

No Packet Left Behind

AWS' journey to running its own hardware and software end-to-end across a global network

Lincoln Dale

Senior Principal Engineer

AWS – AS16509



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Agenda

AWS Global Infrastructure

Our journey to reinventing our network infrastructure

our hardware, software and how we put systems together

Network architecture and software, tools and controllers

How we build and automate our network



AWS Global Infrastructure



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Global Infrastructure

AWS REGIONS, EDGE LOCATIONS, AND THE GLOBAL BACKBONE



AWS Global Infrastructure

AWS REGIONS, EDGE LOCATIONS, AND THE GLOBAL BACKBONE



REGIONAL EXPANSION

- Available Today: 31 Regions
- Coming soon 5 Regions



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Global Infrastructure

AWS REGIONS, EDGE LOCATIONS, AND THE GLOBAL BACKBONE



LOCAL ZONES

- 4 recently launched
- 21 available today
- 30 coming soon



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Global Infrastructure

AWS REGIONS, EDGE LOCATIONS, AND THE GLOBAL BACKBONE



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Global Infrastructure

AWS REGIONS, EDGE LOCATIONS, AND THE GLOBAL BACKBONE



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

AWS Global Infrastructure

AWS REGIONS, EDGE LOCATIONS, AND THE GLOBAL BACKBONE

AWS NETWORK BACKBONE

- Redundant 400 Gbps links
- 245+ Countries & Territories
- Between all Regions, Local Zones, and Edge Locations



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.



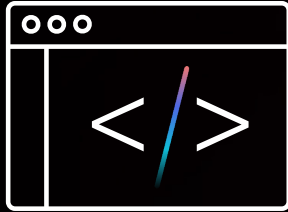
prime video

**THURSDAY
NIGHT
FOOTBALL**

Reinventing our network infrastructure



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

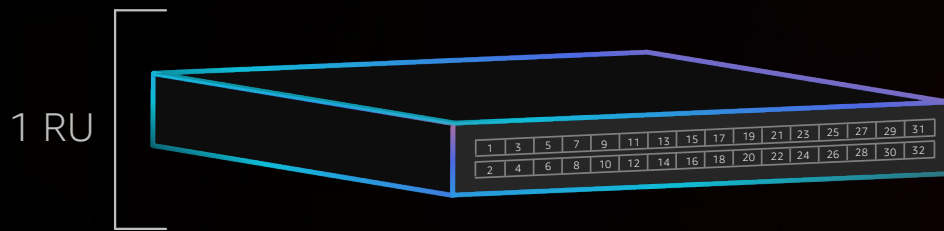


CUSTOM SOFTWARE



CUSTOM HARDWARE

- Simplicity Scales
- Focus on the benefits
- Freedom to examine trade-offs



12.8

TERABITS PER SECOND

DEVICE: 1 x Switch

HEIGHT: 1 x Rack Unit (RU)

PORTS: 32 x 400G



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

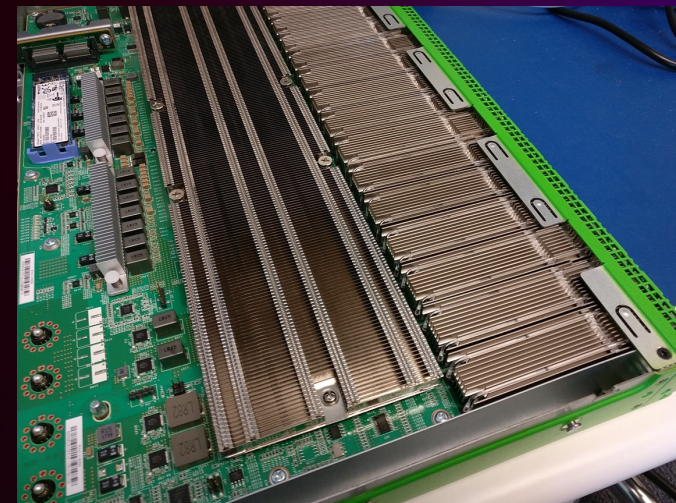
12.8

TERABITS PER SECOND

DEVICE: 1 x Switch

HEIGHT: 1 x Rack Unit (RU)

PORTS: 32 x 400G



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

42 RU



100

TERABITS PER SECOND

DEVICE: 1 rack (32 x switches)

HEIGHT: 42 x Rack Unit (RU)

PORTS: 32 x 32 x 400G



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

100

TERABITS PER SECOND

DEVICE: 1 rack (32 x switches)

