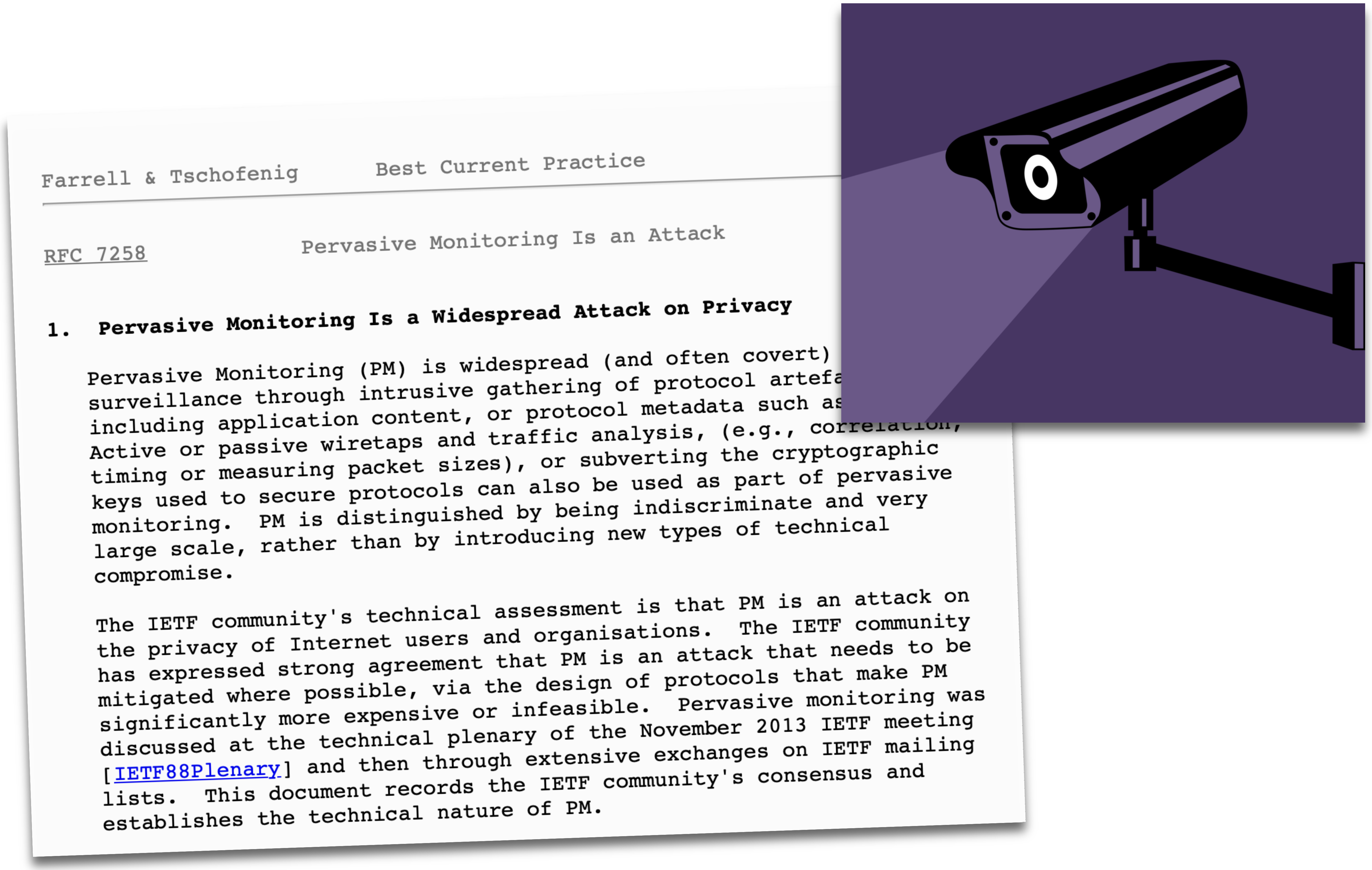
Watching the Watchers *with IPv6*: Nonce-based Inverse Surveillance to Remotely Detect Monitoring

