Advanced Computer Networking (ACN)

IN2097 – WiSe 2025–2026

Prof. Dr.-Ing. Georg Carle, Sebastian Gallenmüller

Christian Dietze, Marcel Kempf, Lorenz Lehle

Chair of Network Architectures and Services
School of Computation, Information and Technology
Technical University of Munich

Why do we need Internet measurements?

Do we really have to?

- The network is well engineered
- Well documented protocols, mechanisms, ...
- Everything built by humans
  - → No unknowns (compare this to physics)
- In theory, we can know everything that is going on

→ No need for measurements?!

Why do we need Internet measurements?

Do we really have to?

- The network is well engineered
- Well documented protocols, mechanisms, ...
- Everything built by humans
  - → No unknowns (compare this to physics)
- In theory, we can know everything that is going on

→ No need for measurements?!

But:

- Distributed multi-domain network
  - → Information only partially available
- Moving target
  - Requirements change
  - Growth, usage, structure changes
- Highly interactive system
- Heterogeneity in all directions
- The total is more than the sum of its pieces
- Built, driven, and used by humans
  - → Errors, misconfigurations, flaws, failures, misuse, ...

Why do we need Internet measurements?

Do we really have to?

- The network is well engineered
- Well documented protocols, mechanisms, ...
- Everything built by humans
  - → No unknowns (compare this to physics)
- In theory, we can know everything that is going on

→ No need for measurements?!

But:

- Distributed multi-domain network
  - → Information only partially available
- Moving target
  - Requirements change
  - Growth, usage, structure changes
- Highly interactive system
- Heterogeneity in all directions
- The total is more than the sum of its pieces
- Built, driven, and used by humans
  - → Errors, misconfigurations, flaws, failures, misuse, ...

Active network measurements are an important research area to understand the Internet and interactions between all its components.

Why do we measure the network?

Network provider view

- Manage traffic
  - Model reality
  - Predict future
  - Plan network
  - Avoid bottlenecks in advance
- Reduce cost
- Accounting

Why do we measure the network?

Network provider view

- Manage traffic
  - Model reality
  - Predict future
  - Plan network
  - Avoid bottlenecks in advance
- Reduce cost
- Accounting

Service provider view

- Get information about clients
- Adjust service to demands
- Reduce load on servers
- Accounting

Why do we measure the network?

Network provider view

- Manage traffic
  - Model reality
  - Predict future
  - Plan network
  - Avoid bottlenecks in advance
- Reduce cost
- Accounting

Service provider view

- Get information about clients
- Adjust service to demands
- Reduce load on servers
- Accounting

Client view

- Get the best possible service
- *Do I get what I paid for?*

Why do we measure the network?

Network provider view

- Manage traffic
  - Model reality
  - Predict future
  - Plan network
  - Avoid bottlenecks in advance
- Reduce cost
- Accounting

Service provider view

- Get information about clients
- Adjust service to demands
- Reduce load on servers
- Accounting

Client view

- Get the best possible service
- *Do I get what I paid for?*

Security view

- Detect malicious traffic
- Detect malicious hosts
- Detect malicious networks

Why do we measure the network?

Network provider view

- Manage traffic
  - Model reality
  - Predict future
  - Plan network
  - Avoid bottlenecks in advance
- Reduce cost
- Accounting

Service provider view

- Get information about clients
- Adjust service to demands
- Reduce load on servers
- Accounting

Client view

- Get the best possible service
- *Do I get what I paid for?*

Security view

- Detect malicious traffic
- Detect malicious hosts
- Detect malicious networks

Researcher view

- Understand the Internet better
- *Could our new routing algorithm handle all this real-world traffic?*
- . . .

Ping

- Checks if host is reachable, alive
- Uses ICMP echo request/reply
- Copy packet data request reply

```
PING net.in.tum.de (131.159.15.24): 56 data bytes
64 bytes from 131.159.15.24: icmp_seq=0 ttl=63 time=4.033 ms
64 bytes from 131.159.15.24: icmp_seq=1 ttl=63 time=13.310 ms
64 bytes from 131.159.15.24: icmp_seq=2 ttl=63 time=58.955 ms
64 bytes from 131.159.15.24: icmp_seq=3 ttl=63 time=7.143 ms
^C
--- net.in.tum.de ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 4.033/20.860/58.955/22.246 ms
```

Listing 1: Sample output of ping

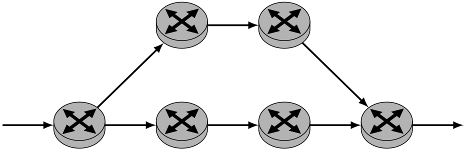
Traceroute

- Allows to follow path taken by packet
- Send UDP/TCP/... packets with increasing TTL to (unlikely) port
- ICMP replies: ‘time exceeded’; last ICMP message: ‘port unreachable’

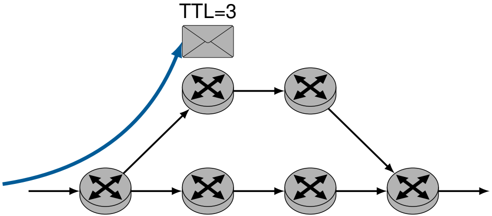
```
$ traceroute gaia.cs.umass.edu
1  scylla (131.159.20.11)  4.263 ms  2.531 ms  2.162 ms
2  nz-bb-net.informatik.tu-muenchen.de (131.159.252.149)  6.124 ms  15.174 ms  3.546 ms
3  nz-csr1-kw5-bb1.informatik.tu-muenchen.de (131.159.252.2)  2.925 ms  4.234 ms  3.033 ms
4  vl-3010.csr1-2wr.lrz.de (129.187.0.149)  5.082 ms  3.387 ms  4.694 ms
5  cr-gar1-be2-147.x-win.dfn.de (188.1.37.89)  3.254 ms  3.274 ms  2.967 ms
6  cr-fra2-hundredgige0-0-0-3.x-win.dfn.de (188.1.144.253)  13.139 ms  12.260 ms  15.702 ms
7  dfn.mx1.fra.de.geant.net (62.40.124.217)  11.365 ms  11.716 ms  16.314 ms
8  ae1.mx1.gen.ch.geant.net (62.40.98.108)  19.889 ms  26.193 ms  19.661 ms
9  ae4.mx1.par.fr.geant.net (62.40.98.152)  28.465 ms  27.664 ms  29.365 ms
10  et-3-1-0.102.rtsw.newy32aoa.net.internet2.edu (198.71.45.236)  104.199 ms  104.173 ms  109.925 ms
11  nox300gw1-i2-re.nox.org (192.5.89.221)  111.437 ms  110.232 ms  109.370 ms
12  umass-re-nox300gw1.nox.org (192.5.89.102)  113.755 ms  115.848 ms  110.634 ms
13  core1-rt-xe-0-0-0.gw.umass.edu (192.80.83.101)  118.469 ms  119.070 ms  114.279 ms
14  lgrc-rt-106-8-po-10.gw.umass.edu (128.119.0.233)  111.948 ms  111.992 ms  111.616 ms
15  128.119.3.32 (128.119.3.32)  112.194 ms  124.315 ms  111.624 ms
16  nscs1bbs1.cs.umass.edu (128.119.240.253)  114.384 ms  166.509 ms  113.220 ms
17  gaia.cs.umass.edu (128.119.245.12)  130.574 ms !Z  114.883 ms !Z  116.865 ms !Z
```

Listing 2: Sample output of traceroute

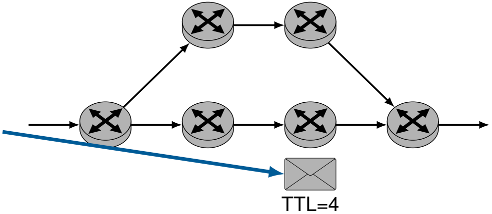
Traceroute: possible anomalies due to load balancing



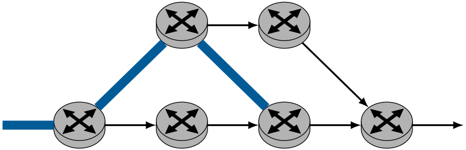
Traceroute: possible anomalies due to load balancing



Traceroute: possible anomalies due to load balancing



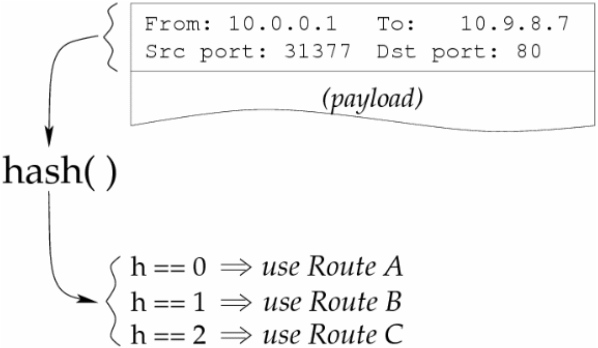
Traceroute: possible anomalies due to load balancing



Load balancing

Per Connection Load balancing:

- Hash *consistently* and use packet headers as *random* values
  - Packets from same TCP connection yield same hash value
  - No reordering within one TCP connection



Paris Traceoute

Idea: Vary header fields that are within the first 28 octets

- TCP: sequence number
- UDP: checksum field
  - Requires manipulation of payload to ensure correctness of checksum
- ICMP: combination of ICMP identifier and sequence number

Experiment results

- Certain routers use first four octets after IP header combined with IP fields for load balancing

Still fails on per packet load balancing

- MDA [1] tries to cover this problem

Further Traceroutes

There are further interesting traceroute tools, e.g.:

- yarrp [2]
  - Stateless
  - Highly parallel
- Scamper [3]
  - All-in-one tool
  - IPv4 & IPv6
  - Built-in alias resolution
- MDA [1]
  - Tries to identify all possible paths
  - Crafts specific packets to find new paths
  - Large overhead
- MDA-Lite [4]
  - Optimized MDA implementation
  - Trade off between performance and completeness

Nmap

Open-source network mapping tool

- https://nmap.org/
- First version in 1997

Modes of operation:

- Host discovery
- Service detection
- OS detection
- Execution of custom scripts

