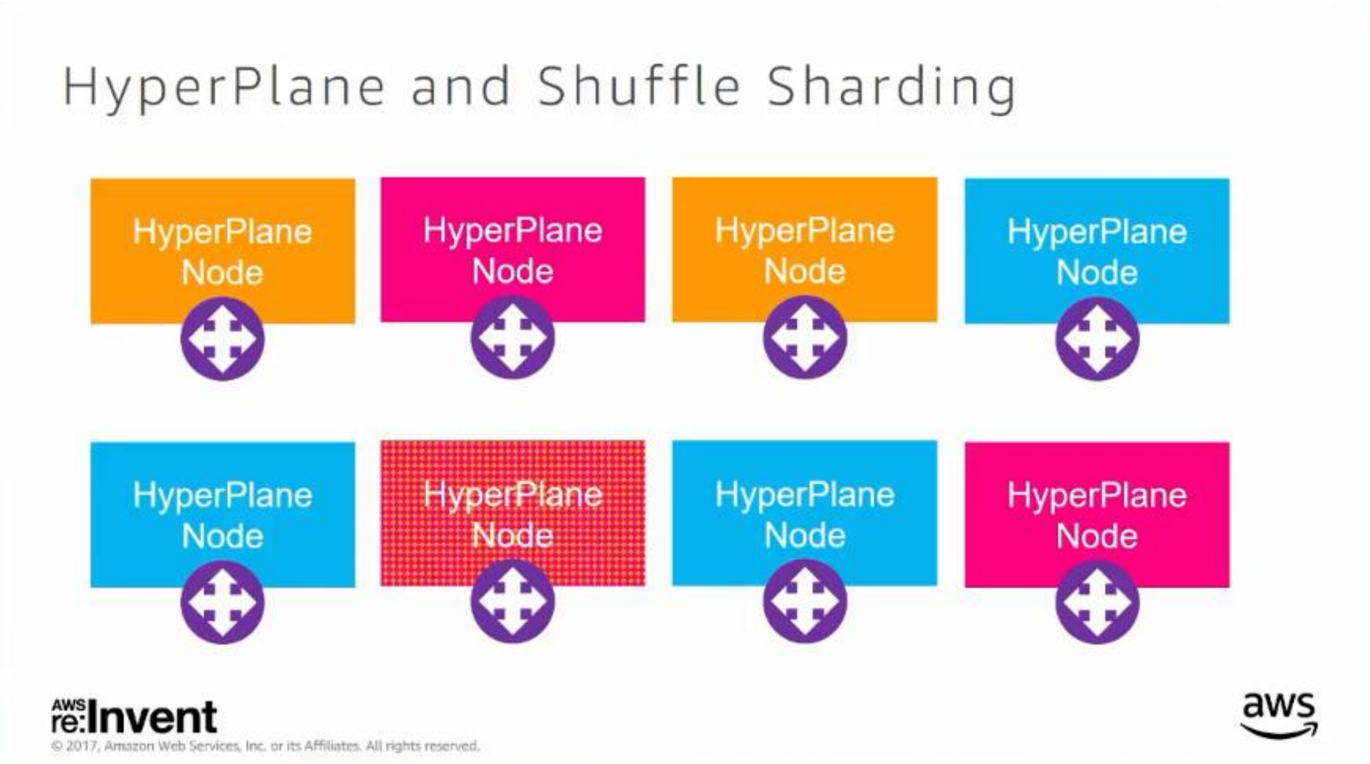
HyperPlane Node

HyperPlane Node

AWS re:Invent

aws

A solution is to allow some customers to use only certain Hyperplane nodes as above, but this is susceptible to the noisy neighbor problem if one customer is doing lots of work and loading up all Hyperplane nodes. Another atrategy is to shard and restrict customers to certain nodes only.