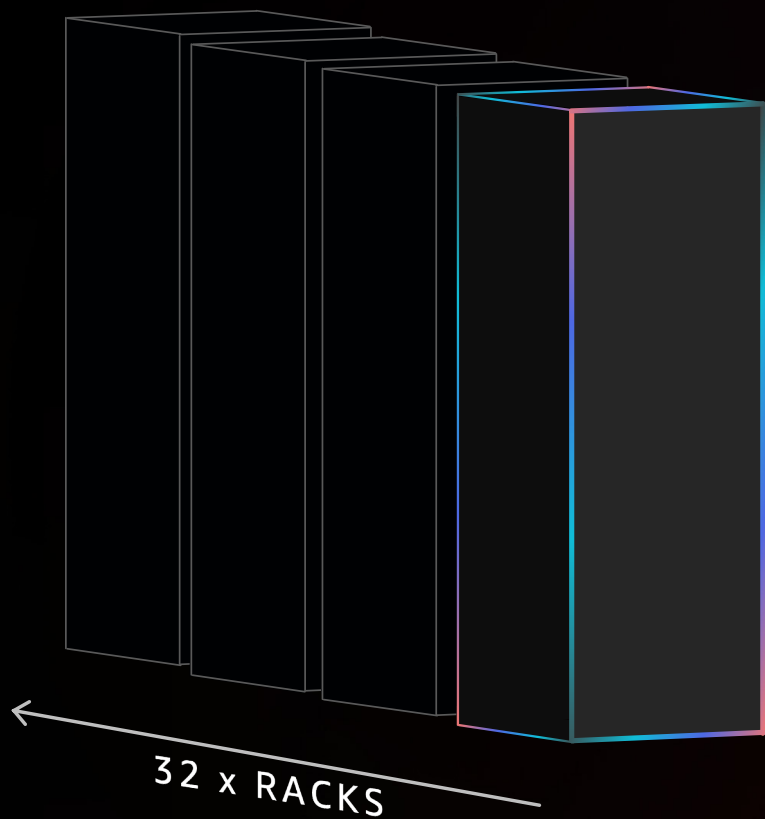
HEIGHT: 42 x Rack Unit (RU)

PORTS: 32 x 32 x 400G



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.





3,200

TERABITS PER SECOND

DEVICE: 32 racks (32 x switches)

HEIGHT: 42 x Rack Unit (RU)

THROUGHPUT/RACK: 100 Tbps



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

How we do it – In rack

Direct-attach copper (DAC) cabling

100G 6.7mm OD at 2.5m

400G 11mm OD at 2.5m

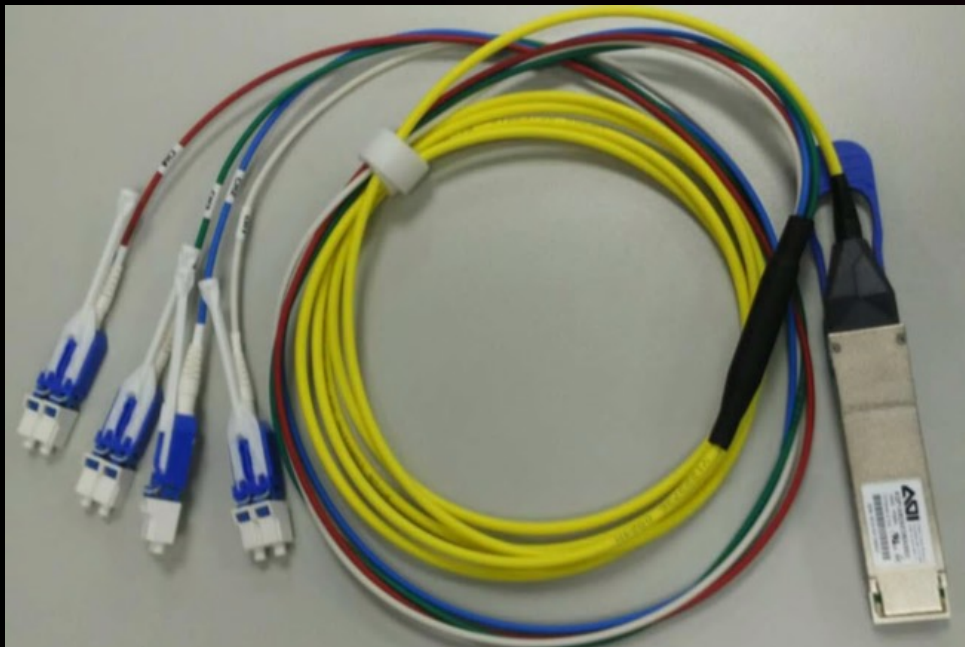
Our Biggest enemy? Cable diameter.