Nmap - Scanning Techniques

- TCP RAW socket scans with certain flags
  - SYN: Find open ports
  - NULL/FIN/Xmas:
    - According to RFC 793 all packets without SYN, ACK, RST result in RST if port is closed, and no response if port is open
    - NULL: No bit set
    - FIN: Only FIN set
    - Xmas: FIN+PUSH+URG
  - ACK: Determine filtered/unfiltered ports in a firewall
  - Window: Same as ACK, lists responses with Window > 0 in RST as open (implementation on certain firewalls)
  - Maimon: Send FIN+ACK, according to RFC 793 all hosts should respond with RST, no matter if port is open or closed
- TCP connect scans
- ICMP ping scan
- UDP payload scan

Nmap - Performance

Internet-wide scans using Nmap:

- Stateful scanning approach
  - Nmap keeps state for every packet in transit
  - Catch timeouts and send retry packets
- Performance
  - Full scan from one system takes 10 days (4k IP addr/sec) [5]
  - 25 Amazon EC2 instances $\rightarrow$ 25 hours (1.6k IP addr/sec) [6]
  - Typically 1 packet sent and 1 packet received per IP addr

ZMap

Adaptation of Nmap for Internet-wide scans

- https://zmap.io/
- Developed at the University of Michigan [7]
- First port-scanner to saturate 1 Gbit/s link: 1.4 Mpps
- Scan entire Internet in 45 minutes
- Later tweaked to saturate 10 Gbit/s link [8]: 14 Mpps

ZMap

Internet-wide scans

- Use TCP SYN or UDP payload scan to find open ports
- Input randomization
  - Pseudo-random number generator
  - Based on multiplicative group of integers modulo p ($2^{32} + 15$)
  - Map 32-bit integer to IPv4 address
- Possible to use multiple worker nodes (shards) on different machines
  - IP will only be scanned once in complete scan

ZMap - Approach

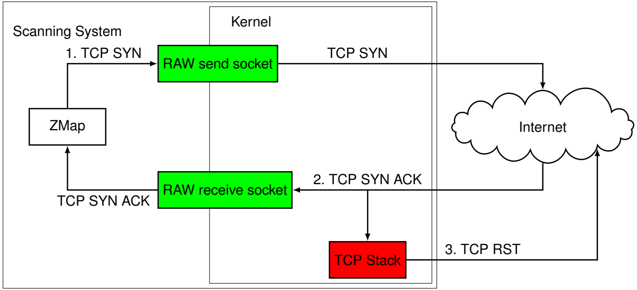
Stateless scanning

- No state for sent packets kept
- Timeout detection not possible
- How to identify responses belonging to scan?
  - Use IP ID = 54321
  - Generate validation based on packet input (e.g. destination IP) using AES
  - Store validation in packet which will be sent (e.g. in sequence number)
  - Validate validation (e.g. sequence number – 1) in received packet

ZMap - Approach

Separate send and receive threads using RAW sockets

- Use RAW socket to directly send and receive packets without kernel TCP stack
- No locking needed
- ZMap send and receive behavior:



ZMap - Approach

Separate probe and output modules

- Probe modules
  - Implement scanning technique
  - E.g. TCP SYN, TCP SYN-ACK, UDP payload
- Output modules
  - Implement processing and output of received responses
  - E.g. IP address only, CSV, database

ZMap - Additional Tools

ZMap is the basis of a large set of additional tools[1]:

- ZGrab
  - Stateful application-layer scanner
  - e.g. for HTTPS, SSH, BACNET
- ZDNS
  - utility for fast DNS lookups
- ZCrypto
  - TLS and X.509 library
  - Certificate parsing and TLS handshake transcription

1 https://zmap.io/

IPv4 ZMap Scans

State of the art:

- Full "0/0" scans

IPv4 ZMap Scans

State of the art:

- Full "0/0" scans
- Out of 4 B addresses only ~ 3.2 B are publicly reachable
    - Excludes private, reserved or announced addresses

IPv4 ZMap Scans

State of the art:

- Full "0/0" scans
- Out of 4 B addresses only ~ 3.2 B are publicly reachable
  - Excludes private, reserved or announced addresses
- Feasible with Nmap/ZMap
  - ZMap scan rate: 20k IP addr/s $\rightarrow$ 37h

State of the art:

- Full "0/0" scans
- Out of 4 B addresses only ~ 3.2 B are publicly reachable
  - Excludes private, reserved or announced addresses
- Feasible with Nmap/ZMap
  - ZMap scan rate: 20k IP addr/s → 37h
- ZMap only provides information whether the address is responsive
  - e.g., an ICMP Ping is possible or a TCP Handshake

→ No information whether an actual service is available
  - Protocol-specific scanners for stateful protocols are required
- Continuous scans to observe changes in the network and deployment

IPv4 ZMap Scans

TCP Port Scan results:

- Conducted from a single vantage point
- First week of August 2022

Service	Port	Responsive
HTTP	80	63 185 323
HTTPS	443	55 797 463
CPE WAN Management	7547	43 118 258
SSH	22	25 612 566
SMTP	25	15 298 930
FTP	21	12 695 736
Alternative HTTP	8080	11 828 087
DNS	53	10 215 627
RDP	3389	8 135 255
Ephemeral Port	60000	7 332 835

IPv4 ZMap Scans

Distribution across the Internet

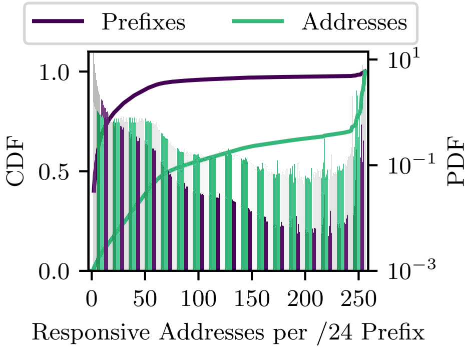
- Based on /24 prefixes
- The smallest prefix routed on the Internet (within BGP)

IPv4 ZMap Scans

Distribution across the Internet

- Based on /24 prefixes
- The smallest prefix routed on the Internet (within BGP)

Port 443:

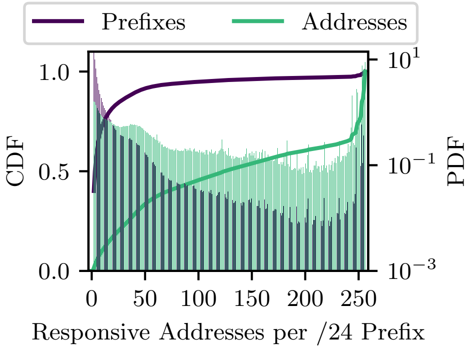


IPv4 ZMap Scans

Distribution across the Internet

- Based on /24 prefixes
- The smallest prefix routed on the Internet (within BGP)

Port 80:

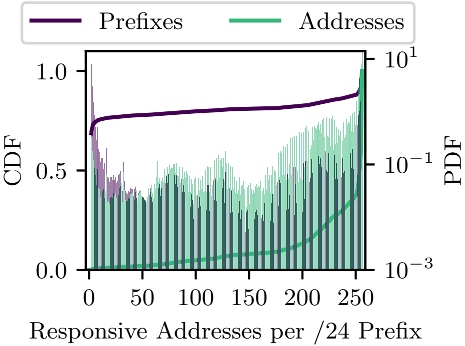


IPv4 ZMap Scans

Distribution across the Internet

- Based on /24 prefixes
- The smallest prefix routed on the Internet (within BGP)

Port 60000:



IPv4 ZMap Scans

Why are more than 90% of addresses responsive for some /24 prefixes?

- In some cases all addresses are used by individual servers.
- But other reasons can potentially be:

IPv4 ZMap Scans

Why are more than 90% of addresses responsive for some /24 prefixes?

- In some cases all addresses are used by individual servers.
- But other reasons can potentially be:
- Tarpits
  - Each address is responsive to slow down scanners

IPv4 ZMap Scans

Why are more than 90% of addresses responsive for some /24 prefixes?

- In some cases all addresses are used by individual servers.
- But other reasons can potentially be:
- Tarpits
  - Each address is responsive to slow down scanners
- Proxies/Middleboxes
  - Devices terminate TCP handshakes for all addresses
  - Decide whether to drop or where to route traffic depending on higher layer services

IPv4 ZMap Scans

Why are more than 90% of addresses responsive for some /24 prefixes?

- In some cases all addresses are used by individual servers.
- But other reasons can potentially be:
- Tarpits
  - Each address is responsive to slow down scanners
- Proxies/Middleboxes
  - Devices terminate TCP handshakes for all addresses
  - Decide whether to drop or where to route traffic depending on higher layer services
- CDNs, e.g., Cloudflare's addressing agility approach [9]
  - This technique decouples IP addresses from domain names and services.
  - The authoritative name server can select the addresses in the query response from a full prefix.
  - Used for on-demand, flexible load balancing.

ZMapv6

Original ZMap implementation supports only IPv4

- Extension of ZMap with IPv6 capabilities $\rightarrow$ ZMapv6
- https://github.com/tumi8/zmap
- Adaptation of scanning core to send and receive IPv6 packets
- Port probe modules for IPv6 scanning: ICMPv6, TCP over IPv6, UDP over IPv6

IPv6 Scans

Challenges

- Vast address space → "0/0" scan not possible
  - Scan rate 20k IP addr/s → 5.4×10^{26} years

→ Hitlists are required

IPv6 Scans

Challenges

- Vast address space → “0/0” scan not possible
  - Scan rate 20k IP addr/s → 5.4×10^{26} years

→ Hitlists are required

A list of **targets**, most likely **responsive**, of **feasible size**.

IPv6 Scans

Challenges

- Vast address space → "0/0" scan not possible
  - Scan rate 20k IP addr/s → 5.4×10^{26} years

→ Hitlists are required

A list of **targets**, most likely **responsive**, of **feasible size**.

Responsive:

- Responsive to at least one protocol (e.g., ICMP, HTTP, ...)
- Different between addresses
- Changes over time

IPv6 Scans

Challenges

- Vast address space → “0/0” scan not possible
  - Scan rate 20k IP addr/s → 5.4×10^{26} years

→ Hitlists are required

A list of **targets**, most likely **responsive**, of **feasible size**.

