



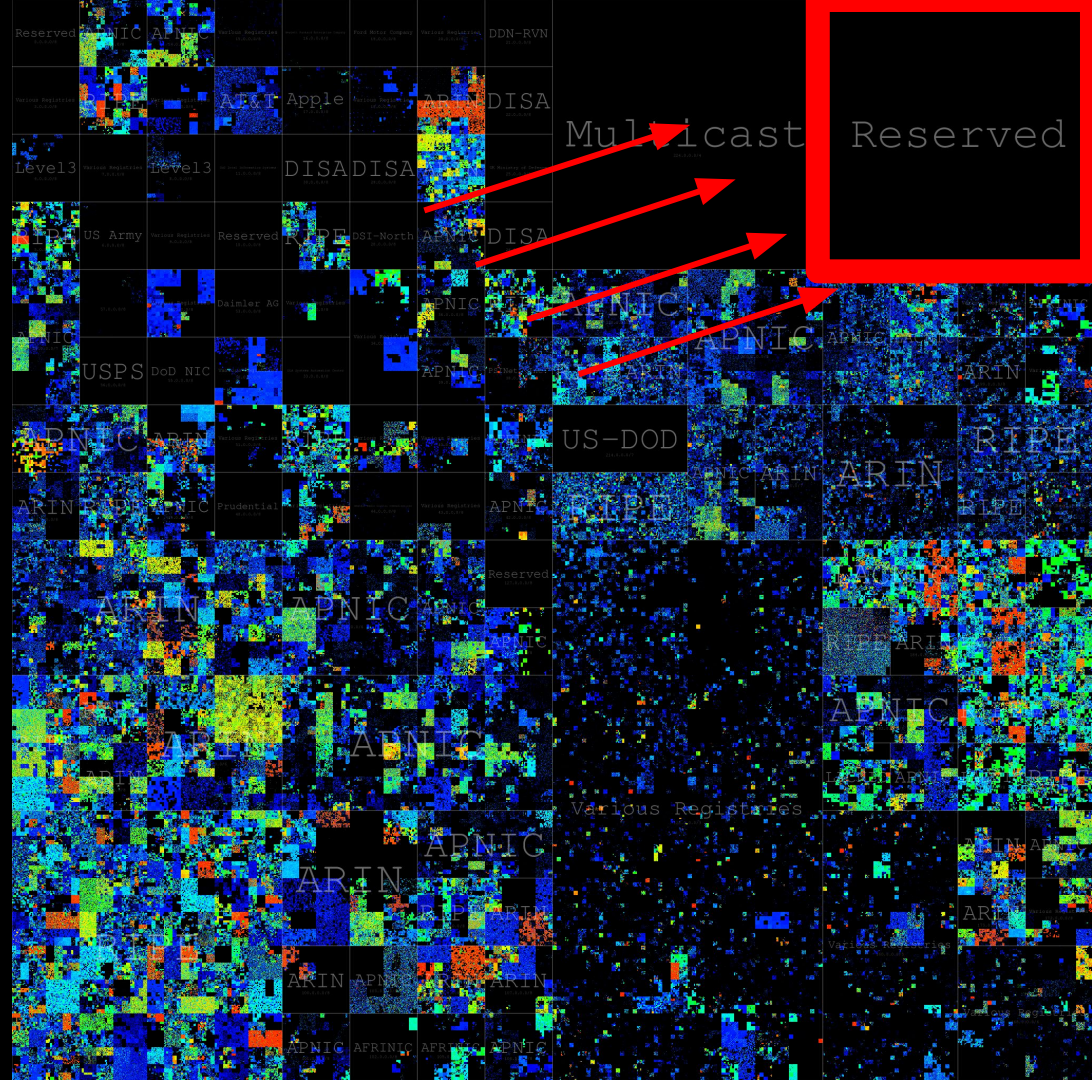
# "Reclaiming" 240.0.0.0/4

Ben Cartwright-Cox

Zbyněk Pospíchal

# Class E Space

- Lives between 240.0.0.0 -> 255.255.255.254
- Right after Multicast space
- Currently sits in "Reserved" State
- Has existed this way for a long time, but no one really cared until...