Active DAC with retimers to reduce cable area

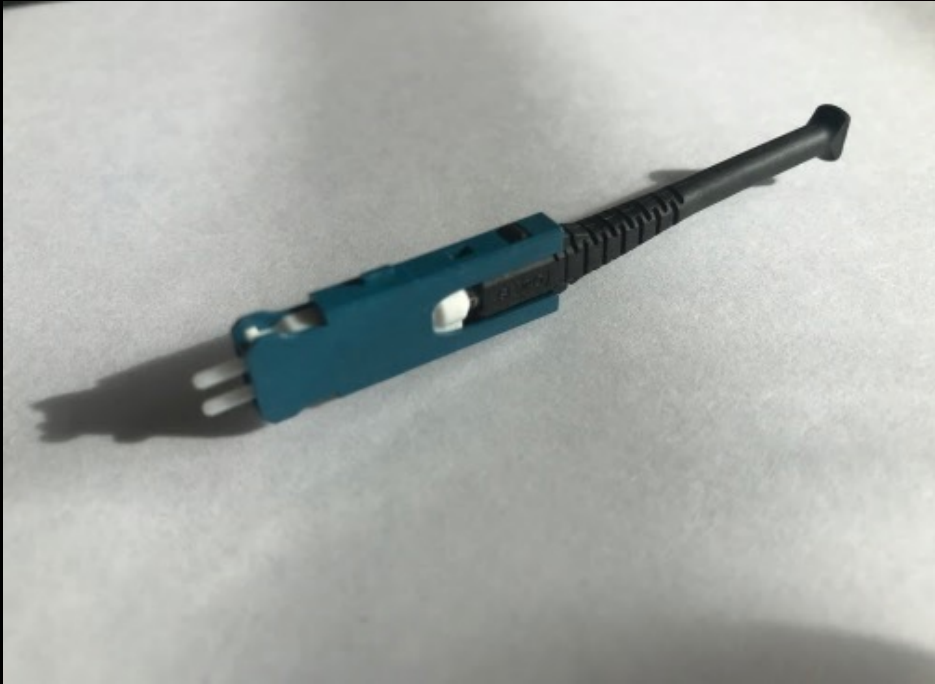




How we do it – Short reach



How we do it – SN connector



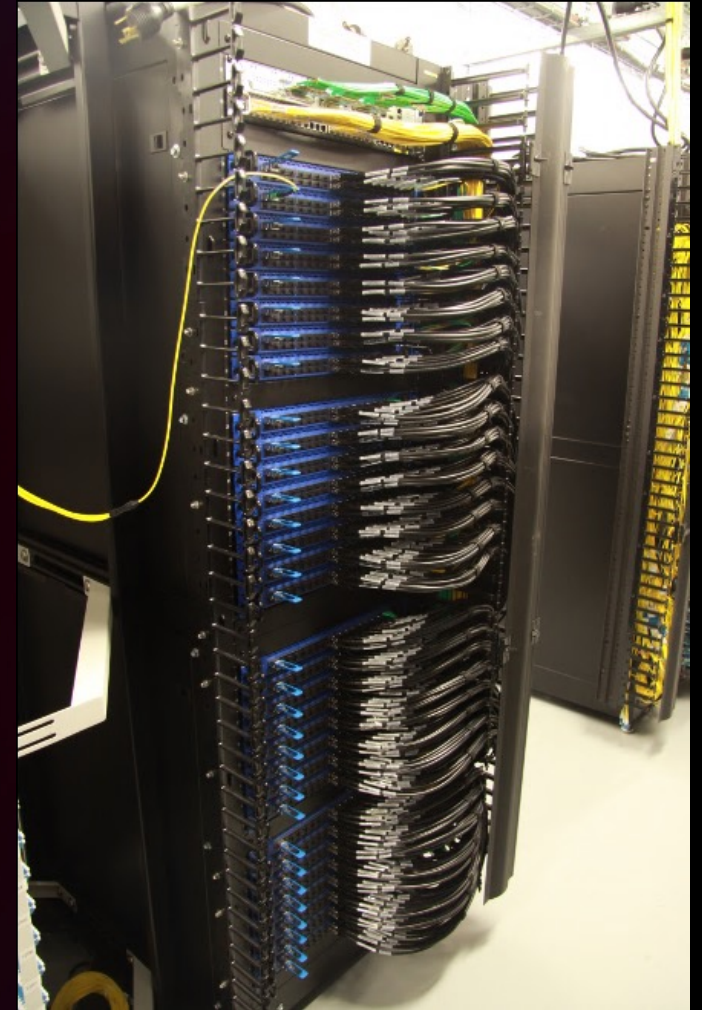
Metal boxes and a lot of cables

Small number of rack variations

Rack and cable switches for burn-in

Collect inventory and compare with bill of materials

Reprogram with AWS controlled binaries



Network Architecture and Software



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Automate everything

Config generation

Deployment coordination

Active telemetry

Auto-remediation

NOC-less

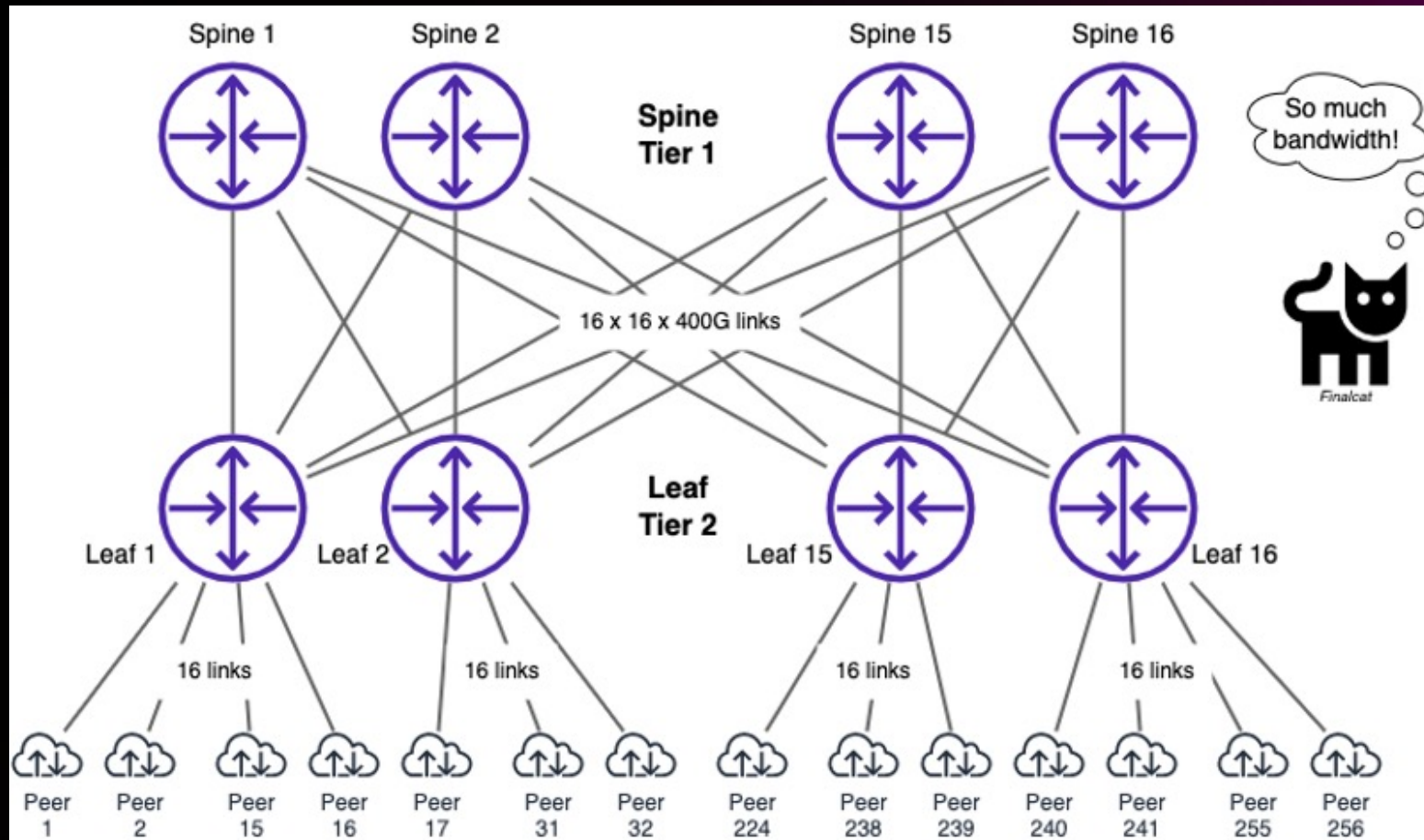


© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.



2 tier Clos

NON OVERSUBSCRIBED ANY PORT TO ANY PORT



How we do it

MEDIUM HAUL

Data center interconnect (DCI)