HyperPlane and Shuffle Sharding

HyperPlane Node

HyperPlane Node

HyperPlane Node

HyperPlane Node

HyperPlane Node

HyperPlane Node

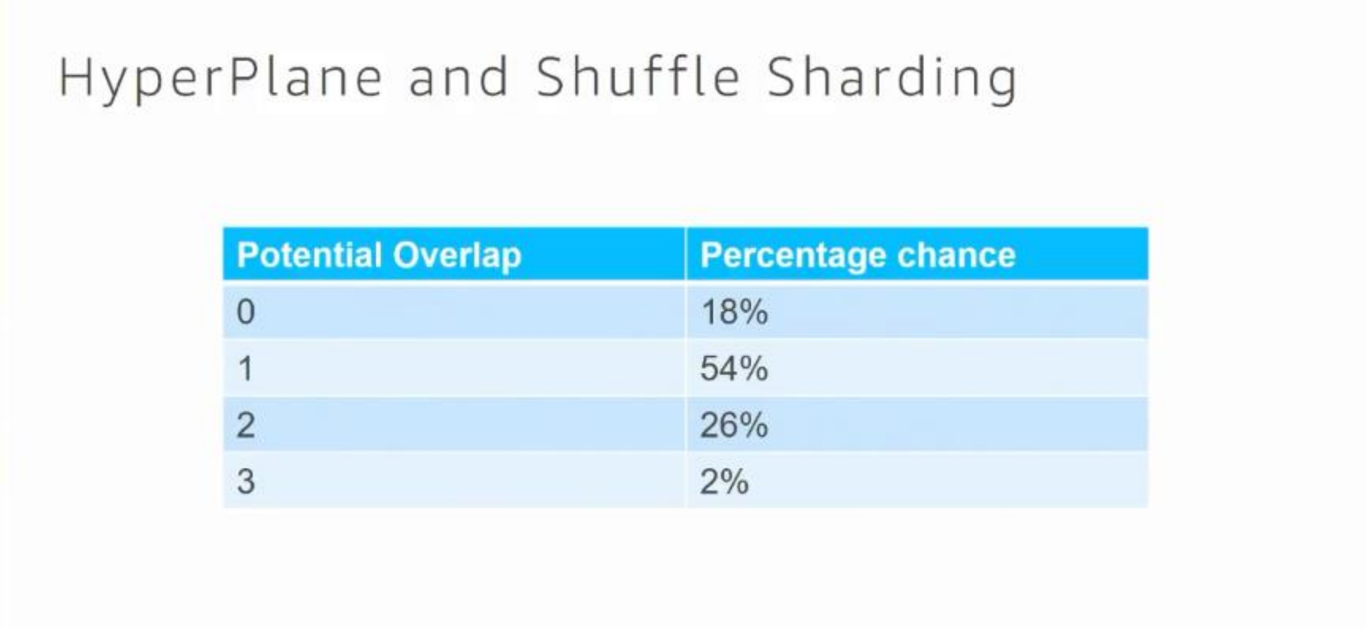
HyperPlane Node

HyperPlane Node

AWS re:Invent

aws

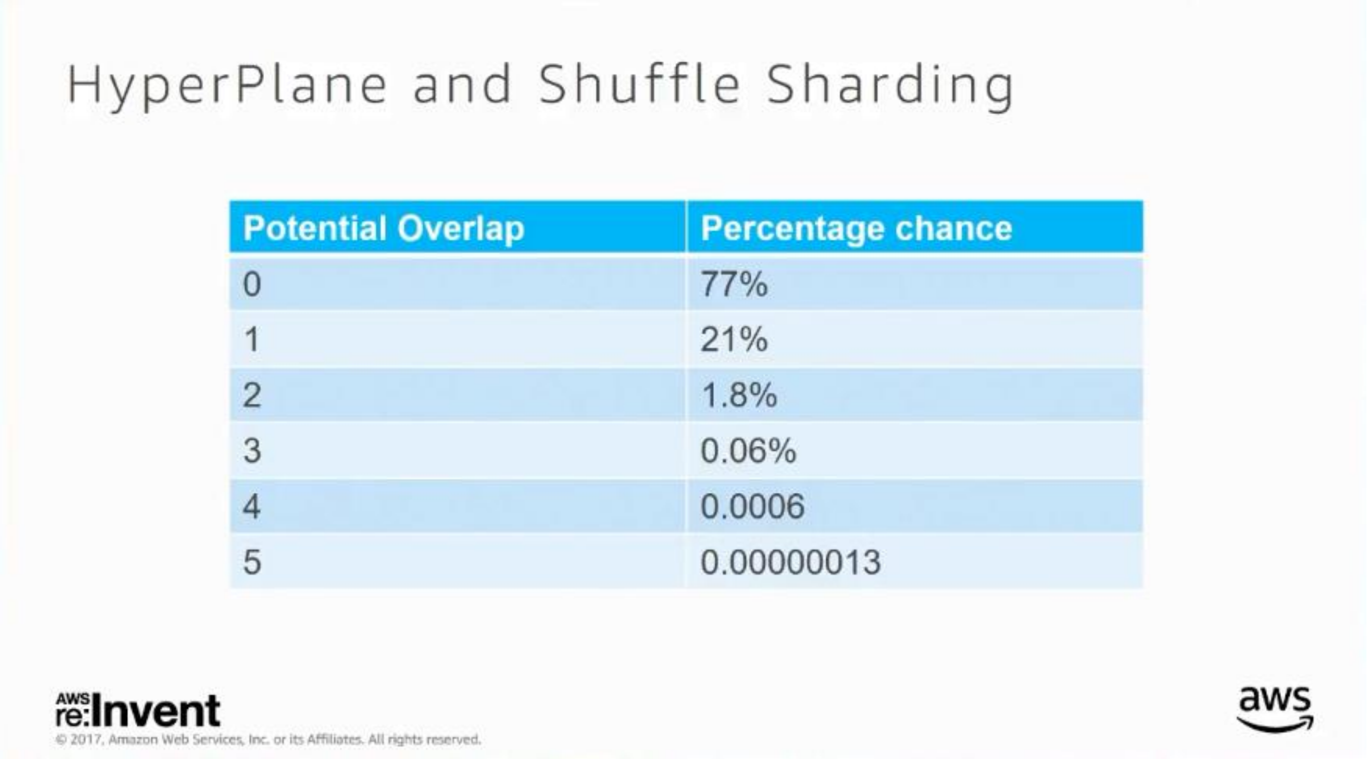
This extends sharding probabilistically.