Laura M. Roberts
Princeton University /
Akamai Technologies

David Plonka
Akamai Technologies

NPS/CAIDA 2020 Virtual IPv6 Workshop
June 17, 2020



Farrell & Tschofenig   Best Current Practice

RFC 7258   Pervasive Monitoring Is an Attack

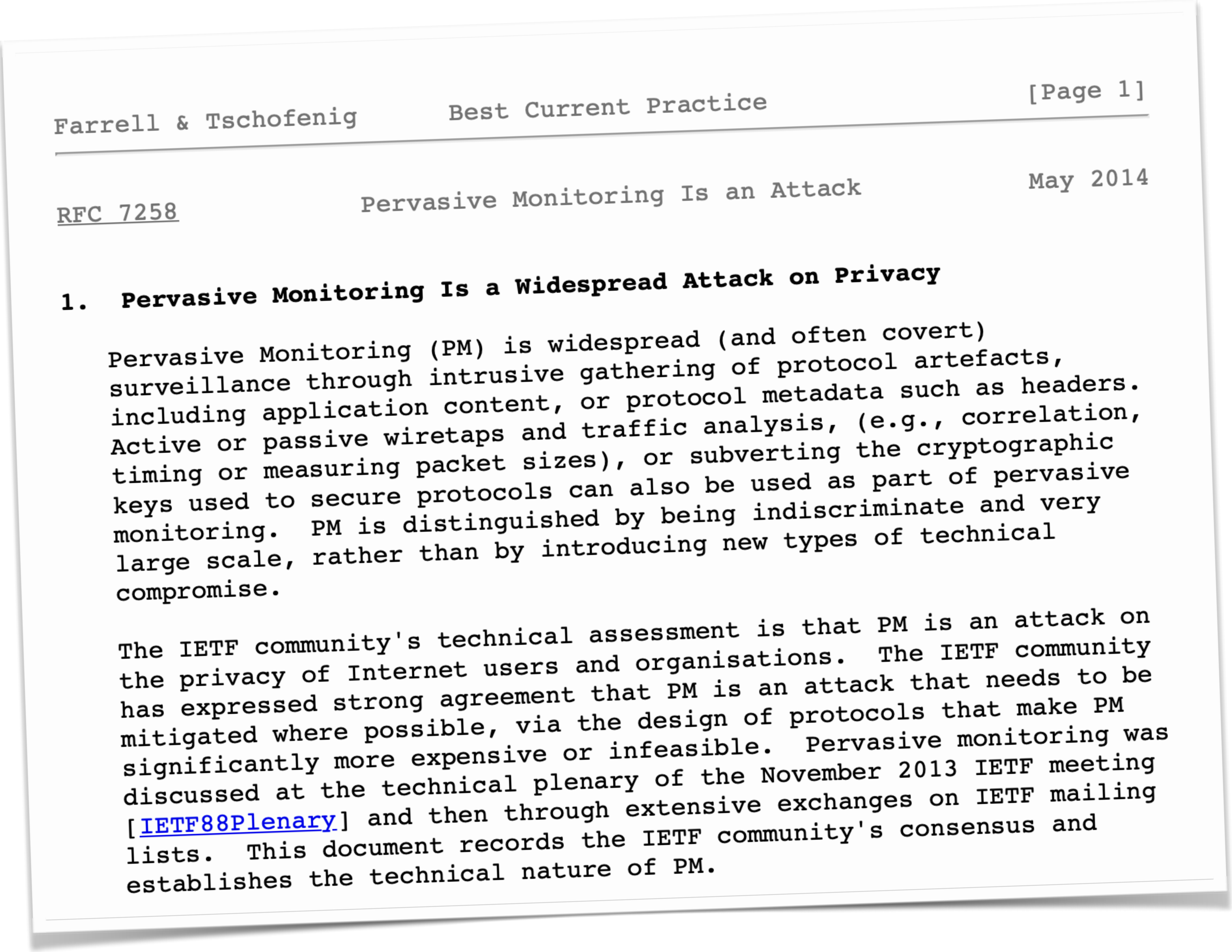
1. Pervasive Monitoring Is a Widespread Attack on Privacy

Pervasive Monitoring (PM) is widespread (and often covert)
surveillance through intrusive gathering of protocol artefa
including application content, or protocol metadata such as
Active or passive wiretaps and traffic analysis, (e.g., correlation,
timing or measuring packet sizes), or subverting the cryptographic
keys used to secure protocols can also be used as part of pervasive
monitoring. PM is distinguished by being indiscriminate and very
large scale, rather than by introducing new types of technical
compromise.