OIF 400G ZR

400G – ZR+ to 400km,
Bright ZR over 1000km

Integrated routing, DWDM, encryption



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

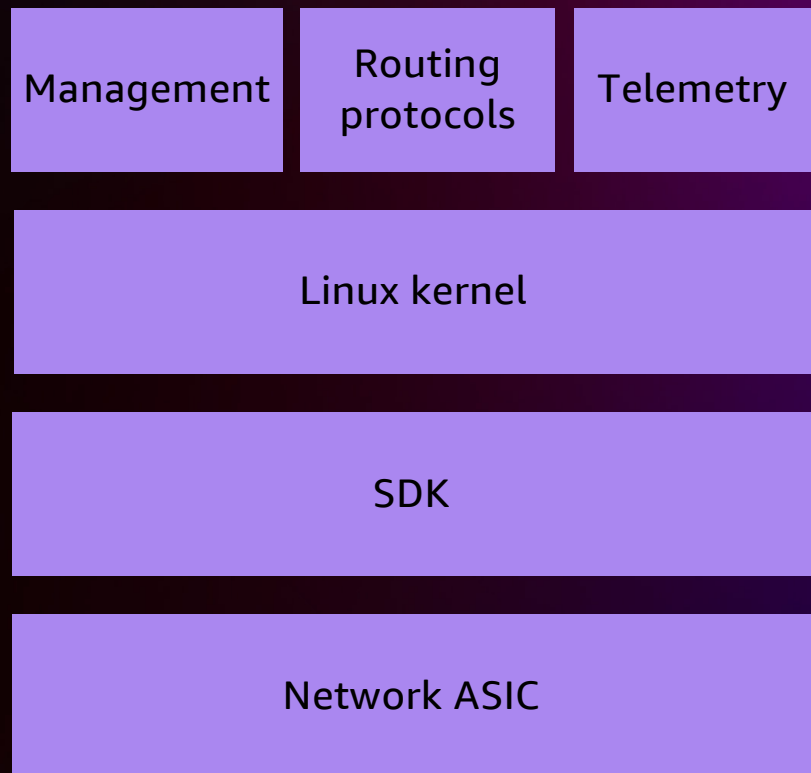
Create

NETWORK OPERATING SYSTEM

Linux-based

Multi-sourced manufacturing

Multi-ASIC



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Create

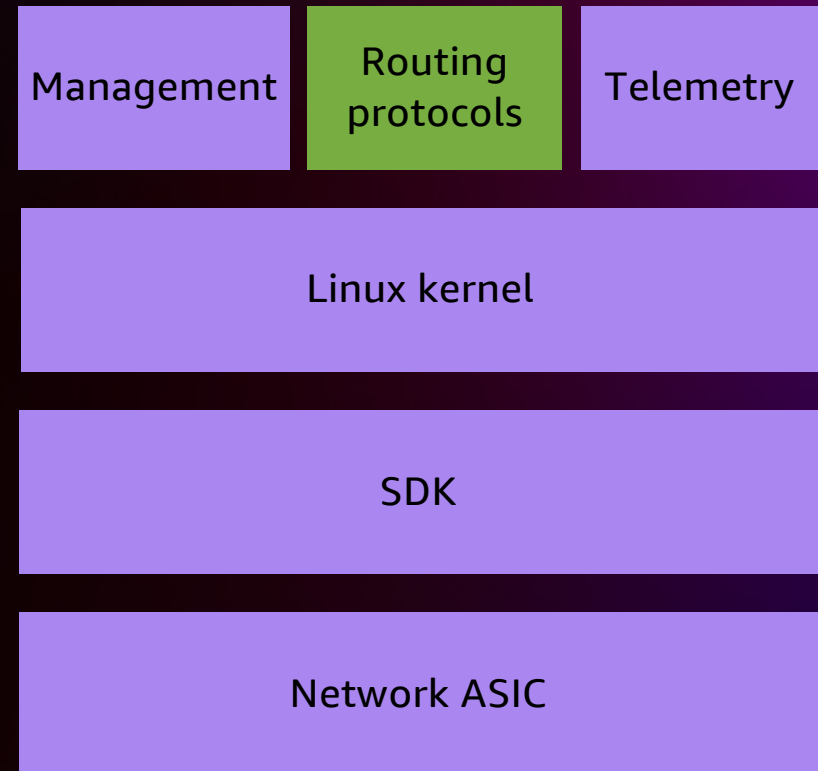
NETWORK OPERATING SYSTEM

Linux-based

Multi-sourced manufacturing

Multi-ASIC

OSPF/BGP ++



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Traceroute from outside AWS

```
% traceroute www.amazon.com
```

```
...  
5  * * *  
6  * * *  
7  52.93.33.77 (52.93.33.77)  1.984 ms 1.789 ms 1.983 ms  
8  52.93.33.130 (52.93.33.130)  2.316 ms 2.362 ms 2.891 ms  
9  150.222.72.105 (150.222.72.105)  3.682 ms 3.044 ms 3.002 ms  
10 * * *  
11 * * *  
12 * * *  
13 * * *  
14 * * *  
15 server-65-8-32-17.mel50.r.cloudfront.net (65.8.32.17)  3.650 ms 4.866 ms 3.033 ms
```



Traceroute from inside AWS

news.ycombinator.com/item?id=32566730

Hacker News new | past | comments | ask | show | jobs | submit

▲ Amazon, Verizon found using IPv4 240/4 addresses (ripe.net)

193 points by dtaht on Aug 23, 2022 | hide | past | favorite | 143 comments

labs.ripe.net/author/qasim-lone



240/4 As Seen by RIPE Atlas



Qasim Lone — 23 Aug 2022

Contributors: John Gilmore, Seth David Schoen, Dave Täht, [Emile Aben](#)

[atlas](#) [research](#) [measurements](#) [internet number resources](#)

121 ❤️ 2 💬 🔗 📌

In this article we use data from RIPE Atlas probes to investigate the usage of 240/4, a block of IPv4 addresses 'reserved for future use', formally known as Class E in the wild.

Take a look
through NA

Traceroute to 8.8.8.8

```
1 * * *
2 * * *
3 * * *
4 10.117.52.85 (10.117.52.85) 0.000 ms