HyperPlane and Shuffle Sharding

Potential Overlap	Percentage chance
0	18%
1	54%
2	26%
3	2%

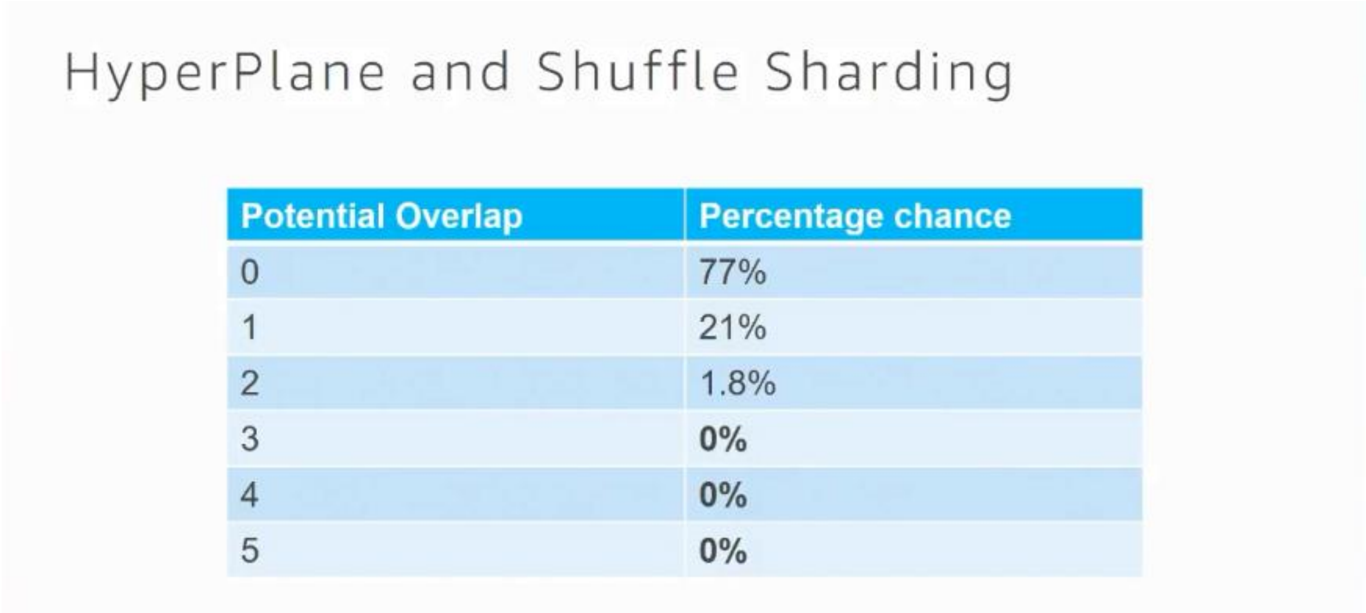
With 8 nodes and choosing 3 nodes per customer sharding, there is an 18% chance that there would be 0 customer overlaps between any 2 random customers, customer A and customer B.