The IETF community's technical assessment is that PM is an attack on
the privacy of Internet users and organisations. The IETF community
has expressed strong agreement that PM is an attack that needs to be
mitigated where possible, via the design of protocols that make PM
significantly more expensive or infeasible. Pervasive monitoring was
discussed at the technical plenary of the November 2013 IETF meeting
[IETF88Plenary] and then through extensive exchanges on IETF mailing
lists. This document records the IETF community's consensus and
establishes the technical nature of PM.

Presented at TMA 2020: https://tma.ifip.org/2020/main-conference/
Open-access preprint: https://arxiv.org/abs/2005.07641

In today’s Internet, pervasive monitoring is deemed a threat.



Farrell & Tschofenig Best Current Practice [Page 1]

RFC 7258 Pervasive Monitoring Is an Attack May 2014

1. **Pervasive Monitoring Is a Widespread Attack on Privacy**

Pervasive Monitoring (PM) is widespread (and often covert) surveillance through intrusive gathering of protocol artefacts, including application content, or protocol metadata such as headers. Active or passive wiretaps and traffic analysis, (e.g., correlation, timing or measuring packet sizes), or subverting the cryptographic keys used to secure protocols can also be used as part of pervasive monitoring. PM is distinguished by being indiscriminate and very large scale, rather than by introducing new types of technical compromise.

The IETF community's technical assessment is that PM is an attack on the privacy of Internet users and organisations. The IETF community has expressed strong agreement that PM is an attack that needs to be mitigated where possible, via the design of protocols that make PM significantly more expensive or infeasible. Pervasive monitoring was discussed at the technical plenary of the November 2013 IETF meeting [IETF88Plenary] and then through extensive exchanges on IETF mailing lists. This document records the IETF community's consensus and establishes the technical nature of PM.

Internet users and service providers don’t know who’s watching their Internet traffic.

We desire a way to detect *who* is monitoring Internet traffic and *where* it's being monitored.

- Want to detect organizations who monitor traffic and systems that monitor traffic, such as network firewalls or email filters

- Want to know where they are, be it along network links or at edges

Research question:
Can we build a system that remotely detects monitoring?

We propose the use of nonces to accomplish this.

- Nonces are single-use, pseudorandom values
- First, we *actively disseminate* nonces, i.e., we transmit them as a packet’s IPv6 source address in an active measurement survey
- Then we *passively listen* for a surveillant to *propagate/react* to the nonce, e.g., to use it in a reverse DNS query
- Because nonces are unique, we can correlate the *dissemination* with subsequent *propagations/reactions*
- We’re also able to glean topological information on paths that nonces traverse, which helps locate where the surveillants might be

We present NOISE, the Nonce Observatory for Inverse Surveillance of Eavesdroppers.

- A novel way to detect monitors of Internet traffic remotely

Agenda

- Describe the system

- Present our results

Let’s describe the system.

We disseminate nonces and listen for reactions.

- There is an **active** component to our system and a **passive** component
- We need a way to **actively** spread nonces (*dissemination*) in Internet traffic and to **passively** detect reactions to these nonces (*propagation*)
- There are various strategies we could use to realize both components
- We used a worldwide, IPv6 traceroute-like measurement campaign to do just that and detect surveillants

Our Strategy - The Nonces

- First we generate 64-bit nonces, and because of IPv6's huge address space, we embed them in (128-bit) IPv6 addresses, for example, in the lower 64 bits

- We generate nonces by encrypting 64 bits of data with the ChaCha20 stream cipher