5 100.64.95.255 (100.64.95.255) 0.000 ms
6 240.1.240.32 (240.1.240.32) 0.000 ms
7 100.66.13.156 (100.66.13.156) 0.000 ms
8 240.1.236.24 (240.1.236.24) 0.000 ms
9 108.166.244.14 (108.166.244.14) 0.000 ms
10 108.166.244.18 (108.166.244.18) 0.000 ms
11 242.0.78.241 (242.0.78.241) 0.000 ms
12 242.0.90.89 (242.0.90.89) 0.000 ms
13 15.230.39.40 (15.230.39.40) 0.000 ms
14 15.230.140.117 (15.230.140.117) 0.000 ms
15 52.93.239.36 (52.93.239.36) 0.000 ms
16 100.100.6.57 (100.100.6.57) 14.555 ms 100.91.177.1 (100.91.177.1) 14.291 ms 100.91.177.27 (100.91.177.27) 15.858 ms
17 100.100.77.70 (100.100.77.70) 14.679 ms 100.100.92.72 (100.100.92.72) 14.630 ms 100.100.76.134 (100.100.76.134) 14.319 ms
18 100.100.69.163 (100.100.69.163) 14.312 ms 100.100.64.165 (100.100.64.165) 45.745 ms 100.100.86.99 (100.100.86.99) 14.297 ms
19 100.100.2.32 (100.100.2.32) 14.361 ms 100.100.88.227 (100.100.88.227) 14.704 ms 100.100.4.24 (100.100.4.24) 15.748 ms
20 99.83.113.93 (99.83.113.93) 15.617 ms 100.100.34.94 (100.100.34.94) 14.689 ms 99.82.181.25 (99.82.181.25) 14.959 ms
21 * 99.83.113.93 (99.83.113.93) 16.883 ms 108.170.246.33 (108.170.246.33) 16.294 ms
22 dns.google (8.8.8.8) 15.325 ms * *
```



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Traceroute from inside AWS (2)

EC2 INSTANCE IN MELBOURNE TRACEROUTE TO ELASTIC IP OF EC2 INSTANCE IN SYDNEY

```
% traceroute -n -q1 3.24.0.0
traceroute to 3.24.0.0 (3.24.0.0), 30 hops max, 60 byte packets
 1  244.5.0.1    1.647 ms
 2  240.1.72.6   0.185 ms
 3  240.1.192.13 10.690 ms
 4  15.230.210.36 13.923 ms
 5  15.230.210.45 19.009 ms
 6  15.230.210.96 10.888 ms
 7  15.230.211.4 11.825 ms
 8  240.1.184.15 11.448 ms
 9  240.1.184.30 11.345 ms
10  242.4.106.53 16.677 ms
11  3.24.0.0    11.696 ms
%
```