HyperPlane and Shuffle Sharding

Potential Overlap	Percentage chance
0	77%
1	21%
2	1.8%
3	0.06%
4	0.0006
5	0.00000013

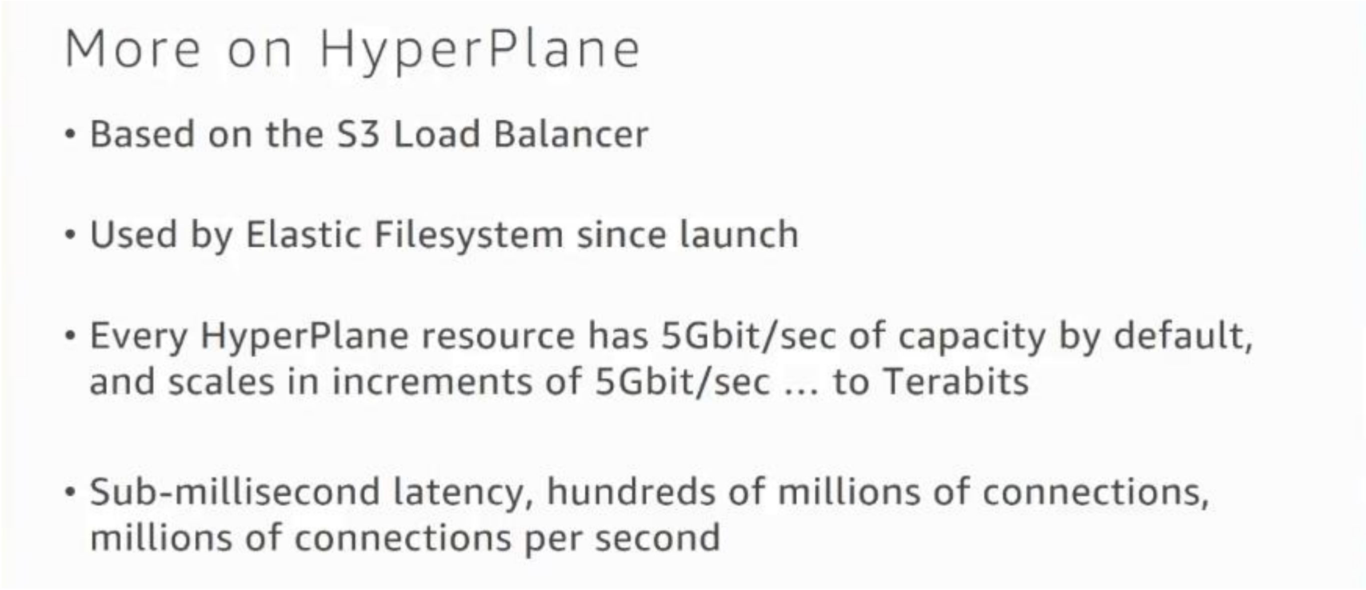
This model has 100 Hyperplane nodes and we give each customer 5 nodes to use. This is combinatorial assignment.