Responsive:

- Responsive to at least one protocol (e.g., ICMP, HTTP, . . .)
- Different between addresses
- Changes over time

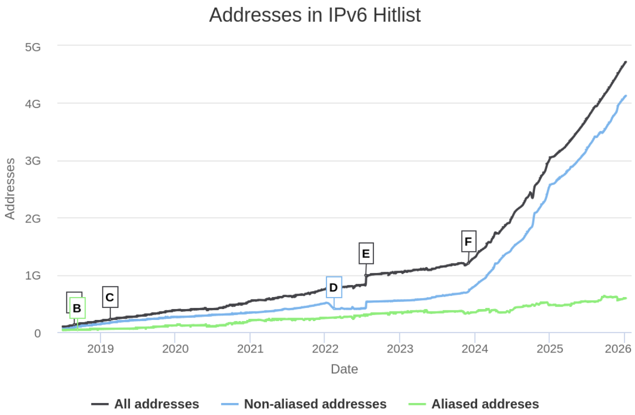
Feasible size:

- Scan duration
- Bandwidth limitations

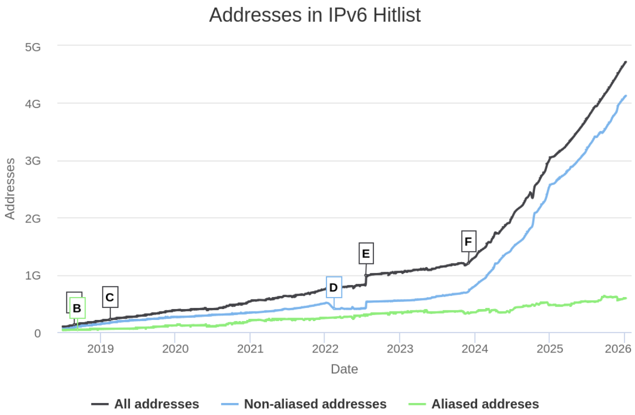
IPv6 Scans

- Research at this Chair
- Aggregates multiple inputs
- Filters aliased prefixes and applies blocklists
- Tests reachability daily
  - ICMPv6
  - TCP/80 (HTTP)
  - TCP/443 (HTTPS)
  - UDP/53 (DNS)
  - UDP/443 (QUIC)
- Uses ZMapv6

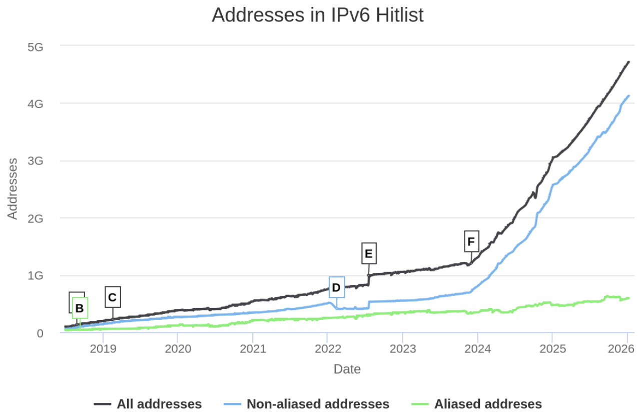
IPv6 Scans
Input



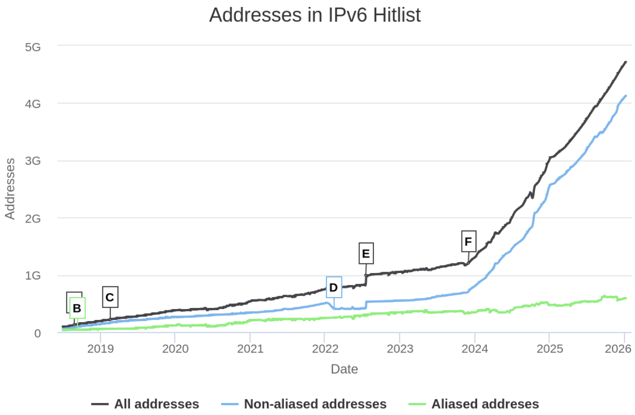
IPv6 Scans
Input



- A: Addition of IPv6 rDNS

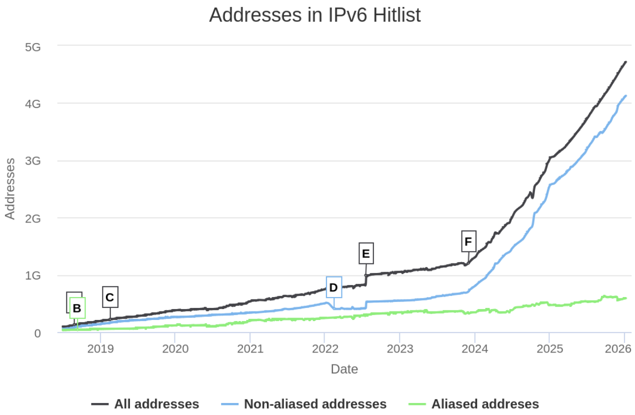


- A: Addition of IPv6 rDNS
- B: Withdrawal of two Amazon EC2 prefixes



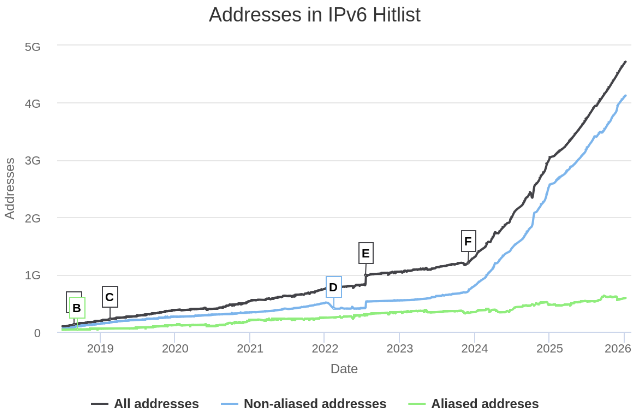
- A: Addition of IPv6 rDNS
- B: Withdrawal of two Amazon EC2 prefixes
- C: Additional IPv6 rDNS results

IPv6 Scans
Input

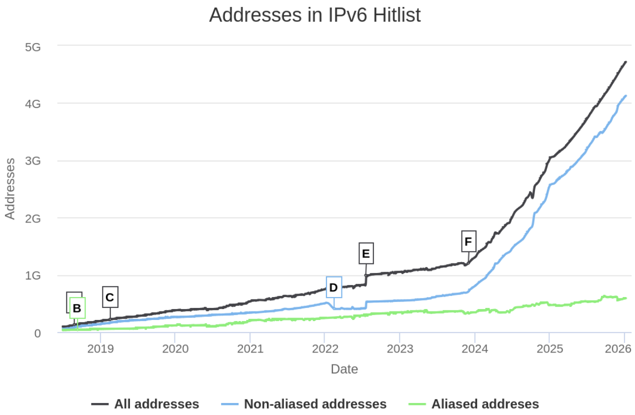


- A: Addition of IPv6 rDNS
- B: Withdrawal of two Amazon EC2 prefixes
- C: Additional IPv6 rDNS results
- D: Removal of addresses impacted by the GFW

IPv6 Scans
Input



- A: Addition of IPv6 rDNS
- B: Withdrawal of two Amazon EC2 prefixes
- C: Additional IPv6 rDNS results
- D: Removal of addresses impacted by the GFW
- E: Addition of new addresses from passive sources and target generation methods



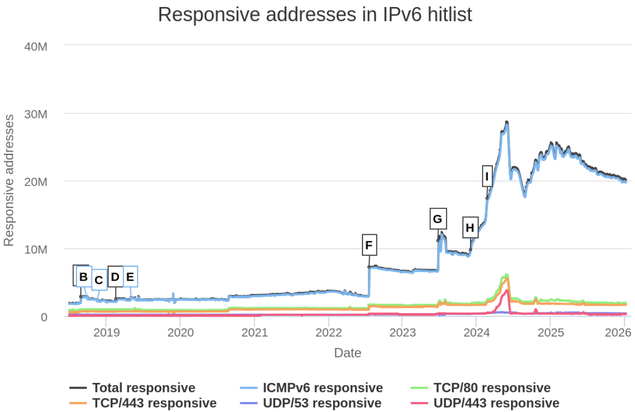
- A: Addition of IPv6 rDNS
- B: Withdrawal of two Amazon EC2 prefixes
- C: Additional IPv6 rDNS results
- D: Removal of addresses impacted by the GFW
- E: Addition of new addresses from passive sources and target generation methods
- F: Addition of IPinfo as external data provider

Filter

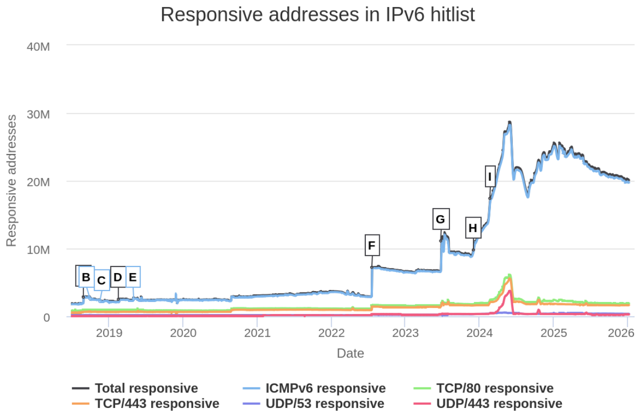
The input has to be filtered by several steps before the responsiveness can be tested.