Disaggregated control plane

COMBINATION OF ON-DEVICE AND OFF-DEVICE

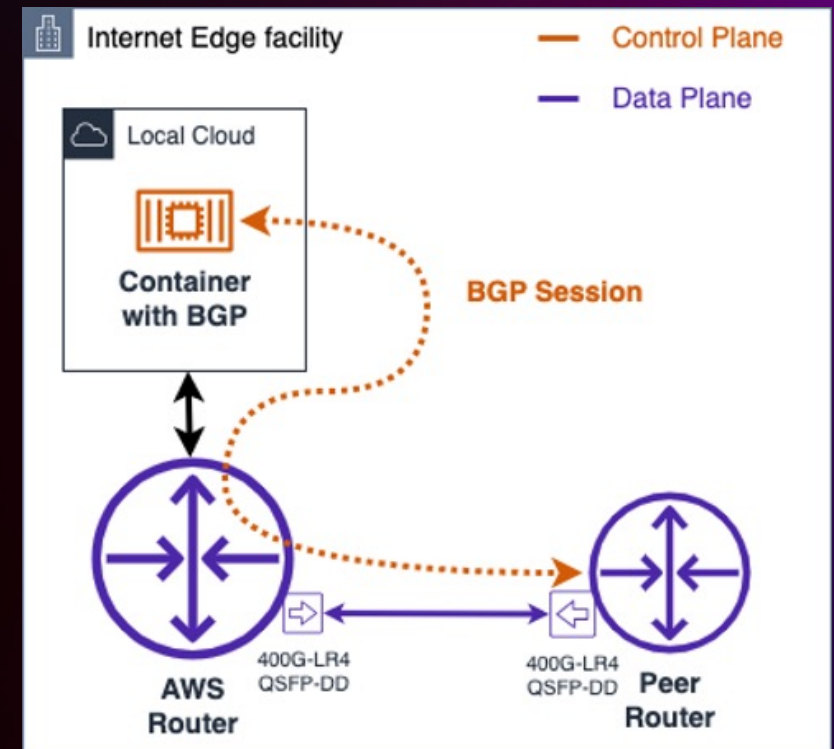
On-device handles local things like LACP, ARP/ND and all aspects of physical connectivity

BGP speaker runs elsewhere

Faster convergence and higher scale than would otherwise be possible

Enables us to iterate/evolve each part separately

Peer doesn't see anything different, TTL1 or TTL255 BGP still works the same way



How you see us / how we see you

COMBINATION OF ON-DEVICE AND OFF-DEVICE

```
re0> ping 189.233.232.1 source 189.233.232.0 size 1472 do-not-fragment
PING 189.233.232.1 (189.233.232.1) from 189.233.232.0 : 1472(1500) bytes of data.
1480 bytes from 189.233.232.1: icmp_seq=1 ttl=255 time=0.901 ms
1480 bytes from 189.233.232.1: icmp_seq=1 ttl=254 time=0.919 ms (DUP!)
1480 bytes from 189.233.232.1: icmp_seq=2 ttl=255 time=0.555 ms
1480 bytes from 189.233.232.1: icmp_seq=2 ttl=254 time=0.614 ms (DUP!)
```

```
% ip -d link show bond1
166: bond1: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 1500 qdisc noqueue master TMS-VRF
state UP mode DEFAULT group default qlen 1000
    link/ether a0:d0:dc:94:52:42 brd ff:ff:ff:ff:ff:ff promiscuity 0 minmtu 68 maxmtu 65535
    bond mode 802.3ad miimon 100 updelay 5000 downdelay 1100 peer_notify_delay 0 use_carrier 1
arp_interval 0 arp_validate none arp_all_targets any primary_reselect always fail_over_mac none
xmit_hash_policy layer2 resend_igmp 1 num_grat_arp 1 all_slaves_active 1 min_links 1
lp_interval 1 packets_per_slave 1 lacp_rate fast ad_select count ad_aggregator 1 ad_num_ports 1
ad_actor_key 12345 ad_partner_key 12345 ad_partner_mac 94:ae:f0:c8:38:dd tlb_dynamic_lb 1
    vrf_slave table 14 addrgenmode eui64 numtxqueues 16 numrxqueues 16 gso_max_size 65536
gso_max_segs 65535
    alias _EXTPEER-LAG_ AUSSIE_BROADBAND #1 AS4764 PS_ID:96445 BGP:bne50-br-fnc-f1-b1-bgp-r2-c1
```