HyperPlane and Shuffle Sharding

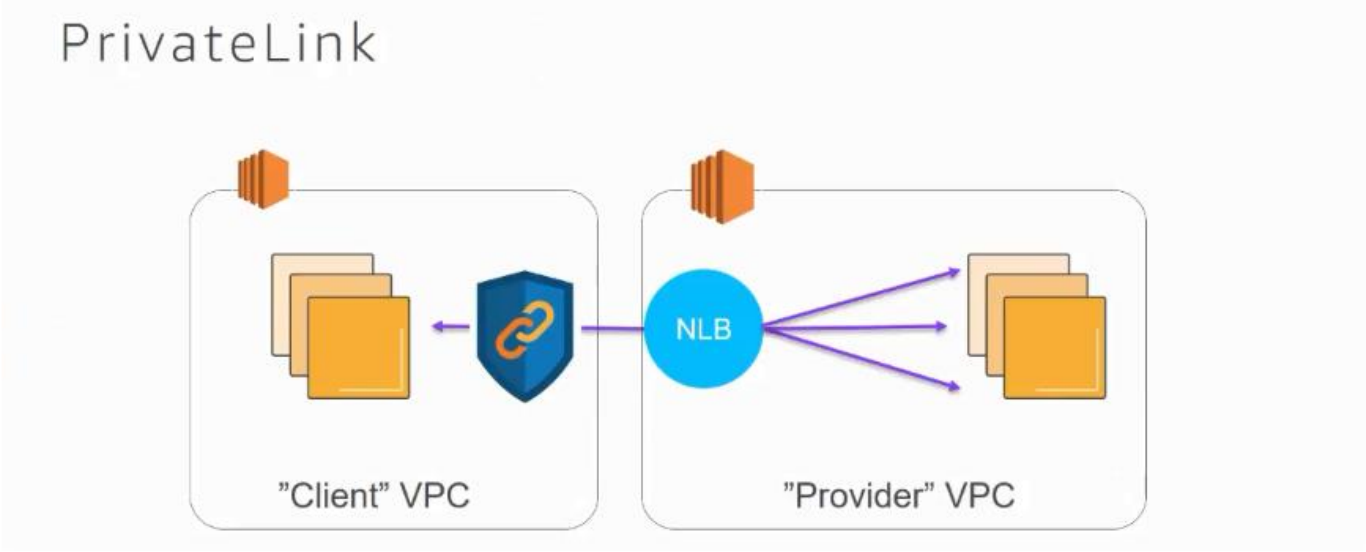
Potential Overlap	Percentage chance
0	77%
1	21%
2	1.8%
3	**0%**
4	**0%**
5	**0%**

This provides resiliency guarantees even when we lose Hyperplane nodes when EC2 instance dies.



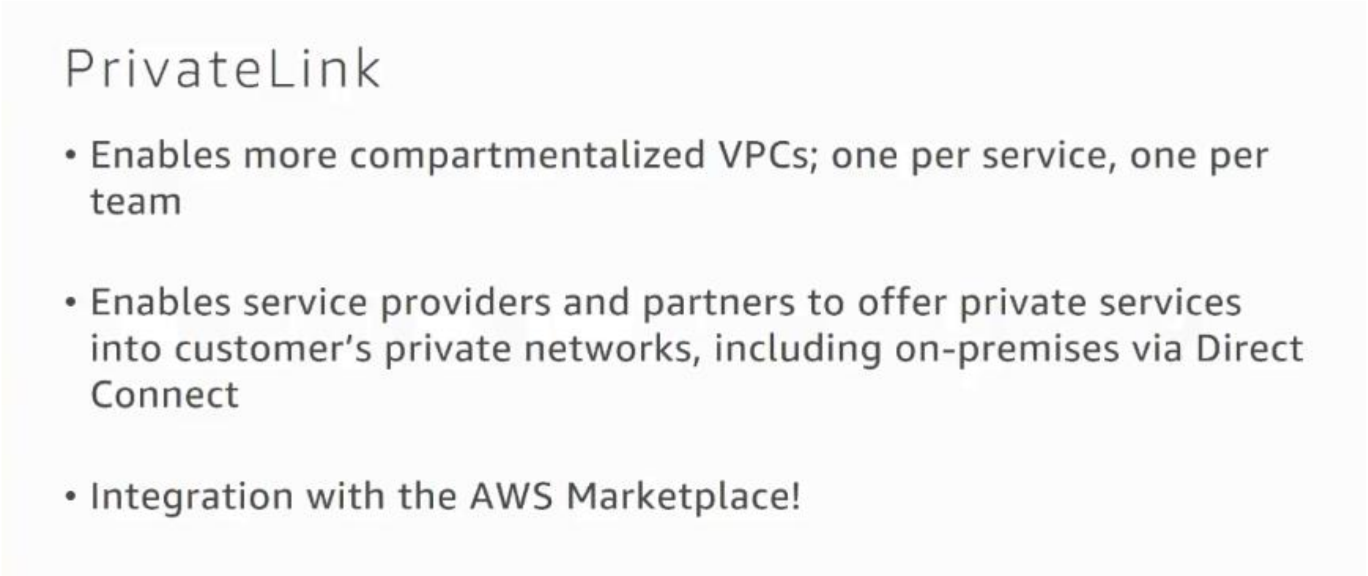
More on HyperPlane

- Based on the S3 Load Balancer
- Used by Elastic Filesystem since launch
- Every HyperPlane resource has 5Gbit/sec of capacity by default, and scales in increments of 5Gbit/sec ... to Terabits
- Sub-millisecond latency, hundreds of millions of connections, millions of connections per second