- Not globally routed
  - Datasets might contain addresses that are not routed
  - Infrastructure changes might change the reachability of addresses
- Blocklisted
  - Aggregated list of blocklisting requests from all scans at our chair
- Aliased prefixes
- Not responsive for 30 consecutive days

IPv6 Scans
Results

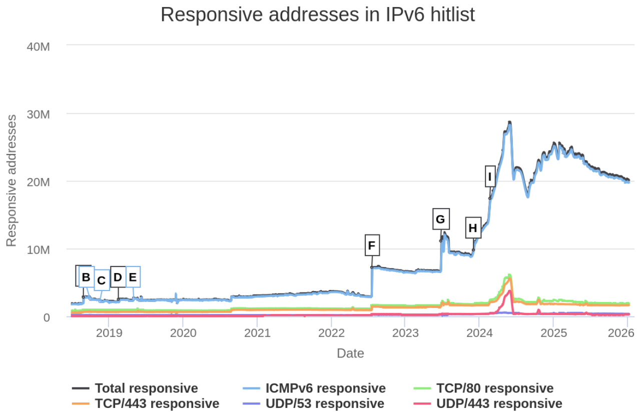


Results



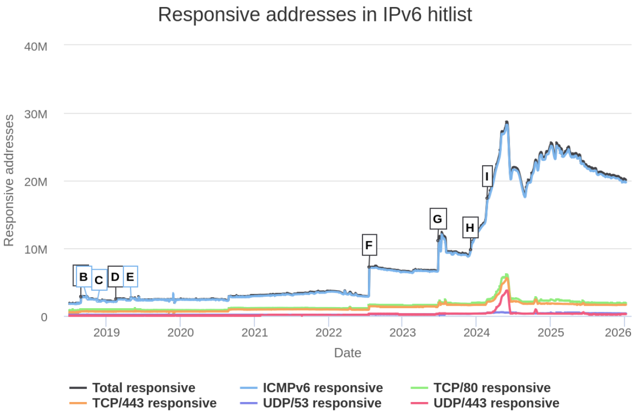
- A, D, E: Addition of IPv6 rDNS

Results



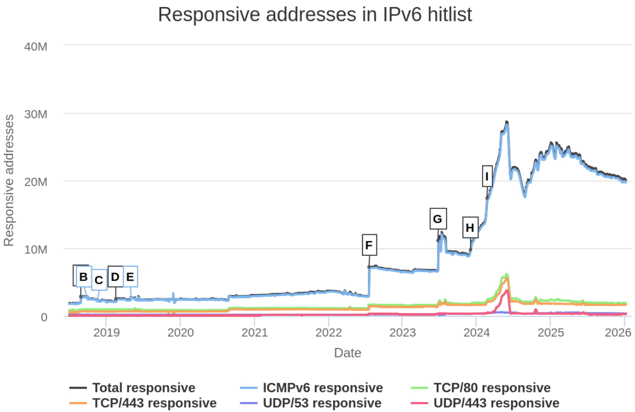
- A, D, E: Addition of IPv6 rDNS
- B: LATNET SERVISS stopped responding

IPv6 Scans
Results



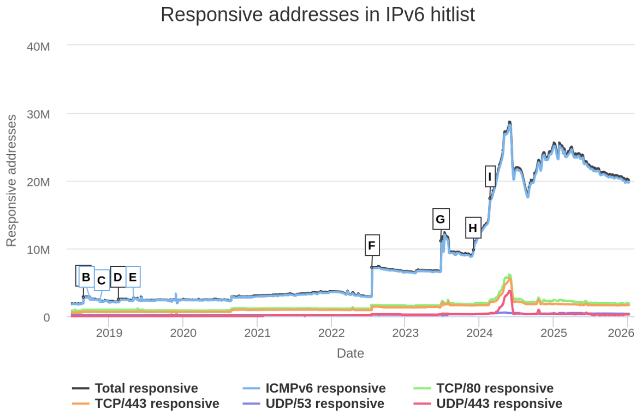
- A, D, E: Addition of IPv6 rDNS
- B: LATNET SERVISS stopped responding
- C: Online S.A.S. stopped responding

IPv6 Scans
Results



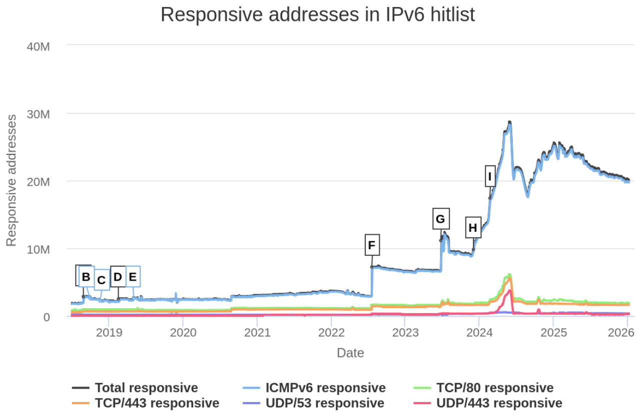
- A, D, E: Addition of IPv6 rDNS
- B: LATNET SERVISS stopped responding
- C: Online S.A.S. stopped responding
- F,G: Addition of new address candidates based on target generation algorithms

IPv6 Scans
Results



- A, D, E: Addition of IPv6 rDNS
- B: LATNET SERVISS stopped responding
- C: Online S.A.S. stopped responding
- F,G: Addition of new address candidates based on target generation algorithms
- H: Addition of ipinfo.io as new data collaborator

IPv6 Scans
Results



- A, D, E: Addition of IPv6 rDNS
- B: LATNET SERVISS stopped responding
- C: Online S.A.S. stopped responding
- F,G: Addition of new address candidates based on target generation algorithms
- H: Addition of ipinfo.io as new data collaborator
- I: Reevaluation of addresses filtered after being 30-days unresponsive

New Sources

While the existing IPv6 Hitlist sources regularly update the input, new sources have not been added.

We evaluated different approaches to extend our hitlist [10]:

- Target Generation:
  - 6Tree, 6Graph, 6GAN, 6VecLM,
  - Distance Clustering (DC) (our custom algorithm)
- 30-day unresponsive addresses

		Responsive	
Method	Addr	Addr. ↓	ASes
6Graph	125.8 M		
6Tree	37.6 M		
DC	5.3 M		
6GAN	3.3 M		
6VecLM	70.3 k		
30-day Unresp.	405.0 M		

New Sources

While the existing IPv6 Hitlist sources regularly update the input, new sources have not been added.

We evaluated different approaches to extend our hitlist [10]:

- Target Generation:
  - 6Tree, 6Graph, 6GAN, 6VecLM,
  - Distance Clustering (DC) (our custom algorithm)
- 30-day unresponsive addresses

Method	Addr	Responsive Addr. ↓	Responsive ASes
6Graph	125.8 M	3.8 M	10.7 k
6Tree	37.6 M	2.2 M	11.5 k
DC	5.3 M	651.9 k	5.5 k
6GAN	3.3 M	4.3 k	39
6VecLM	70.3 k	1.0 k	105
30-day Unresp.	405.0 M	1.3 M	9.0 k

→ All sources contribute additional responsive addresses.
→ We identified 5.6 M new responsive IPv6 addresses from 14.6 k ASes.

Hitlists:

- Lots of possible sources
- Knowledge about sources is important
- Number of IP addresses is not only metric $\rightarrow$ evaluate reachability and stability
- Optimal sources depend on type of measurement (end-user devices, servers, routers,. . .)
- Be aware of biases in your hitlist (address distribution, prefix/AS distribution, aliased prefixes)

Is client tracking impossible?

The privacy extension prevents tracking of clients by randomization of the interface ID.

- In general, most end devices implement the privacy extension
- 90% of IPv6 addresses seen by a large CDN are only seen once in long-running analyses [11]

Is client tracking impossible?

The privacy extension prevents tracking of clients by randomization of the interface ID.

- In general, most end devices implement the privacy extension
- 90% of IPv6 addresses seen by a large CDN are only seen once in long-running analyses [11]
- But how about Customer Premises Equipment (CPE)?
  - e.g., private networks use a single, fixed router (the CPE) as gateway to the Internet

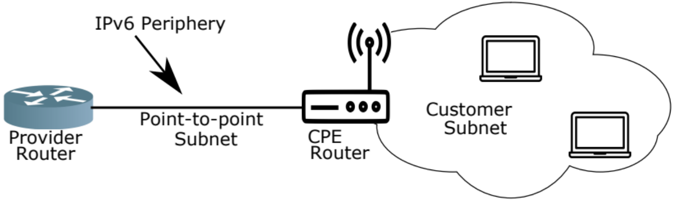


Figure 1: Common IPv6 architecture [12]

Is client tracking impossible?

The privacy extension prevents tracking of clients by randomization of the interface ID.

- In general, most end devices implement the privacy extension
- 90% of IPv6 addresses seen by a large CDN are only seen once in long-running analyses [11]
- But how about Customer Premises Equipment (CPE)?
  - e.g., private networks use a single, fixed router (the CPE) as gateway to the Internet
  - → In 2020, an approach to discover the IPv6 periphery was presented [12]
  - → A measurement revealed 64.8M router addresses
  - → 30M addresses were found using EUI-64

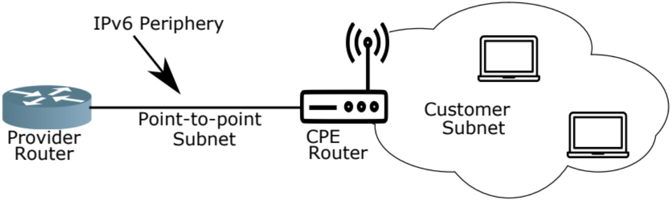


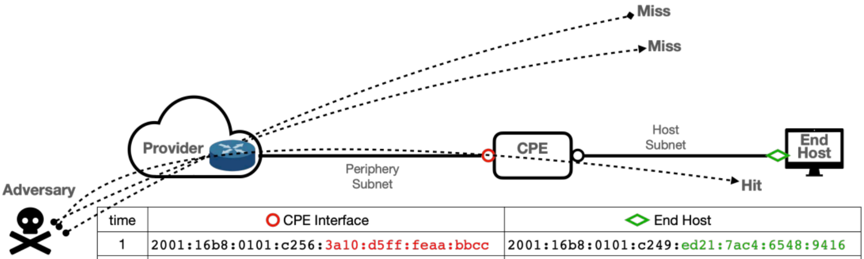
Figure 1: Common IPv6 architecture [12]

Is client tracking impossible?

- To prevent tracking based on the assigned prefix, providers often rotate their assignments

Is client tracking impossible?

- To prevent tracking based on the assigned prefix, providers often rotate their assignments
- But behavioral analysis of providers and CPE’s using EUI-64 can be used to track prefixes [13]
- While clients can not be tracked, CPEs using EUI-64 identifiers can be actively found.



time	CPE Interface	End Host
1	2001:16b8:0101:c256:3a10:d5ff:feaa:bbcc	2001:16b8:0101:c249:ed21:7ac4:6548:9416
2	2001:16b8:0103:74fe:3a10:d5ff:feaa:bbcc	2001:16b8:0103:7421:42c1:02b5:5ff3:4bec
3	2001:16b8:0101:1f20:3a10:d5ff:feaa:bbcc	2001:16b8:0101:1529:ac1a:1e66:0801:c844

Figure 2: CPE with missing privacy extension [13]

Is client tracking impossible?