# IPv4 Exhaustion

- The IPv4 machine stopped providing IPv4 addresses at scale
- At what point did we "run out" of IPv4 addresses is a debate
  - A pretty boring debate as it kind of does not matter to the business case in question
- People go looking for "other" addresses (As there will always be some people who think that IPv6 is not ever going to reach universal adoption)
  
- Class E is not alone in being eyed up for reclamation, 0.0.0.0/8 could *technically* be reclaimed (other than 0.0.0.0), and 127.0.0.0/8 could *technically* be reclaimed (with some obvious restrictions)

# Some hearsay about how we got here in the first place

- It was not obvious that the internet was going to be actually used
- Nor was it obvious (at the start) that we would run out of 32 bit address space
- Several blocks were left unusable for unicast
  - 0.0.0.0/8
    - Could have easily been changed to 0.0.0.0/24 or even 0.0.0.0/32, when classful routing got going
  - 127.0.0.0/8
    - Similar story to 0.0.0.0/8
  - 240.0.0.0/4
    - Was left around incase of a mythical "3rd type of routing" was discovered (that wasn't unicast or multicast)
  - 224.0.0.0/4 (yes multicast)
    - There is draft-gilmore-taht-v4uniext-01 that suggests 225/8 through 231/8 to become unicast

# Let me make my views known (I'm not crazy I promise)

- You should be deploying IPv6
- Class E (and others) will likely never get into the global routing table
  - Changing carrier policy is hard
  - Changing end user devices is hard
  - Who wants addresses that might not work for some users???
    - You could joke that already exists, *it's called IPv6*
- However Class E is a interesting idea for local addressing

# Class E as a local unicast space

- It happens that 10.0.0.0/8 (and other RFC1918 space) is not actually as big as some people hoped it would be
- 240.0.0.0/4 is however, very big
- There is already some precedent in using 240.0.0.0/4 space for this:
  - AWS uses 240.0.0.0/4 space internally for some network devices [1]
  - They also exist in some other networks too, seen in traceroutes [2]
  - Also some Home/SMB networks are using 240/4 space (somehow?) [3]
  - Fan container networking by Canonical uses it [4]

[1] <https://www.youtube.com/watch?v=0tcR-iQce7s&t=1709s>

[2] <https://labs.ripe.net/author/qasim-lone/2404-as-seen-by-ripe-atlas/>

[3] <https://blog.benjojo.co.uk/post/ip-address-squatting>

[4] <https://canonical.com/blog/introducing-the-fan-simpler-container-networking>

# Class E as a weird bodge

- Cloudflare has a weird option to hash IPv6 addresses into Class E IPv4 addresses
  - My own experience is that nearly no one uses this, because it's easy for a client to abuse
  - <https://developers.cloudflare.com/network/pseudo-ipv4/>
  - TL;dr MD5 the IPv6 address and mask that into a Class E space address



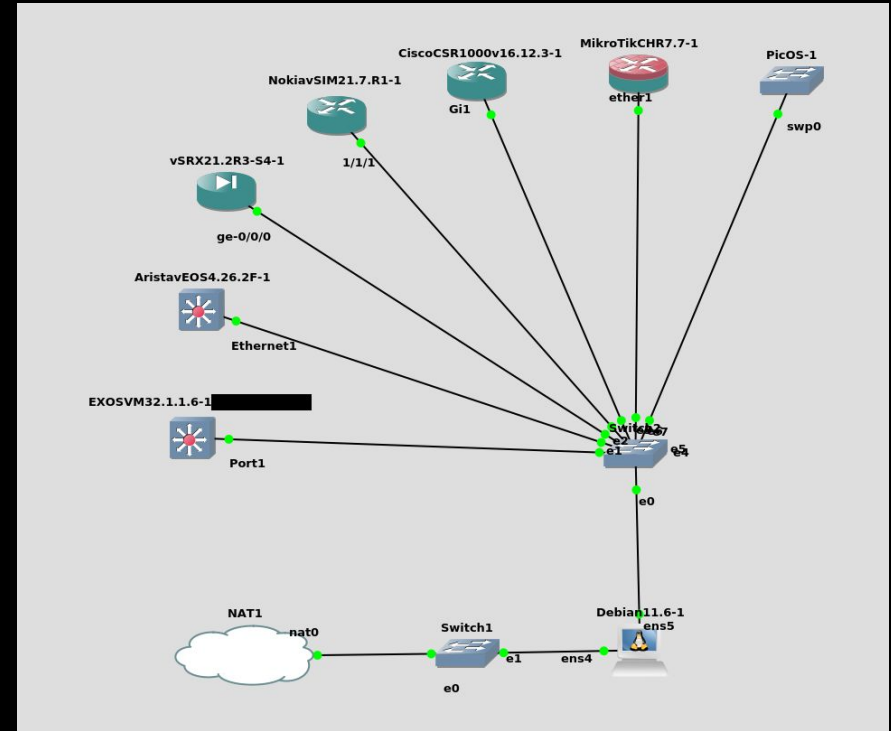
# Unfortunately, *Vendors!*

- **Stuff that works**
  - Almost all Linux distros post 2008
  - Android post 2009
  - MacOS / OSX post 2008/9 (iOS implicated in support as well)
  - OpenBSD post October 2022
  - Arista EOS Post 2021 (use `ipv4 routable 240.0.0.0/4`)
  - JunOS post 2009! (use `edit routing-options martians 240/4 or longer allow`)
- **Stuff that does not work**
  - Windows
  - NetBSD / FreeBSD