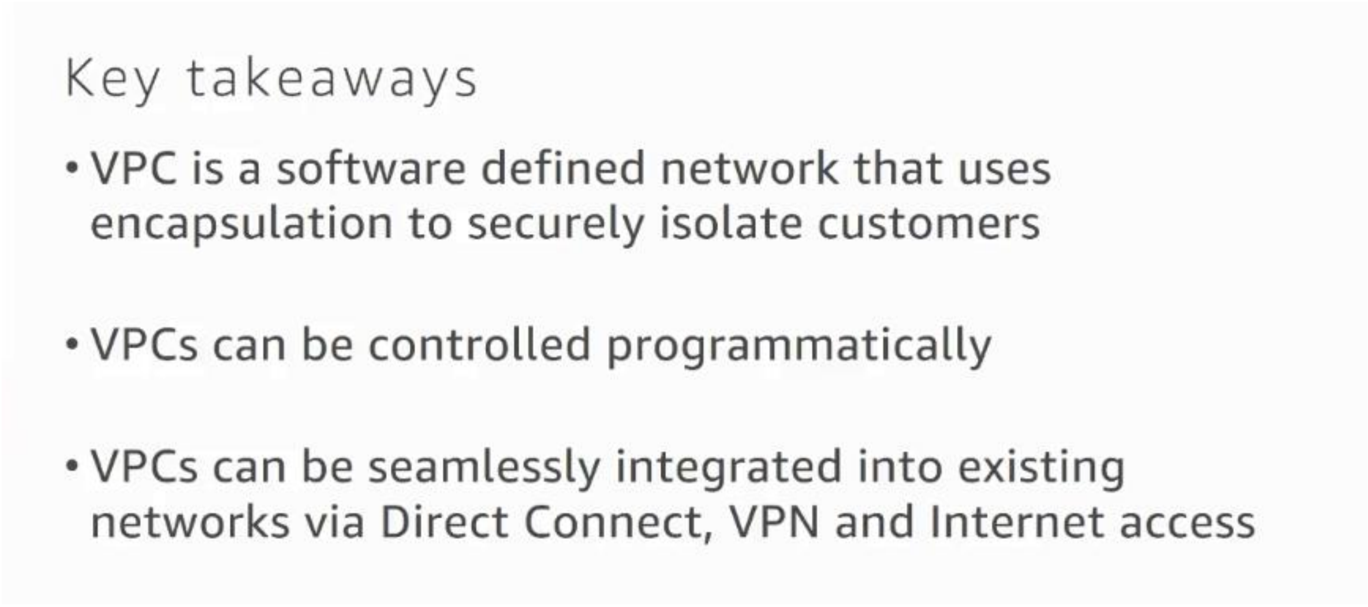
PrivateLink

This is stateful tracking between VPCs using Hyperplanes



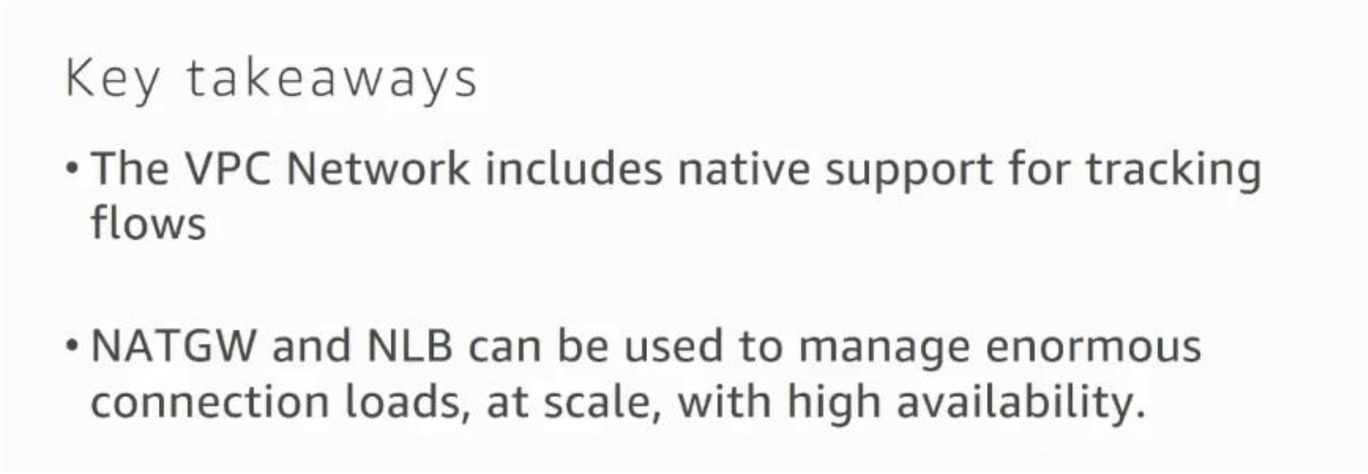
PrivateLink

- Enables more compartmentalized VPCs; one per service, one per team
- Enables service providers and partners to offer private services into customer's private networks, including on-premises via Direct Connect
- Integration with the AWS Marketplace!



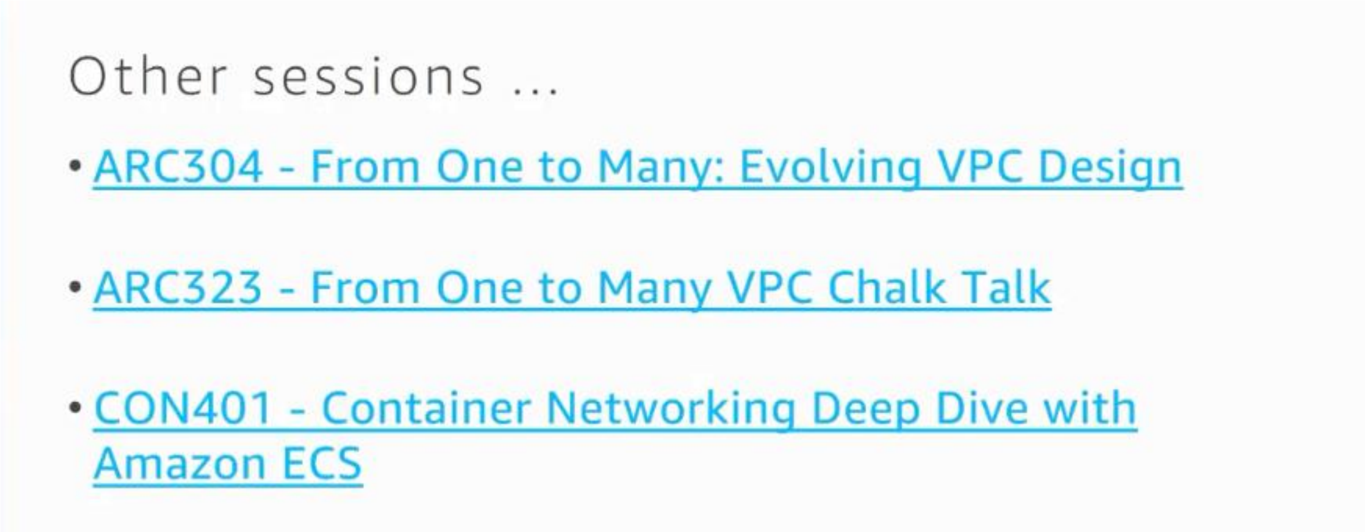
Key takeaways

- VPC is a software defined network that uses encapsulation to securely isolate customers
- VPCs can be controlled programmatically
- VPCs can be seamlessly integrated into existing networks via Direct Connect, VPN and Internet access



Key takeaways

- The VPC Network includes native support for tracking flows
- NATGW and NLB can be used to manage enormous connection loads, at scale, with high availability.



Other sessions …

- ARC304 - From One to Many: Evolving VPC Design
- ARC323 - From One to Many VPC Chalk Talk
- CON401 - Container Networking Deep Dive with Amazon ECS

AWS re:Invent

Thank you!

AWS re:Invent

aws