- In theory, testing all targets in a /48 is infeasible
- How can we reduce the search space and effectively track EUI-64 identifier?
  - Customers receive at least /64 prefixes and often larger
  - Providers often use only parts of their owned prefixes
  - Prefixes are often assigned at nibbles (e.g., /56, /60, /64)

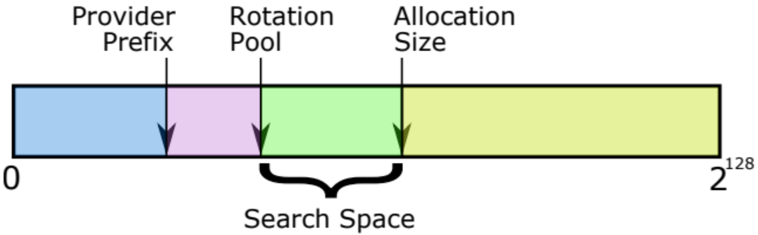
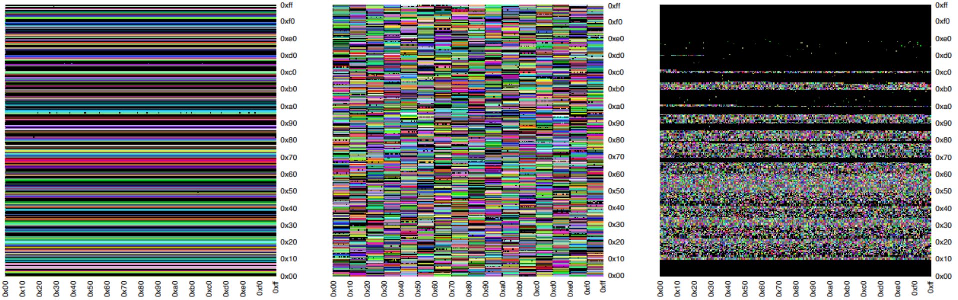


Figure 3: Limiting the search space to track IPv6 hosts: for a provider, infer the : i) size of the allocations to customers; and ii) range of prefixes used for rotation. [13]

Is client tracking impossible?

- A recent study reveals that high speed probing, behavioral analysis of providers and CPE's using EUI-64 can be used to track prefixes [13]
- Different allocation schemes can be found, e.g., /56 or /64 assignments



(a) Entel (Bolivia): /56 allocations **(b) BH Telecom (Bosnia): /60 allocations** **(c) Starcat (Japan): /64 allocations**

Figure 4: The y-axis of plots represents the 7^{th} byte of a probed address, while the x-axis denotes the 8^{th} byte; each pixel represents a probed /64 network. Each color represents a different responsive source address, while black indicates no response was received when probing to an address in that /64 network. [13]

Is client tracking impossible?

- Active scans can be used to identify specific CPEs even if providers rotate their assignments
- Based on a reduced search space, an adversary can effectively implement tracking

→ The tracking relies on CPEs using EUI-64

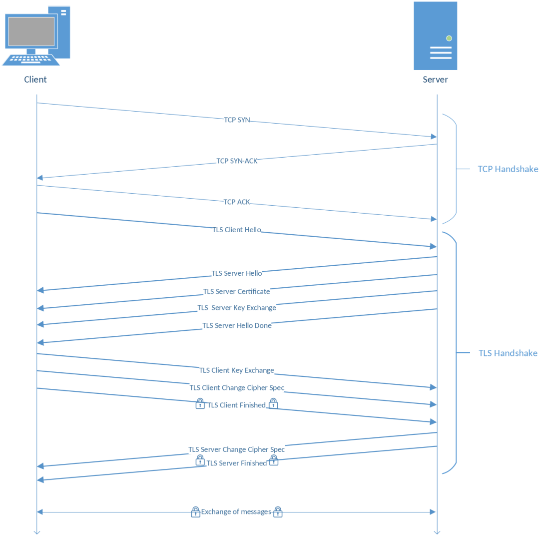
→ CPE manufacturers should change the device behavior

- The given paper resulted in a change of behavior within a large manufacturer
- For more details see the original work [13]

TLS

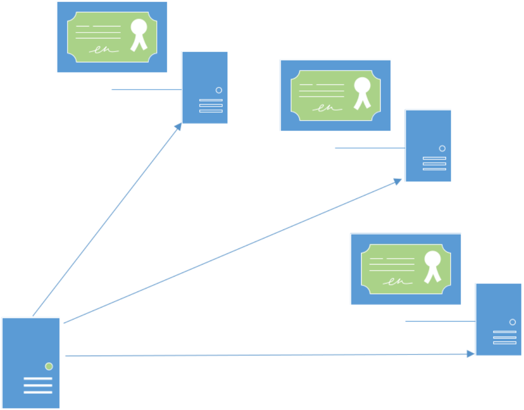
- TLS: Transport Layer Security
  - SSL 3.0
  - TLS 1.0
  - TLS 1.1
  - TLS 1.2
  - TLS 1.3
- Security foundation for HTTPS, IMAPS, SMPTS, DoT, DoH, . . .

→ Evaluate TLS Deployment



Certificate Scanning

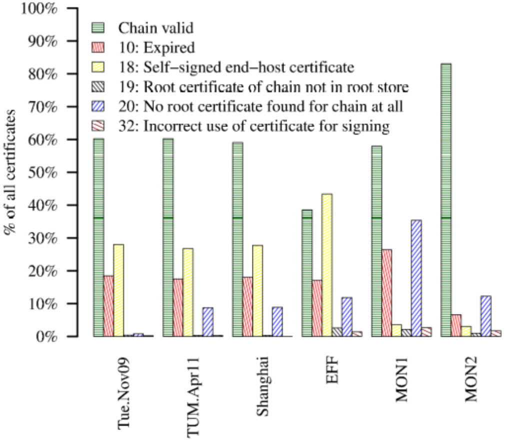
- Methodology
  1. Identify hosts offering TLS service (HTTPS, IMAPS,. . .)
  2. Download certificate chains
  3. Analyze and validate chains
- Challenges
  - Targets (0/0?)
  - Performance
  - Evaluation metrics



Certificate Scanning

Analysis of the TLS landscape [14]

- Active and passive measurements
  1. Analyses of certificate chains
  2. Expiry
  3. Algorithms
- Conclusion:
  - TLS landscape in sorry state (expired, no root cert, . . .)
  - But: situation improves over time [15]



Evolution of TLS Scanning

	Holz et al. (2011) [14]	Now
Targets	• Alexa Top 1M	• Full IPv4 & IPv6 hitlist

Evolution of TLS Scanning

	Holz et al. (2011) [14]	Now
Targets	• Alexa Top 1M	• Full IPv4 & IPv6 hitlist
Server Name Indication (SNI)	• Not used	• Alexa Top 1M • $>$ 1000 TLD Zone files • Reverse DNS

Evolution of TLS Scanning

	Holz et al. (2011) [14]	Now
Targets	• Alexa Top 1M	• Full IPv4 & IPv6 hitlist
Server Name Indication (SNI)	• Not used	• Alexa Top 1M • $>$ 1000 TLD Zone files • Reverse DNS
Software stack	• Nmap • OpenSSL	• ZMap • Custom-built scanner for TLS and HTTPS

Evolution of TLS Scanning

	Holz et al. (2011) [14]	Now
Targets	• Alexa Top 1M	• Full IPv4 & IPv6 hitlist
Server Name Indication (SNI)	• Not used	• Alexa Top 1M • $>$ 1000 TLD Zone files • Reverse DNS
Software stack	• Nmap • OpenSSL	• ZMap • Custom-built scanner for TLS and HTTPS
Performance	• Weeks for 1M hosts	• Day(s) for complete Internet (several hundert millions of hosts)

Evolution of TLS Scanning

	Holz et al. (2011) [14]	Now
Targets	• Alexa Top 1M	• Full IPv4 & IPv6 hitlist
Server Name Indication (SNI)	• Not used	• Alexa Top 1M • $>$ 1000 TLD Zone files • Reverse DNS
Software stack	• Nmap • OpenSSL	• ZMap • Custom-built scanner for TLS and HTTPS
Performance	• Weeks for 1M hosts	• Day(s) for complete Internet (several hundert millions of hosts)
Frequency	• Single measurements	• Continuously running measurement service

TLS 1.3

New features in TLS 1.3

- 1-RTT handshakes by default
  - Use presumed cipher suite selection
- 0-RTT handshake with resumption possible
  - PSK for early data
  - Forward secrecy after early data
- Privacy
  - Client certificates are encrypted
  - SNI not encrypted (RFC Draft for encrypted Client Hello in TLS 1.3)
- Grease mechanism
  - Send random version data to increase robustness

QUIC Measurements

- **2013** Google introduces QUIC
- **2016** First tracked IETF draft is published
  and a dedicated working group is created
- **2018** In *"A First Look at QUIC in the Wild"*, Rüth et al. [16] report widespread deployment of Google QUIC mainly by large providers
- **2020** QUIC carries a third of Google traffic[2]
  75 % of Facebook's traffic is HTTP/3 and QUIC[3]
- RFC9000 is released on May 27, 2021

1 https://blog.chromium.org/2020/10/chrome-is-deploying-http3-and-ietf-quic.html

2 https://engineering.fb.com/2020/10/21/networking-traffic/how-facebook-is-bringing-quic-to-billions/

3 https://hacks.mozilla.org/2021/04/quic-and-http-3-support-now-in-firefox-nightly-and-beta/

As a new fundamental network protocol with widespread early adoption, QUIC requires early analysis and researchers tools to analyze QUIC deployments.

→ We provided an Internet-wide measurement study shortly before the final RFC release [17]

Research Questions:

1. How can we detect QUIC deployments?
   - → IPv4 + IPv6 ZMap modules
   - → HTTPS DNS RR
   - → HTTP ALT-SVC header
2. Who deploys QUIC?
3. Which QUIC versions are deployed?
4. Can we successfully connect to QUIC servers and analyze deployments?
   - → We developed and published the QScanner, a highly parallelized stateful QUIC scanner

How can we detect QUIC deployments?