The curious case of flaky IPv6 NS

LINUX MCAST_RESOLICIT (NON-DEFAULT) REQUIRED FOR TO TRIGGER SRC+DST LINK-LOCAL IPV6 NS

src FE80 dst 2620 fails, src FE80 dst FF80 WORKS TL;DR: Many people get FE80 ACLs wrong

```
% ip -ts monitor neigh dev bond1
[2023-01-13T02:58:15.544747] 2620:107:4008:xxx::2 dev bond1 lladdr d4:6a:35:35:4c:92 router PROBE
[2023-01-13T02:58:15.649269] fe80::d66a:35ff:fe25:4c92 dev bond1 lladdr d4:6a:35:35:4c:92 router PROBE
[2023-01-13T02:58:15.650764] fe80::d66a:35ff:fe25:4c92 dev bond1 lladdr d4:6a:35:35:4c:92 router REACHABLE
[2023-01-13T02:58:45.852977] 99.83.1xx.xx dev bond1 lladdr d4:6a:35:35:4c:92 PROBE
[2023-01-13T02:58:45.854469] 99.83.1xx.xx dev bond1 lladdr d4:6a:35:35:4c:92 REACHABLE
[2023-01-13T02:58:46.112645] fe80::d66a:35ff:fe25:4c92 dev bond1 lladdr d4:6a:35:35:4c:92 router PROBE
[2023-01-13T02:58:46.114825] fe80::d66a:35ff:fe25:4c92 dev bond1 lladdr d4:6a:35:35:4c:92 router REACHABLE
[2023-01-13T02:58:52.413809] 2620:107:4008:xxx::2 dev bond1 router FAILED
[2023-01-13T02:59:07.779235] 2620:107:4008:xxx::2 dev bond1 lladdr d4:6a:35:35:4c:92 router REACHABLE
[2023-01-13T02:59:16.305279] 99.83.1xx.xx dev bond1 lladdr d4:6a:35:35:4c:92 PROBE
[2023-01-13T02:59:16.306371] 99.83.1xx.xx dev bond1 lladdr d4:6a:35:35:4c:92 REACHABLE
[2023-01-13T02:59:16.473164] 2620:107:4008:xxx::2 dev bond1 lladdr d4:6a:35:35:4c:92 router PROBE
[2023-01-13T02:59:16.570665] fe80::d66a:35ff:fe25:4c92 dev bond1 lladdr d4:6a:35:35:4c:92 router PROBE
[2023-01-13T02:59:16.574393] fe80::d66a:35ff:fe25:4c92 dev bond1 lladdr d4:6a:35:35:4c:92 router REACHABLE
[2023-01-13T02:59:46.767019] 99.83.1xx.xx dev bond1 lladdr d4:6a:35:35:4c:92 PROBE
[2023-01-13T02:59:46.770263] 99.83.1xx.xx dev bond1 lladdr d4:6a:35:35:4c:92 REACHABLE
[2023-01-13T02:59:47.025611] fe80::d66a:35ff:fe25:4c92 dev bond1 lladdr d4:6a:35:35:4c:92 router PROBE
[2023-01-13T02:59:47.026513] fe80::d66a:35ff:fe25:4c92 dev bond1 lladdr d4:6a:35:35:4c:92 router REACHABLE
[2023-01-13T02:59:53.341824] 2620:107:4008:xxx::2 dev bond1 router FAILED
[2023-01-13T03:00:07.779211] 2620:107:4008:xxx::2 dev bond1 lladdr d4:6a:35:35:4c:92 router REACHABLE
```

36 IPv6 NS sent,
none answered

As soon as we age
out the entry..
..it then answers

..rinse/repeat..

```
% tcpdump -i bond1 -n -p --direction=out 'icmp6'
04:15:57.597793 IP6 fe80::a2d0:dcff:fefc:8ed6 > 2620:107:4008:xxx::2: ICMP6, neigh solicitation, who has 2620:107:4008:xxx::2
04:16:02.717802 IP6 fe80::a2d0:dcff:fefc:8ed6 > 2620:107:4008:xxx::2: ICMP6, neigh solicitation, who has 2620:107:4008:xxx::2
04:16:07.837808 IP6 fe80::a2d0:dcff:fefc:8ed6 > 2620:107:4008:xxx::2: ICMP6, neigh solicitation, who has 2620:107:4008:xxx::2
04:16:10.407026 IP6 fe80::a2d0:dcff:fefc:8ed6 > fe80::d66a:35ff:fe35:4c92: ICMP6, neighbor advertisement, tgt is fe80::a2d0:d
04:16:12.957792 IP6 fe80::a2d0:dcff:fefc:8ed6 > ff02::1:ff00:2: ICMP6, neighbor solicitation, who has 2620:107:4008:xxx::2
```



Cisco handling of link-local IPv6 nexthops

NEIGH SOLICITATION ON UNRESOLVED V6 NEXTHOP (VERSUS NOT USING THE ONE THAT DID RESOLVE!)

```
# tcpdump on container terminating BGP, we see route announced to peer
...
21:20:10.427367 IP6 (flowlabel 0x7236d, hlim 1, next-header TCP (6) payload length: 286)
2620:107:XXXX:YYYY::1.41991 > 2620:107:XXXX:YYYY::2.179: Flags [P.], seq 1660:1926, ack 38, win 15745, length 266: BGP
Update Message (2), length: 117
  Origin (1), length: 1, Flags [T]: IGP
    0x0000: 00
  AS Path (2), length: 6, Flags [T]: 16509
    0x0000: 0201 0000 407d
  Multi Exit Discriminator (4), length: 4, Flags [O]: 1000
    0x0000: 0000 03e8
  Multi-Protocol Reach NLRI (14), length: 44, Flags [OE]:
    AFI: IPv6 (2), SAFI: Unicast (1)
    nexthop: 2620:107:XXXX:YYYY::1, fe80::f040:4861, nh-length: 32, no SNPA
    2605:a7c0:12a::/48
    2605:a7c0:10a::/48
...
```

... because we provided a link-local nexthop
... alongside a unicast nexthop

```
# tcpdump on physical device terminating peering session
% sudo tcpdump -i bond13 -n -vve icmp6
tcpdump: listening on bond13, link-type EN10MB (Ethernet), capture size 262144 bytes
21:52:17.498480 94:ae:f0:c3:a0:d9 > 33:33:ff:40:48:61, ethertype IPv6 (0x86dd), length 86: (class 0xe0, hlim 255, next-header ICMPv6 (58) payload length: 32) fe80::96ae:f0ff:fec3:a0d9 > ff02::1:ff40:4861: [icmp6 sum ok] ICMP6, neighbor solicitation, length 32, who has fe80::f040:4861
  source link-address option (1), length 8 (1): 94:ae:f0:c3:a0:d9
...
```

Why is the peer sending us a NS for an address we don't have?