# Testing dynamic routing on vendors

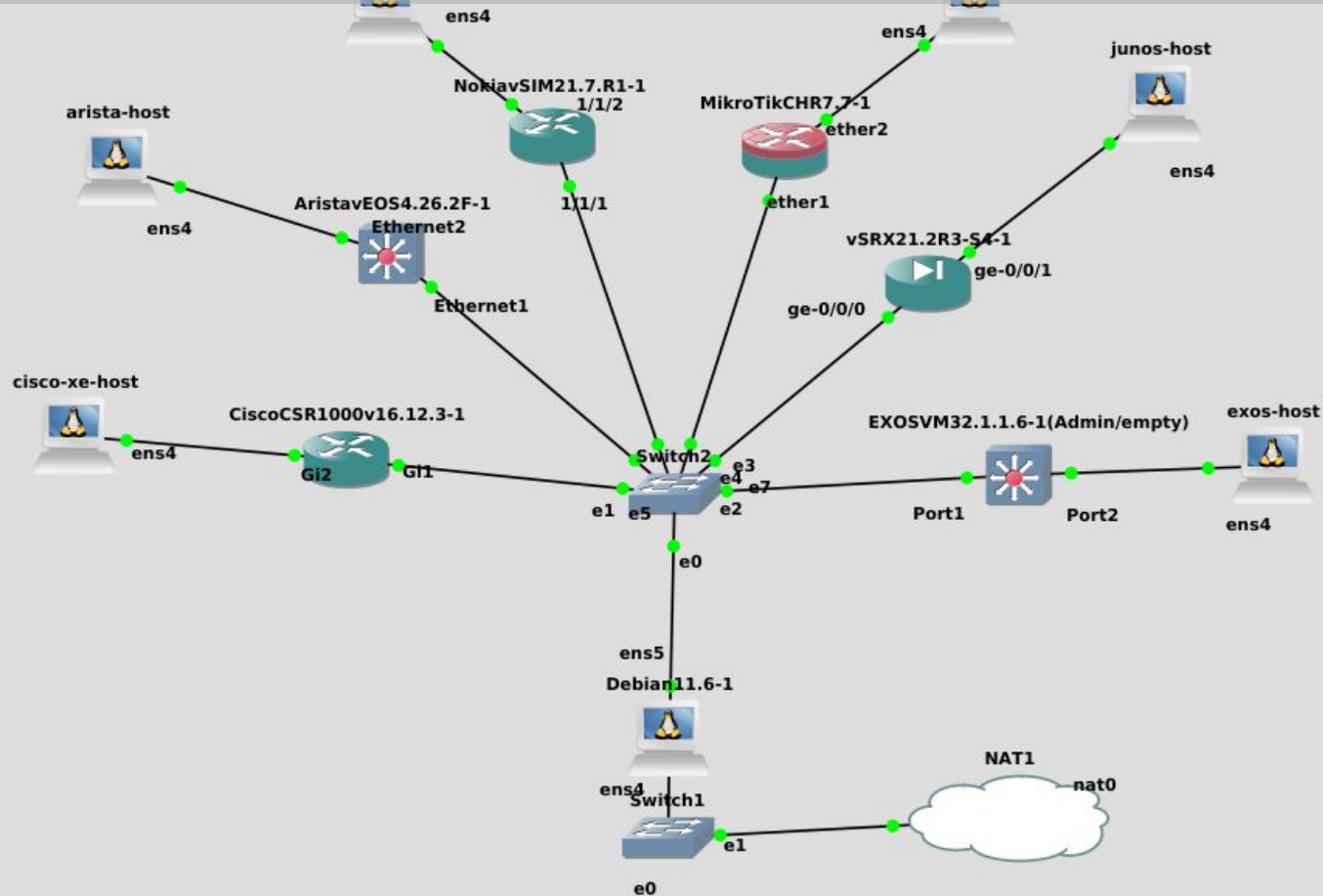
- Build a router ~~petting~~ testing zoo
- Test if they can all peer with each other on OSPF/BGP using class E link locals, send, and install Class E routes