ZMap module:

- QUIC relies on UDP
  - → ZMap needs to send valid QUIC packets
- Relies on the QUIC version negotiation
  - Server responses should contain all supported versions
  - No state is created at the server
  - No computational expensive cryptography is necessary
- Requires no input (at least for IPv4)
- ZMap reports most addresses supporting the QUIC version negotiation
  - Domains can be mapped to only 10 % of addresses

		Scanned		Results	
		Targets	Addresses	ASes	Domains
ZMap	IPv4	3 023 298 514	2 134 964	4736	30 970 316
	IPv6	24 434 296	210 997	1704	17 972 799

How can we detect QUIC deployments?

HTTPS DNS Resource Records

- Based on a new IETF draft [18]
- Specifies DNS resource records to provide service information
  - Can include ALPN values indicating QUIC support
  - `simple.example 7200 IN HTTPS 1 . alpn=h3`
- Requires domains to resolve

→ HTTPS DNS RRs results in the fewest amount of deployments

		Scanned Targets	Results Addresses	ASes	Domains
HTTPS	IPv4	213 689 057	85 092	1287	2 962 708
	IPv6		69 684	112	2 736 040

How can we detect QUIC deployments?

HTTP ALTSVC Headers

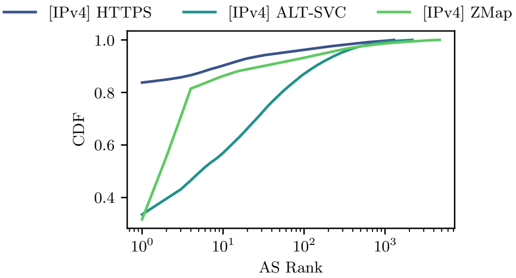
- HTTP header containing alternative service information
  - Can include ALPN values indicating QUIC support
  - `alt-svc: h3=":443"; ma=86400, h3-29=":443"; ma=86400, h3-28=":443"; ma=86400, h3-27=":443"; ma=86400`
- Requires HTTP(s) capable targets and scans
- ALT-SVC reveals the most domains with QUIC support

		Scanned		Results	
		Targets	Addresses	ASes	Domains
ALT-SVC	IPv4	375 338 772	232 585	2174	36 907 770
	IPv6	69 458 318	283 169	292	16 979 759

Who deploys QUIC?

To analyze who is involved in the deployment of QUIC, we analyzed originating ASes:

- Deployments are dominated by large providers
- ZMap results in addresses located in more than 4.7 k ASes
- HTTPS DNS Resource Records are strongly biased towards Cloudflare

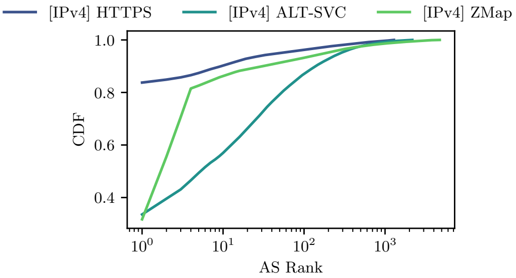


Rank	Provider	ZMap #IPv4 Addr.	#Domains
1	Cloudflare	676 483	23 843 989
2	Google	510 450	6 006 547
3	Akamai	320 646	23 206
4	Fastly	232 776	938 649
5	Cloudflare London	23 489	61 979

Who deploys QUIC?

To analyze who is involved in the deployment of QUIC, we analyzed originating ASes:

- Deployments are dominated by large providers
- ZMap results in addresses located in more than 4.7 k ASes
- HTTPS DNS Resource Records are strongly biased towards Cloudflare

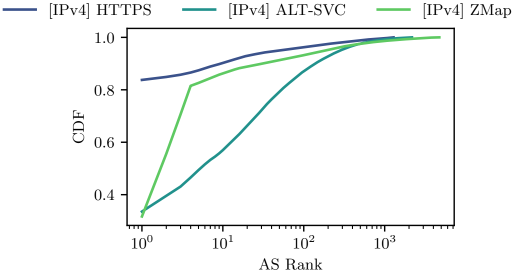


Rank	Provider	HTTPS DNS RR #IPv4 Addr.	#Domains
1	Cloudflare	71 278	2 887 327
2	DigitalOcean	969	1256
3	Google	719	1235
4	Amazon	709	814
5	OVH	708	1034

Who deploys QUIC?

To analyze who is involved in the deployment of QUIC, we analyzed originating ASes:

- Deployments are dominated by large providers
- ZMap results in addresses located in more than 4.7 k ASes
- HTTPS DNS Resource Records are strongly biased towards Cloudflare

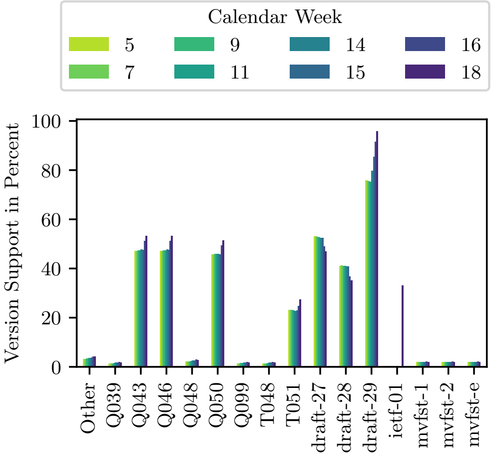


Rank	Provider	ALT-SVC #IPv4 Addr.	#Domains
1	Cloudflare	78 033	19 286 420
2	OVH	14 011	1 691 721
3	GTS Telecom	8160	234 149
4	A2 Hosting	8068	858 932
5	DigitalOcean	6556	135 910

Which QUIC versions are deployed?

We regularly scanned with ZMap between February and May 2021:

- 50 % of found targets still supported Google QUIC versions
- More than 90 % supported the latest draft that should be deployed (Draft-29)
- First deployments announced Version 1 even before the final RFC release



Can we successfully connect to QUIC servers?

QScanner (`https://github.com/tumi8/QScanner`)

- Stateful scanner based on quic-go that conducts full handshakes
- Supports the latest drafts and Version 1
- Allows HTTP requests after successful handshakes
- Extracts widespread information:
  - connection information
  - TLS properties
  - X.509 certificates
  - HTTP headers

→ We are able to successfully complete handshakes with more than 26 M targets

	IPv4 (%)	
	no SNI	SNI
Total Targets	2 M	17M
Success	7.25	76.06
Version Mismatch	8.83	5.77
Timeout	34.50	11.09
Crypto Error (0x128)	48.26	5.73
Other	1.16	1.35

- Low success rate without a server name identifier
- Version mismatches were mainly due to an iterative roll-out of IETF QUIC at Google
  - They do not occur in current scans
- Including the server name identifier drastically increases the success rate
  - Addresses from ZMap without domains have to be treated carefully

Can we identify different QUIC deployments based on configurations?

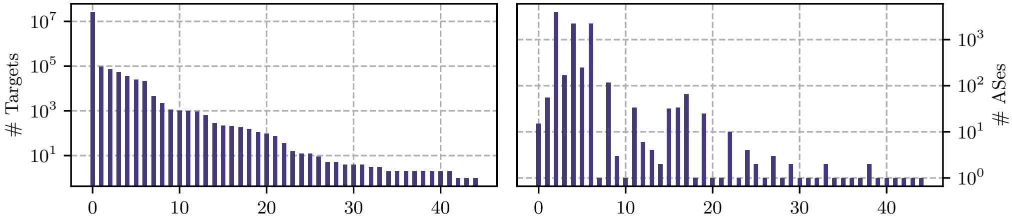
Servers share a set of QUIC Transport Parameters during the handshake:

- 17 different parameters exist, e.g.,
  - initial size of the flow control window
  - the maximum number of allowed streams
- A new TLS extension was defined to send transport parameters (see RFC9001)

→ The QScanner extracts server values

→ Can we identify different QUIC deployments based on configurations?

Can we identify different QUIC deployments based on configurations?



Transport parameters differ within order of magnitudes

- We find 45 different parameter sets
- The most common set is used by Cloudflare and 15 additional ASes
- Three parameter sets are seen in more than 1000 ASes
- Two out of these are seen in combination with a single HTTP Server header value:
  - *proxygen-bolt*

→ These targets are edge PoPs from Facebook and not set up by individuals

Conclusion

- → Different means to detect QUIC deployments exist, each offering unique targets
- → Widespread deployment of QUIC can be found
  - more than 2M addresses in 4700 ASes
- → The overall state was solid and ready for the RFC release
  - 26 M targets result in successful handshakes
  - More than 90 % of targets support the latest draft or version 1
- → Mainly driven by large providers
  - We identified deployments in many ASes as edge PoPs of large providers

Impact of COVID-19 Pandemic on the Internet
Introduction

- Pandemic is a rare and special event
- Work from home and Stay at home orders posed challenges to the Internet
- Fundamental importance of the Internet and digitalization in general to these measures

Impact of COVID-19 Pandemic on the Internet
Introduction

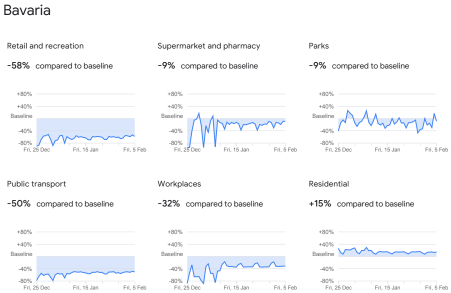
- Pandemic is a rare and special event
- Work from home and Stay at home orders posed challenges to the Internet
- Fundamental importance of the Internet and digitalization in general to these measures
- Expectation
  - Increased load with abnormal patterns and access points
  - Higher load on residential networks
  - General higher load due to higher media consumption and video conferencing

Impact of COVID-19 Pandemic on the Internet
Introduction

- Pandemic is a rare and special event
- Work from home and Stay at home orders posed challenges to the Internet
- Fundamental importance of the Internet and digitalization in general to these measures
- Expectation
  - Increased load with abnormal patterns and access points
  - Higher load on residential networks
  - General higher load due to higher media consumption and video conferencing
- Overall the Internet managed to handle the traffic increase

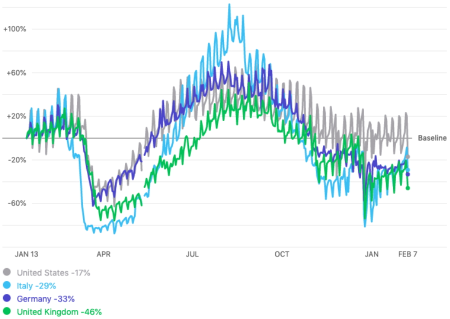
Impact of COVID-19 Pandemic on the Internet
Motivation

- Google and Apple provided mobility reports based on their data
- What is the effect on the Internet?