- We do this because it's important that our nonces be unpredictable

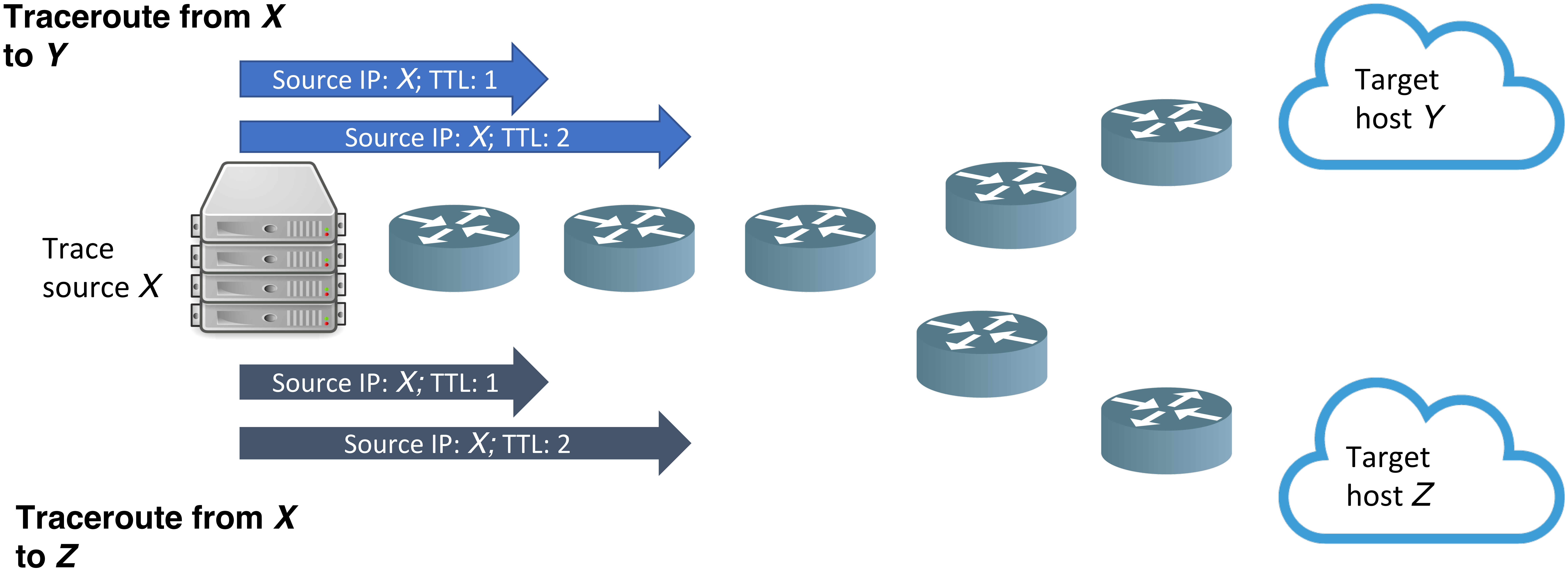
- If they were predictable, an adversary could craft and transmit valid nonces itself, instead of by merely reacting to ours, confusing our analysis

Our Strategy - The Active Component

- With our “nonced” IPv6 addresses in hand, we disseminate them by running a **special** traceroute campaign.

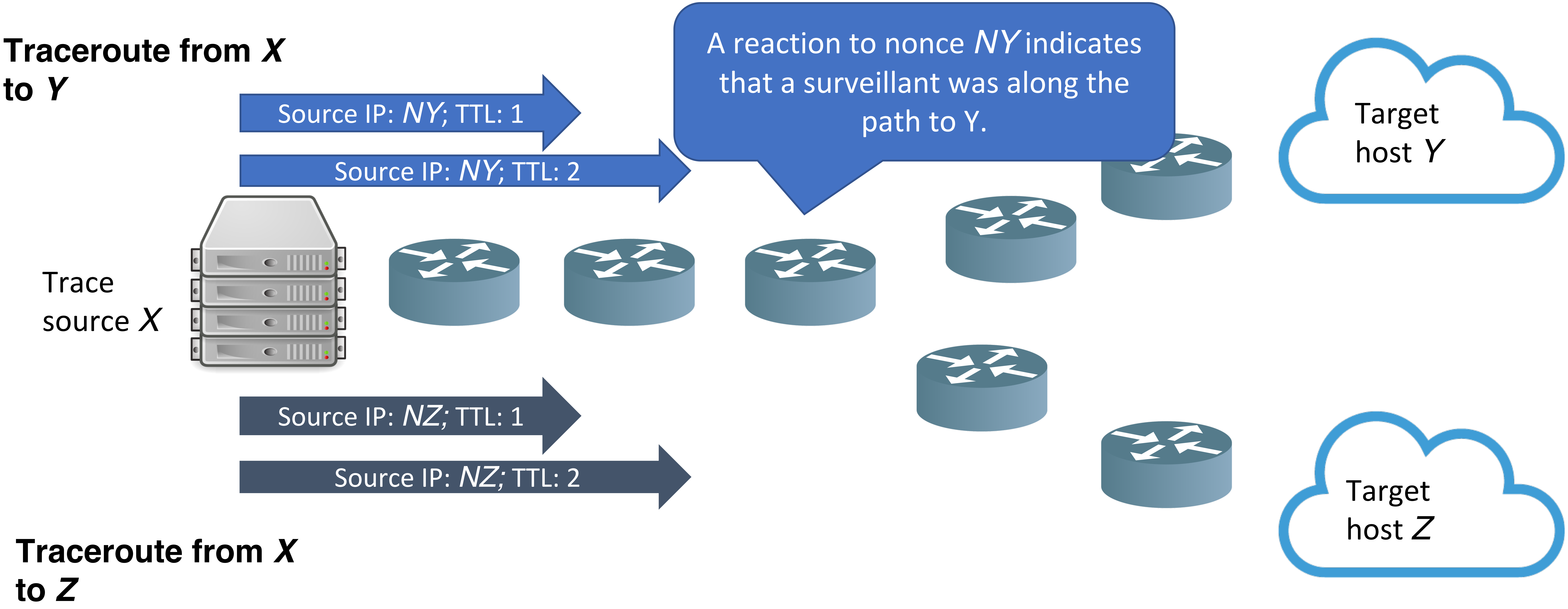
First, let’s review how regular traceroute works.