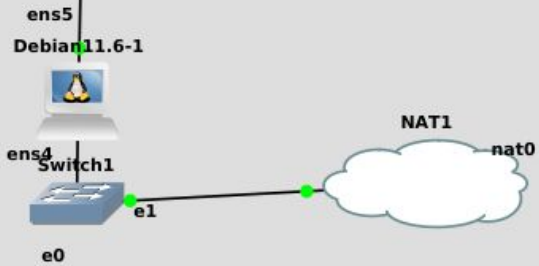
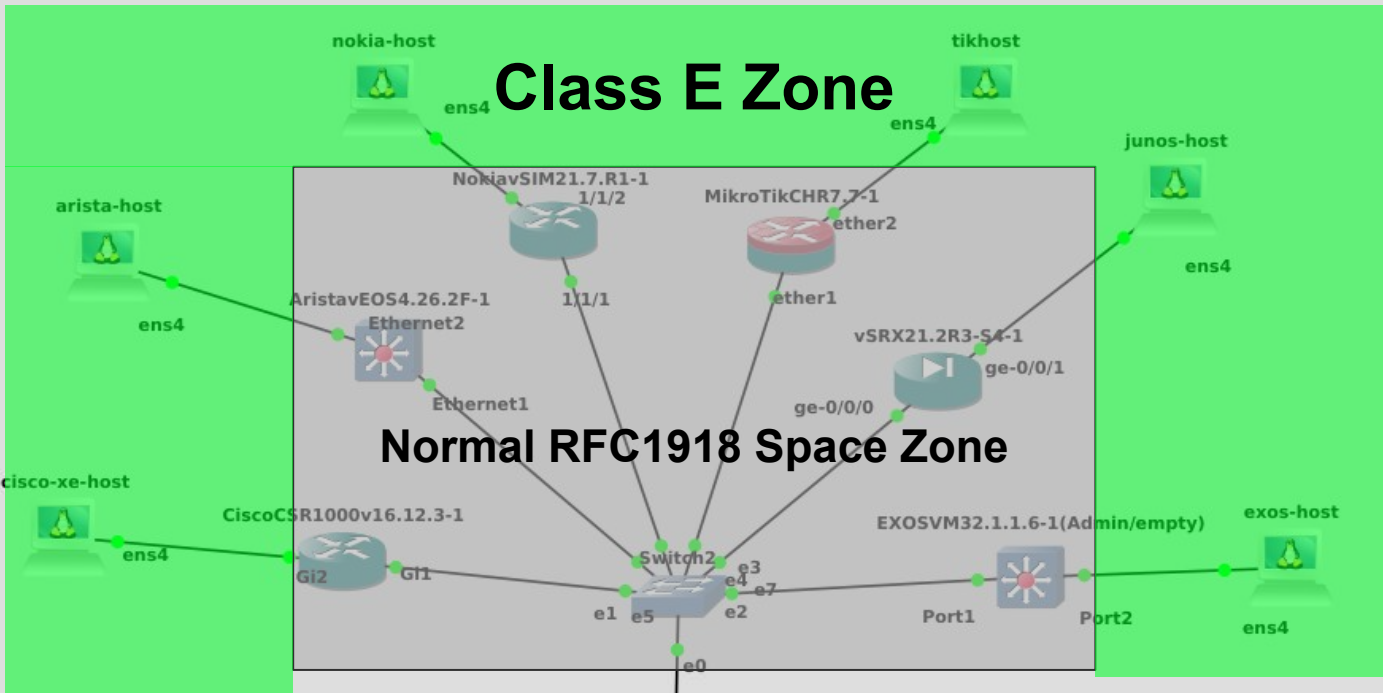
# Vendor results (BGP peering on Class E space)

- EXOS 32.1.1.6-1
  - Won't set Class E Link locals
- Arista (v)EOS 4.29.0.2F
  - Works fine with adjustments
- JunOS 22.X
  - Works fine with adjustments
- RouterOS 7.7
  - Works fine
- IOS XR
  - Works fine
- IOS XE
  - Won't set Class E Link locals
- Nokia SR-OS
  - Won't set Class E Link locals
- Huawei VRP
  - Won't set Class E Link locals

# A different experiment



# Class E Zone



## JunOS quirks

```
root@vsrx# commit
```

```
[edit system services dhcp pool]
```

```
'240.0.2.0/24'
```

```
Subnet specified cannot be used in a pool
```

```
error: configuration check-out failed
```