AWS Global Infrastructure

AWS REGIONS, EDGE LOCATIONS, AND THE GLOBAL BACKBONE

AWS NETWORK BACKBONE

- Redundant 400 Gbps links
- 245+ Countries & Territories
- Between all Regions, Local Zones, and Edge Locations



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

<http://ec2-reachability.amazonaws.com/>

- Sydney ping/traceroute 3.24.0.0
- Melbourne ping/traceroute 16.50.164.255
- Perth ping/traceroute 96.0.1.218
- (see <http://ec2-reachability.amazonaws.com/> for other IPv4/v6 locations)
- `curl https://blue.diagnostics.globalaccelerator.aws/api/stack
{"ip":"13.248.102.33","stack":"MEL50-1"}`
- Under 2msec (ideally <1msec) RTT wired ethernet CPE on NBN FTTP -> you -> us



https://interconnect.amazon/

SELF-SERVICE IX PEERING VIA PEERINGDB CREDENTIALS (FOR ALL ELSE PEERING-APAC@AMAZON.COM)

Peering

Sessions

Endpoints

Internet services

Amazon Peering

Fast and secure interconnect for new and existing customers

Amazon Peering offers a secure, high-speed, and reliable way to connect your network to Amazon's global network.

Benefits

- Peering made easy
- Create peering sessions in minutes
- Internet Exchange status in a few clicks

Peering

Sessions

Endpoints

Peering > Sessions > Create session

Step 1
Specify peering endpoints

Step 2
Configure the sessions to create

Step 3
Select contact details

Step 4
Review and create

Specify peering endpoints

Peering sessions are established between Amazon endpoints and your network ports on a given facility.

► Steps to create a new peering session

Peering endpoints (2/440)

Select the Amazon peering endpoints you want to create a peering session with.

Search

4 matches < 1 > ⚙

Location = NL, Amsterdam, AMS-IX, Europe

Clear filters

	Location	Peering type	Amazon IP address
<input checked="" type="checkbox"/>	NL, Amsterdam, AMS-IX, Europe	PUBLIC	80.249.210.100
<input checked="" type="checkbox"/>	NL, Amsterdam, AMS-IX, Europe	PUBLIC	2001:7f8:1::a501:6509:2
<input type="checkbox"/>	NL, Amsterdam, AMS-IX, Europe	PUBLIC	80.249.210.217
<input type="checkbox"/>	NL, Amsterdam, AMS-IX, Europe	PUBLIC	2001:7f8:1::a501:6509:1

Destination network

Select what networks you want Amazon to create a peering relationship with. These networks are identified by the autonomous system number (ASN) and IP address of the router present in the exchange.

ASN


Select AS numbers from those in the list owned by your organization.

Choose options

46489
Amazon IVS / Twitch

Cancel

Next

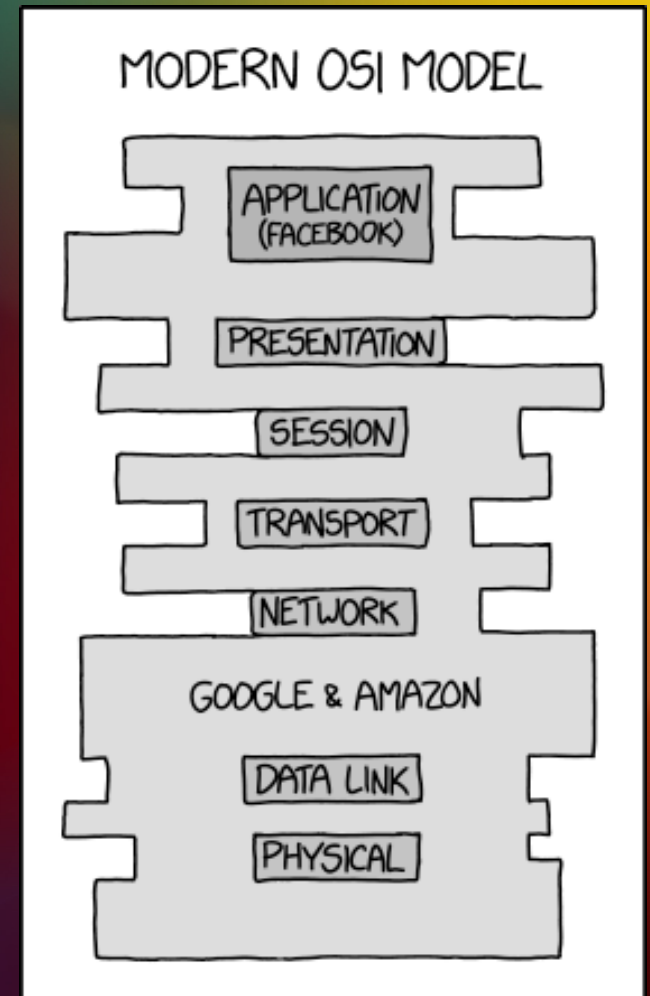


© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank You

Lincoln Dale

Senior Principal Engineer
AWS – AS16509



Source: <https://xkcd.com/2105>, Randall Munroe



© 2022, Amazon Web Services, Inc. or its affiliates. All rights reserved.