- Probes are sent from the IP address of the source host to the targets



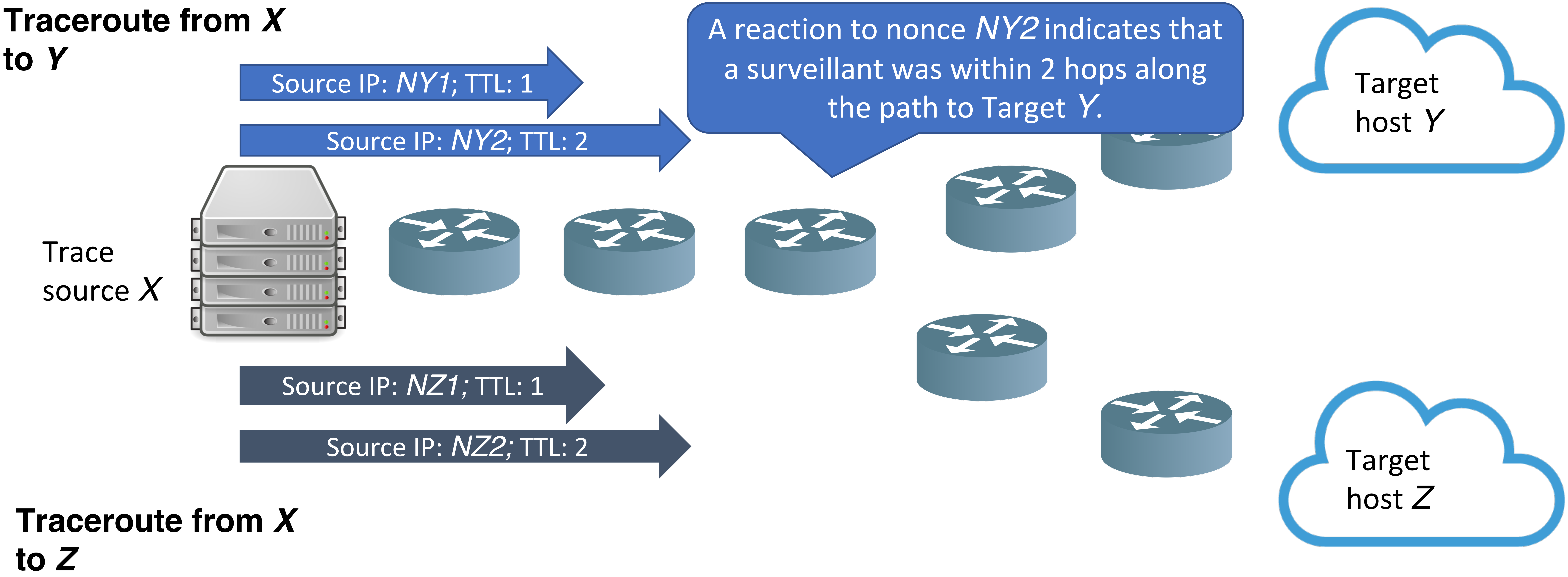
In our special traceroute campaign, we craft or *forge* one-time-use, nonce-laden source addresses.