Everything works, except the DHCP server

# Arista (vEOS) Quirks

```
localhost#configure
```

```
localhost(config)#router general
```

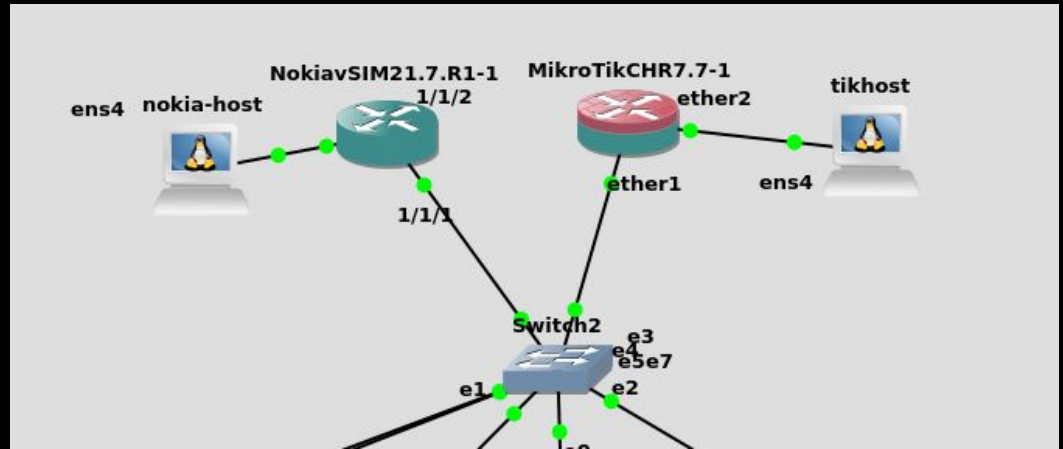
```
localhost(config-router-general)#ipv4 routable 240.0.0.0/4
```

```
% Unavailable command (not supported on this hardware platform)
```

"I will just have to assume this works"

# OSPF Quirks

- Class E Prefixes from other neighbors will be carried over OSPF, to other members.
- However SROS/IOS-XE will not install them into the RIB or FIB
- This is lethal since it means that the router will claim it can route these packets, but will drop them instead.



```
nokia-host # birdc s ro | grep 240
192.0.3.0/24      unicast [direct1 15:43:15.222] *
240.0.1.0/24     unicast [ospf1 15:42:58.257] * I
240.0.2.0/24     unicast [ospf1 15:42:58.257] * I
240.255.1.1/32   unicast [direct1 15:43:32.751] *
6.6.6.6/32       unicast [direct1 15:43:32.751] *
192.168.122.0/24 unicast [ospf1 15:42:58.257] * I
```

```
[admin@MikroTik] > tool/traceroute 240.255.1.1
Columns: LOSS, SENT, LAST
#  LOSS  SENT  LAST
1  100%   146  timeout
2  100%   146  timeout
3  100%   146  timeout
4  100%   146  timeout
5  100%   146  timeout
```

```
[admin@MikroTik] > tool/traceroute 6.6.6.6
Columns: ADDRESS, LOSS, SENT, LAST, AVG, BEST, WOR
#  ADDRESS  LOSS  SENT  LAST  AVG  BEST  WOR
1  192.0.2.6  0%    5    1.7ms  1.7  1.6  1.9
2  6.6.6.6   0%    5    1.9ms  2    1.5  2.5
```



# Vendor results (routing 'E' prefixes via OSPF/BGP)

- Arista (v)EOS 4.29.0.2F
  - I can't test routing, but OSPF etc works
- JunOS 22.X
  - Works! (With previous tweaks)
- RouterOS 7.7
  - Works!
- IOS XR
  - Works!
- IOS XE
  - 🚩 Will relay class E OSPF routes, won't install into FIB 🚩
- Nokia SR-OS
  - 🚩 Will relay class E OSPF routes, won't install into FIB 🚩
- Huawei VRP
  - 🚩 Will relay class E OSPF routes, won't install into FIB 🚩

And then, out of nowhere


# Surprise test from AS8747 (Quantcom, a.s.)




Fri, Jan 19, 2024

class E routing experiment - Mozilla Thunderbird

File Edit View Go Message Tools Help

Get Messages Write Tag

From Zbyněk Pospíchal <zbynek.pospichal@quantcom.cz> 

To tech@lists.nix.cz <tech@lists.nix.cz> , six-providers@six.sk <six-providers@six.sk> , vixoper@vix.at 

Subject **class E routing experiment**

11:34

Dear peering partners,

We started to announce a prefix 242.242.0.0/16 from our experimental ASN 8747. 242.242.1.1 should be pingable if such a route passes your filters and router setup.

Current test results are that IOS XR supports class E routes with no issues by default. With JunOS, using a command "set routing-options martians" etc. is necessary. Your feedback regarding other platforms is appreciated.

--









































S pozdravem/Best Regards,

Zbyněk Pospíchal  
Quantcom, a.s.

Done































# RIPE Atlas findings

50% of probes inside their own network can reach it!