Google Mobility Report

`https://www.google.com/covid19/mobility/`



Apple Mobility Report

`https://covid19.apple.com/mobility`

Impact of COVID-19 Pandemic on the Internet
IMC 2020

- Early research from IMC 2020[4]
- Submission deadline was in begin of June 2020
- Presentations were in October 2020
- Four interesting papers on the topic:
    - Feldmann et al., The Lockdown Effect: Implications of the COVID-19 Pandemic on Internet Traffic [19]
    - Lutu et al., A Characterization of the COVID-19 Pandemic Impact on a Mobile Network Operator Traffic [20]
    - Fontugne et al., Persistent Last-mile Congestion: Not so Uncommon [21]
    - Böttger et al., How the Internet reacted to Covid-19 – A perspective from Facebook's Edge Network [22]

4 https://conferences.sigcomm.org/imc/2020/

Weekend effect

Approach by Feldmann et al. [19]

- Compared traffic volume throughout the day on a Wednesday and a Saturday, pre and during lockdown

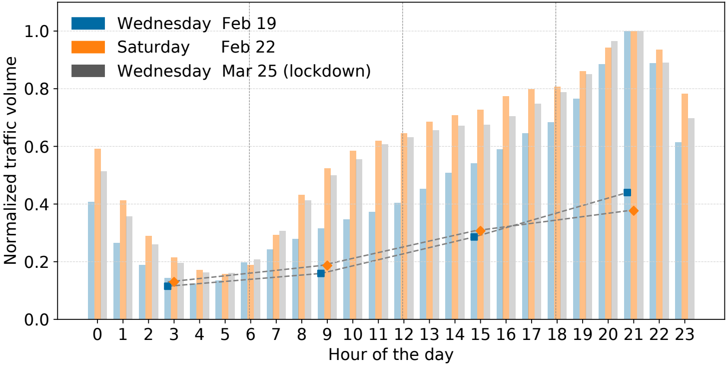


Figure 2a by Feldmann et al. [19]

The Lockdown Effect [19]

ISP Day Patterns

- They used the learned pattern and assigned each day a label
- Blue if the day matches the usual pattern (e.g. Sunday with weekend pattern)
- Orange if it does not match (Wednesday with weekend pattern)
- Data from a Central European ISP

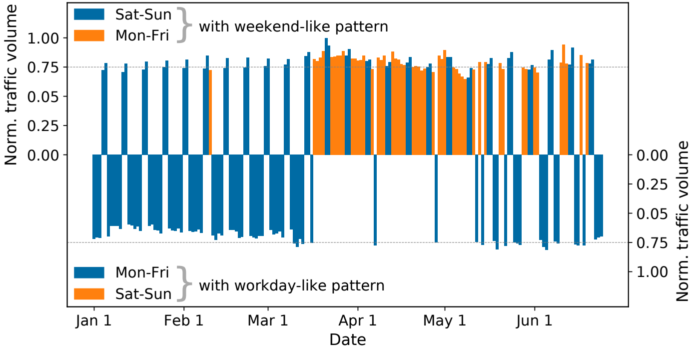


Figure 2b by Feldmann et al. [19]

IXP Day Patterns

- Same approach as before from a Central European IXP

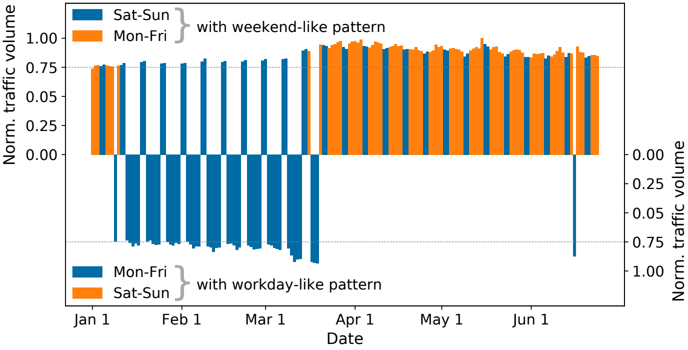


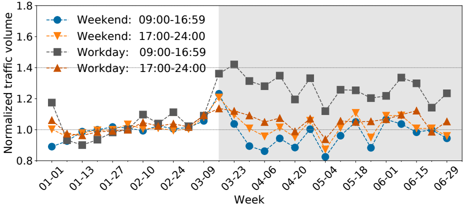
Figure 2c by Feldmann et al. [19]

Hypergiants

Definition

- Originally called so by Arbor networks
- First defined by Labovitz et al. [23]
- Describes companies which generate a disproportionate share of the traffic (high outbound traffic ratios)
- E.g. Google, Netflix, Cloudflare, Akamai

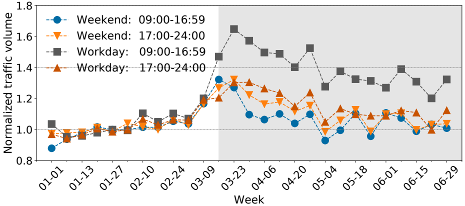
The Lockdown Effect [19]
Hypergiants traffic



Hypergiants traffic. Figure 4a by Feldmann et al. [19]

Analysis by Feldmann et al. [19]

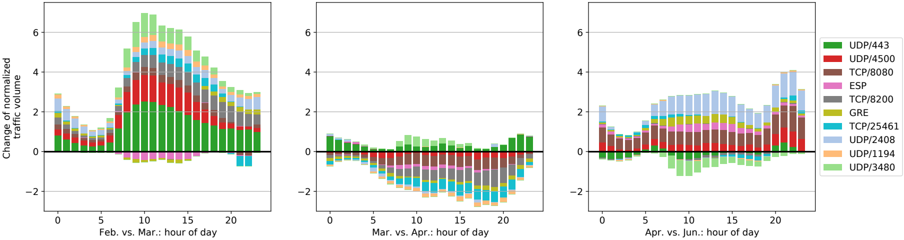
- Used NetFlow and IPFIX data to analyze traffic of hypergiants
- No difference between the four categories until lockdown
- Increase of hypergiants by 40 %
- Other ASes increase by about 60 %



Other ASes traffic. Figure 4b by Feldmann et al. [19]

The Lockdown Effect [19]
Transport Layer Analysis

- By analyzing the used destination ports Feldmann et al. [19] inferred service usage
- UDP/443 is QUIC and mainly used by Google and Akamai
- UDP/4500 is for IPSec NAT traversal
- GRE and ESP transport the real IPSec traffic
  - Usually mainly used between companies
- TCP/8200 and TCP/25461 are used by TV streaming services



IXP in Central Europe. Figure 7 by Feldmann et al. [19]

The Lockdown Effect [19]
Gaming Category

- Filters for 5 ASNs and 57 known gaming related ports
- Used number of IP addresses as an abstraction for households
- Data shown is from an IXP in Southern Europe

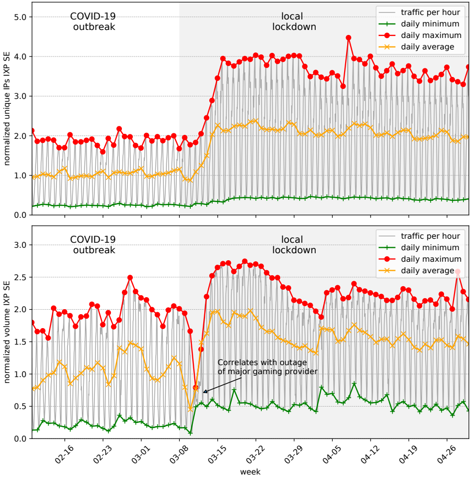


Figure 9 by Feldmann et al. [19]

The Lockdown Effect [19]
All Categories

- All labeled categeries
- Paper also contains the graphs for the Central European IXP and Southern European IXP

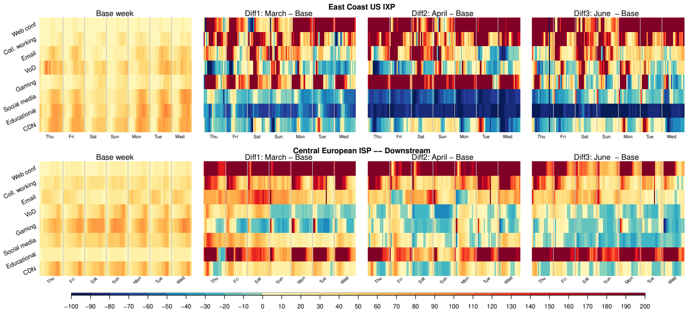


Figure 10 by Feldmann et al. [19]

Characterization of the Pandemic [20]

Analysis of the Mobile Network

Analysis by Lutu et al. [20]:

- Investigated effect on UK Mobile Network Operator (Telefonica)
- E.g.: Used the cell data to quantify mobility
- Can provide local data for cities and city districts
- Expecially analyzed mobility of inner London residents (see figure below)

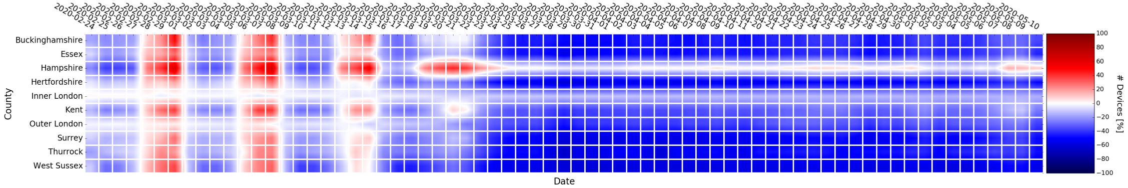


Figure 7 by Lutu et al. [20]

Last-mile Congestion [21]

Inferring Congestion from Traceroutes

Approach by Fontugne et al. [21] to analyze last-mile congestion

- Uses data from RIPE Atlas
- Subtracted latency of last non public routed address from latency of first public routed address
- Apply medians on 30 minute buckets to reduce noise
- Compute queuing delay by observing deviation from minimum median RTT value