Settings & Status	Latest Results	Map	Tracemon	IPMap	Downloads				
Probe	ASN (IPv4)	ASN (IPv6)		Time (UTC)	RTT	Hops	Success		
13200	29208		 	2024-01-19 18:11	0.718		8	✗	
19753	29208		 	2024-01-19 18:11	0.479		7	✗	
20515	29208		 	2024-01-19 18:11	✗		6	✗	
22072	29208		 	2024-01-19 18:11	✗		6	✗	
30955	29208		 	2024-01-19 18:11	1.345		7	✓	
60830	29208		 	2024-01-19 18:11	1.837		8	✓	
62603	29208		 	2024-01-19 18:11	✗		6	✗	
1003739	29208	29208	 	2024-01-19 18:11	8.857		9	✓	
1004141	29208		 	2024-01-19 18:11	1.805		6	✓	
1004199	29208		 	2024-01-19 18:11	4.832		6	✓	

## RIPE Atlas findings (2)

Again 50% of probes from their downstreams can get to it!

Probe	ASN (IPv4)	ASN (IPv6)		Time (UTC)	RTT	Hops	Success		
7211	25248	25248	 	2024-01-19 18:19	0.602		7		
13490	48926	48926	 	2024-01-19 18:19	2.082		9		
33287	205535		 	2024-01-19 18:19	9.854		4		
53850	203488	203488	 	2024-01-19 18:19	3.315		8		
1005950	48926	48926	 	2024-01-19 18:19	1.805		7		
1006785	203488	203488	 	2024-01-19 18:19	0.494		7		

## "Who wants space that does not work?"

- Pre (total) IPv4 depletion one of my old employers got a /12
- This /12 was immediately announced, and it became clear upon testing that not the whole internet could reach it!
- Some people have/had bogon filters, and since this new /12 **was** unallocated! So it was filtered!
- This was a extreme pain in the ass to fix, We moved some customers to the /12 and worked with everyone to fix the issue
- It took ~10 weeks, but nearly every offender had fixed the problem at that point

Now imagine that but with IP space that needs config tweaks in most of your network



# Pros and Cons of reclaiming 240/4

## Pros

- IPv4 Link locals or internal prefixes can now be moved away from RFC1918 space
- If easy, could provide a temporary (**LOL**) fix to your internal IP space problems

## Cons

- Network vendor support is poor
- Endpoint support is poor (Windows)
- "Ownership" of 240.0.0.0/4 is up for debate
- **A lot of the work to make 240/4 work could be spent on IPv6 instead**
- Other people's weird uses of 240/4 may collide with you. Making addresses in 240/4 murky

# In summary (In Ben's eyes)

- Getting 240/4
  - For local unicast networks?
    - Not too insane, we already see networks doing this and some vendors supporting it for those networks, Clearly it works for some
  - For inter-network unicast?
    - Insane.
- Getting 0.0.0.0/8 / 127.0.0.0/8
  - For local unicast?
    - Insane. Massive amounts of work for no real gain
  - For inter-network unicast?
    - Insane.

# Practice > Theory

Zbyněk Pospíchal to show what Quantcom  
saw!

## Scanning /0 from Class E space

- 3.6Bil (ish) ICMP ping sent by Zbyněk
- 184,496 reponses (0.005%), 380,286,307 would indicate fully reachable
- Networks that peer with Quantcom, since RS would not accept this
- Mostly thanks to IOS XR ability to forward 240/4 with no additional config

# Large scale acceptance

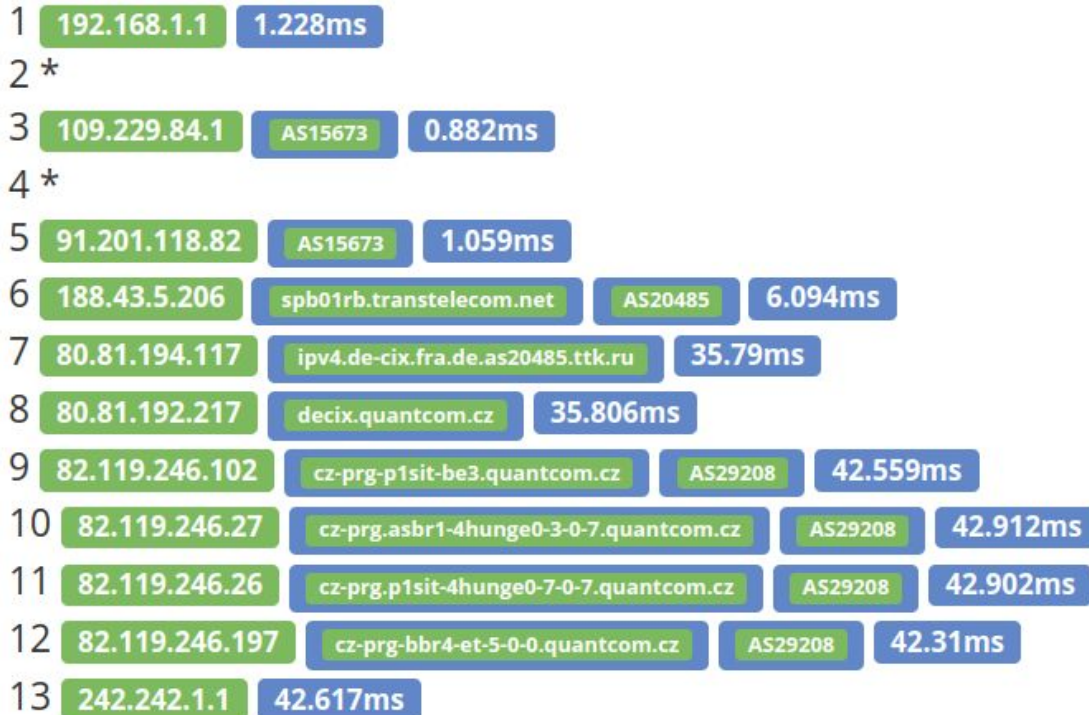
Interesting networks that accept include:

- Akamai (Prolexic)
- Quad9
- AT&T EMEA / AS2686
- TTK / JSC TransTeleKom / AS20485
- Orange Romania / AS9050
- Cloudflare's Prague PoP / AS13335
- several networks in Vietnam, UAE etc.

## Latest Traceroute Result for Measurement #70179055 ×

2024-04-22 10:05 UTC

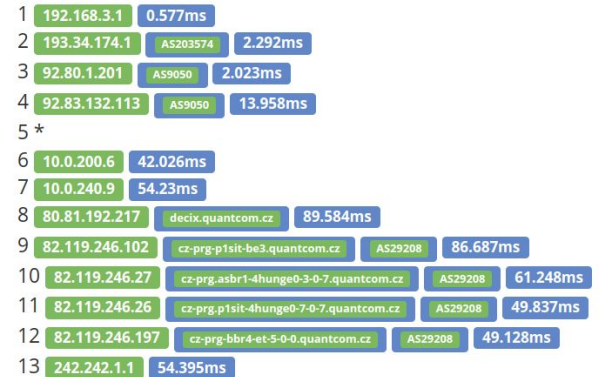
Traceroute to 242.242.1.1 (242.242.1.1), 48 byte packets



## Latest Traceroute Result for Measurement #70179345 ×

2024-04-22 10:17 UTC

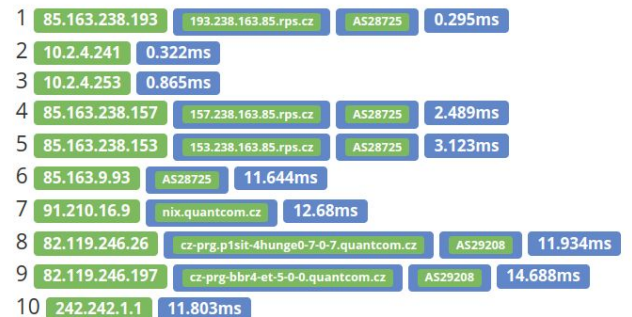
Traceroute to 242.242.1.1 (242.242.1.1), 48 byte packets

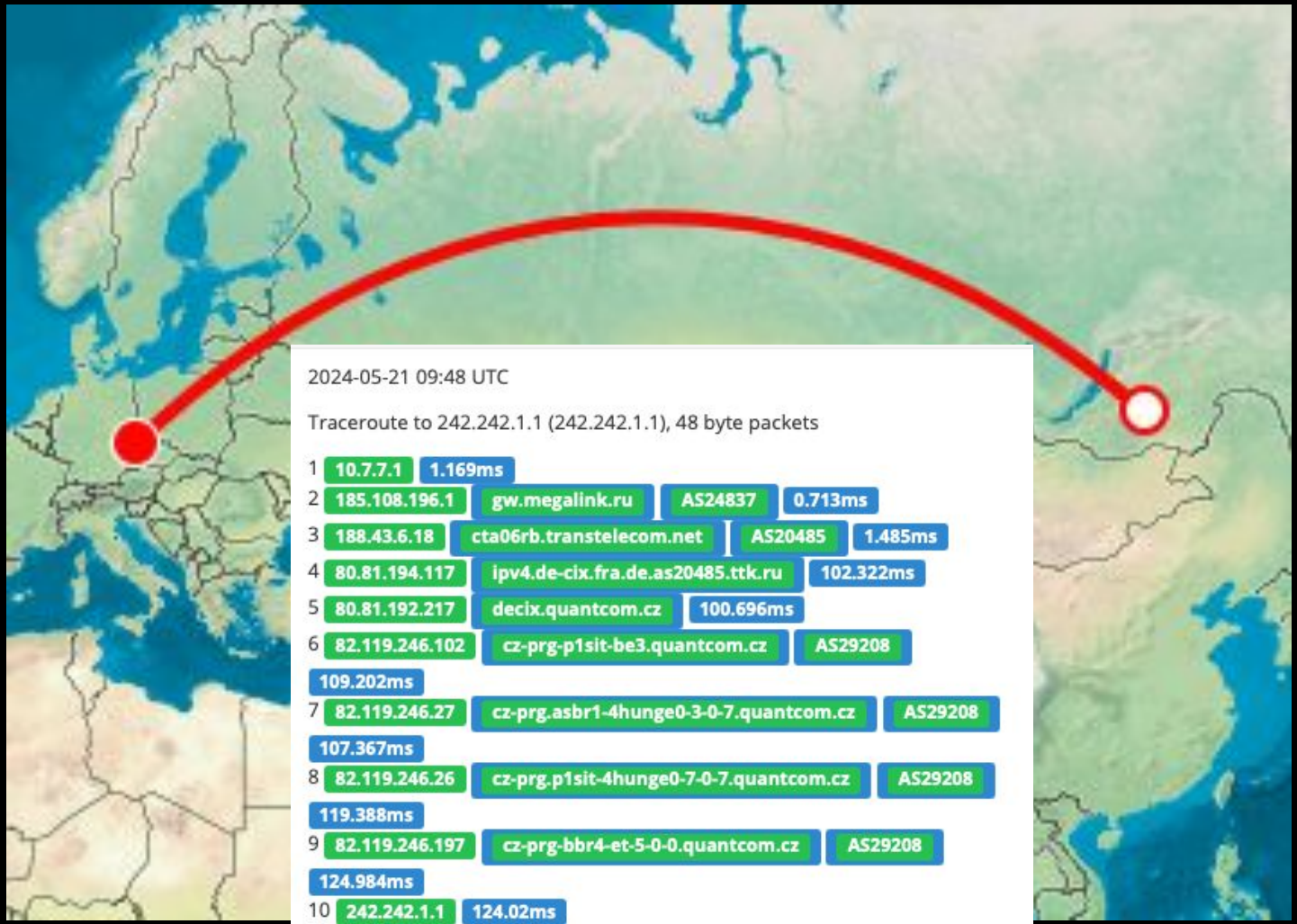


## Latest Traceroute Result for Measurement #70178933 ×

2024-04-22 09:56 UTC

Traceroute to 242.242.1.1 (242.242.1.1), 48 byte packets





2024-05-21 09:48 UTC

Traceroute to 242.242.1.1 (242.242.1.1), 48 byte packets

- 1 10.7.7.1 1.169ms
- 2 185.108.196.1 gw.megalink.ru AS24837 0.713ms
- 3 188.43.6.18 cta06rb.transtelecom.net AS20485 1.485ms
- 4 80.81.194.117 ipv4.de-cix.fra.de.as20485.ttk.ru 102.322ms
- 5 80.81.192.217 decix.quantcom.cz 100.696ms
- 6 82.119.246.102 cz-prg-p1sit-be3.quantcom.cz AS29208  
109.202ms
- 7 82.119.246.27 cz-prg.asbr1-4hunge0-3-0-7.quantcom.cz AS29208  
107.367ms
- 8 82.119.246.26 cz-prg.p1sit-4hunge0-7-0-7.quantcom.cz AS29208  
119.388ms
- 9 82.119.246.197 cz-prg-bbr4-et-5-0-0.quantcom.cz AS29208  
124.984ms
- 10 242.242.1.1 124.02ms



# Quad9 DNS responses

```
$ dig -b 242.242.1.1 @9.9.9.9 isc.org AAAA
; <<>> DiG 9.16.27-Debian <<>> -b 242.242.1.1 @9.9.9.9 isc.org AAAA
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 22090
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;isc.org.          IN      AAAA
;; ANSWER SECTION:
isc.org.          300    IN      AAAA      2001:500:6b:2::28
;; Query time: 172 msec
;; SERVER: 9.9.9.9#53(9.9.9.9)
;; WHEN: Wed Apr 03 15:00:21 CEST 2024
;; MSG SIZE rcvd: 64
```

Conclusion

# So should you use 240/4?

No, unless you are totally out of RFC1918 and have no other options

Shy? Email [ripe88@benjojo.co.uk](mailto:ripe88@benjojo.co.uk)

# Questions?



<- Raw Data