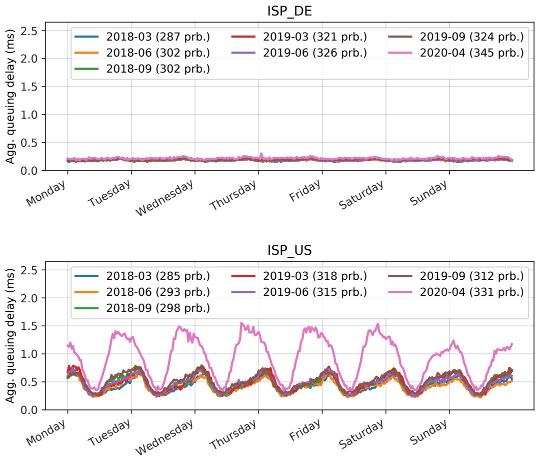


Figure 1 by Fontugne et al. [21]

Last-mile Congestion [21]
Inferring Congestion from Traceroutes

- Uses frequency analysis to find last-mile congestions
- Finds persistent last mile congestion for the US ISP
- Number of congested ASes increseas from 10% to 55% during in April 2020

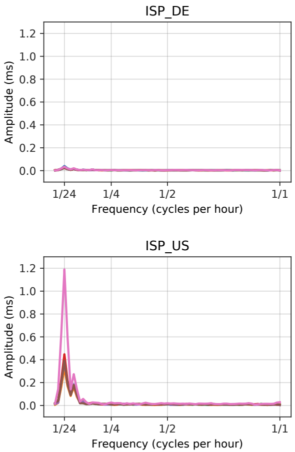
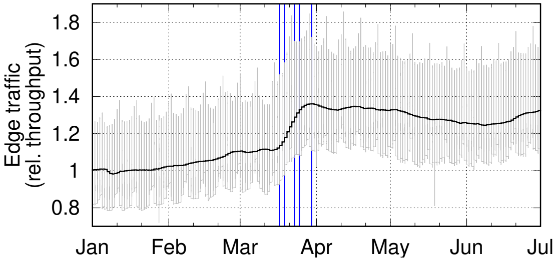


Figure 2 by Fontugne et al. [21]

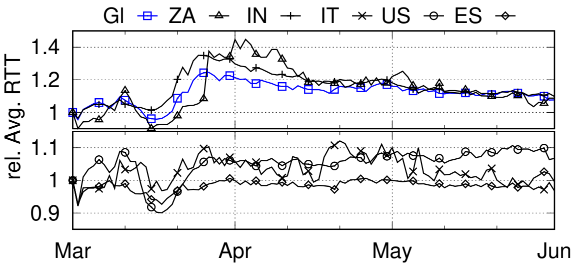
Main Contributions

Approach by Böttger et al. [22]

- Used data collected at Facebooks edge to infer changes



Total traffic growth. Figure 1 by Böttger et al. [22]



Change in latency of selected countries. Figure 11 by Böttger et al. [22]

[1] D. Veitch, B. Augustin, R. Teixeira, and T. Friedman, "Failure control in multipath route tracing," in IEEE INFOCOM 2009, 2009, pp. 1395–1403. DOI: 10.1109/INFCOM.2009.5062055.

[2] R. Beverly, "Yarrp'ing the internet: Randomized high-speed active topology discovery," in Proceedings of the 2016 Internet Measurement Conference, ser. IMC '16, New York, NY, USA: Association for Computing Machinery, 2016, 413–420, ISBN: 9781450345262. [Online]. Available: https://doi.org/10.1145/2987443.2987479.

[3] M. Luckie, "Scamper: A scalable and extensible packet prober for active measurement of the internet," in Proceedings of the 10th ACM SIGCOMM Conference on Internet Measurement, ser. IMC '10, New York, NY, USA: Association for Computing Machinery, 2010, 239–245, ISBN: 9781450304832. DOI: 10.1145/1879141.1879171. [Online]. Available: https://doi.org/10.1145/1879141.1879171.

[4] K. Vermeulen, S. D. Strowes, O. Fourmaux, and T. Friedman, "Multilevel mda-lite paris traceroute," in Proceedings of the Internet Measurement Conference 2018, ser. IMC '18, New York, NY, USA: Association for Computing Machinery, 2018, 29–42, ISBN: 9781450356190. DOI: 10.1145/3278532.3278536. [Online]. Available: https://doi.org/10.1145/3278532.3278536.

[5] O. Gasser, R. Holz, and G. Carle, "A deeper understanding of ssh: Results from internet-wide scans," in 2014 IEEE Network Operations and Management Symposium (NOMS), IEEE, 2014, pp. 1–9.

[6] N. Heninger, Z. Durumeric, E. Wustrow, and J. A. Halderman, "Mining your ps and qs: Detection of widespread weak keys in network devices," in Presented as part of the 21st {USENIX} Security Symposium ({USENIX} Security 12), 2012, pp. 205–220.

[7] Z. Durumeric, E. Wustrow, and J. A. Halderman, "Zmap: Fast internet-wide scanning and its security applications," in Presented as part of the 22nd {USENIX} Security Symposium ({USENIX} Security 13), 2013, pp. 605–620.

[8] D. Adrian, Z. Durumeric, G. Singh, and J. A. Halderman, "Zippier zmap: Internet-wide scanning at 10 gbps," in Proceedings of the 8th USENIX Conference on Offensive Technologies, ser. WOOT'14, San Diego, CA: USENIX Association, 2014, pp. 8–8. [Online]. Available: http://dl.acm.org/citation.cfm?id=2671293.2671301.

[9] M. Fayed, L. Bauer, V. Giotsas, et al., "The Ties That Un-Bind: Decoupling IP from Web Services and Sockets for Robust Addressing Agility at CDN-Scale," in Proceedings of the 2021 ACM SIGCOMM 2021 Conference, ser. SIGCOMM '21, New York, NY, USA: Association for Computing Machinery, 2021.

[10] J. Zirngibl, L. Steger, P. Sattler, O. Gasser, and G. Carle, "Rusty Clusters? Dusting an IPv6 Research Foundation,", Nice, France, 2022.

[11] D. Plonka and A. Berger, "Temporal and spatial classification of active ipv6 addresses," in Proceedings of the 2015 Internet Measurement Conference, ser. IMC '15, Tokyo, Japan: Association for Computing Machinery, 2015, 509–522, ISBN: 9781450338486.

[12] E. C. Rye and R. Beverly, "Discovering the ipv6 network periphery," in Passive and Active Measurement, A. Sperotto, A. Dainotti, and B. Stiller, Eds., Cham: Springer International Publishing, 2020, pp. 3–18.

[13] E. Rye, R. Beverly, and K. C. Claffy, "Follow the scent: Defeating ipv6 prefix rotation privacy," in Proceedings of the 21st ACM Internet Measurement Conference, ser. IMC '21, Virtual Event: Association for Computing Machinery, 2021, 739–752, ISBN: 9781450391290.

[14] R. Holz, L. Braun, N. Kammenhuber, and G. Carle, "The ssl landscape: A thorough analysis of the x.509 pki using active and passive measurements," in Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference, ser. IMC '11, New York, NY, USA: Association for Computing Machinery, 2011, 427–444, ISBN: 9781450310130. [Online]. Available: https://doi.org/10.1145/2068816.2068856.

Internet-wide Measurements

[15] P. Kotzias, A. Razaghpanah, J. Amann, K. G. Paterson, N. Vallina-Rodriguez, and J. Caballero, "Coming of age: A longitudinal study of tls deployment," in Proceedings of the Internet Measurement Conference 2018, ser. IMC '18, New York, NY, USA: Association for Computing Machinery, 2018, 415–428, ISBN: 9781450356190. [Online]. Available: https://doi.org/10.1145/3278532.3278568.

[16] J. Rüth, I. Poese, C. Dietzel, and O. Hohlfeld, "A First Look at QUIC in the Wild," in Passive and Active Measurement, Springer International Publishing, 2018, pp. 255–268, ISBN: 978-3-319-76481-8.

[17] J. Zirngibl, P. Buschmann, P. Sattler, B. Jaeger, J. Aulbach, and G. Carle, "It's over 9000: Analyzing early quic deployments with the standardization on the horizon," in Proceedings of the 2021 Internet Measurement Conference, Virtual Event, USA: ACM, Nov. 2021. DOI: 10.1145/3487552.3487826.

[18] B. M. Schwartz, M. Bishop, and E. Nygren, "Service binding and parameter specification via the DNS (DNS SVCB and HTTPS RRs)," Internet Engineering Task Force, Internet-Draft draft-ietf-dnsop-svcb-https-08, Oct. 2021, Work in Progress, 60 pp. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-ietf-dnsop-svcb-https-08.

[19] A. Feldmann, O. Gasser, F. Lichtblau, et al., "The lockdown effect: Implications of the covid-19 pandemic on internet traffic," in Proceedings of the ACM Internet Measurement Conference, ser. IMC '20, New York, NY, USA: Association for Computing Machinery, 2020, 1–18.

[20] A. Lutu, D. Perino, M. Bagnulo, E. Frias-Martinez, and J. Khangosstar, "A characterization of the covid-19 pandemic impact on a mobile network operator traffic," in Proceedings of the ACM Internet Measurement Conference, ser. IMC '20, New York, NY, USA: Association for Computing Machinery, 2020, 19–33, ISBN: 9781450381383.

[21] R. Fontugne, A. Shah, and K. Cho, "Persistent last-mile congestion: Not so uncommon," in Proceedings of the ACM Internet Measurement Conference, ser. IMC '20, New York, NY, USA: Association for Computing Machinery, 2020, 420–427, ISBN: 9781450381383.

[22] T. Böttger, G. Ibrahim, and B. Vallis, "How the internet reacted to covid-19: A perspective from facebook's edge network," in Proceedings of the ACM Internet Measurement Conference, ser. IMC '20, New York, NY, USA: Association for Computing Machinery, 2020, 34–41, ISBN: 9781450381383.

[23] C. Labovitz, S. Iekel-Johnson, D. McPherson, J. Oberheide, and F. Jahanian, "Internet inter-domain traffic," in Proceedings of the ACM SIGCOMM 2010 Conference, ser. SIGCOMM '10, New York, NY, USA: Association for Computing Machinery, 2010, 75–86, ISBN: 9781450302012.