- We emit packets with those rather than the host’s usual source address. Here we show one nonce per destination.


Traceroute from *X* to *Y*
Source IP: *NY*; TTL: 1
Source IP: *NY*; TTL: 2
A reaction to nonce *NY* indicates that a surveillant was along the path to Y.
Target host *Y*
Trace source *X*
Source IP: *NZ;* TTL: 1
Source IP: *NZ;* TTL: 2
Target host *Z*
Traceroute from *X* to *Z*

Let’s have forged source IPv6 addresses for *each TTL (hop limit)*.

- The IPv6 number space is huge so we can afford to place a unique nonce in every packet we emit; Offers us finer granularity in determining where the surveillant actually was along the path



Traceroute from *X* to *Y*

Source IP: *NY1*; TTL: 1

Source IP: *NY2*; TTL: 2

A reaction to nonce *NY2* indicates that a surveillant was within 2 hops along the path to Target *Y*.

Target host *Y*

Trace source *X*

Source IP: *NZ1;* TTL: 1

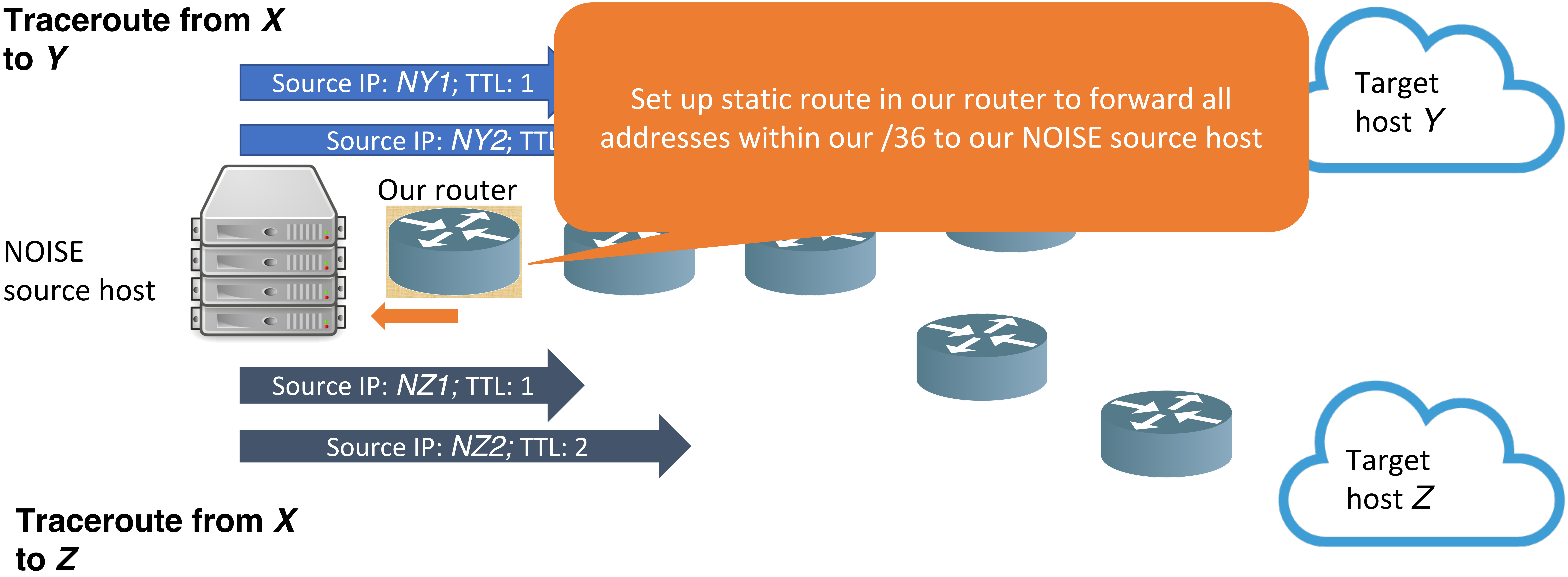
Source IP: *NZ2;* TTL: 2

Target host *Z*

Traceroute from *X* to *Z*

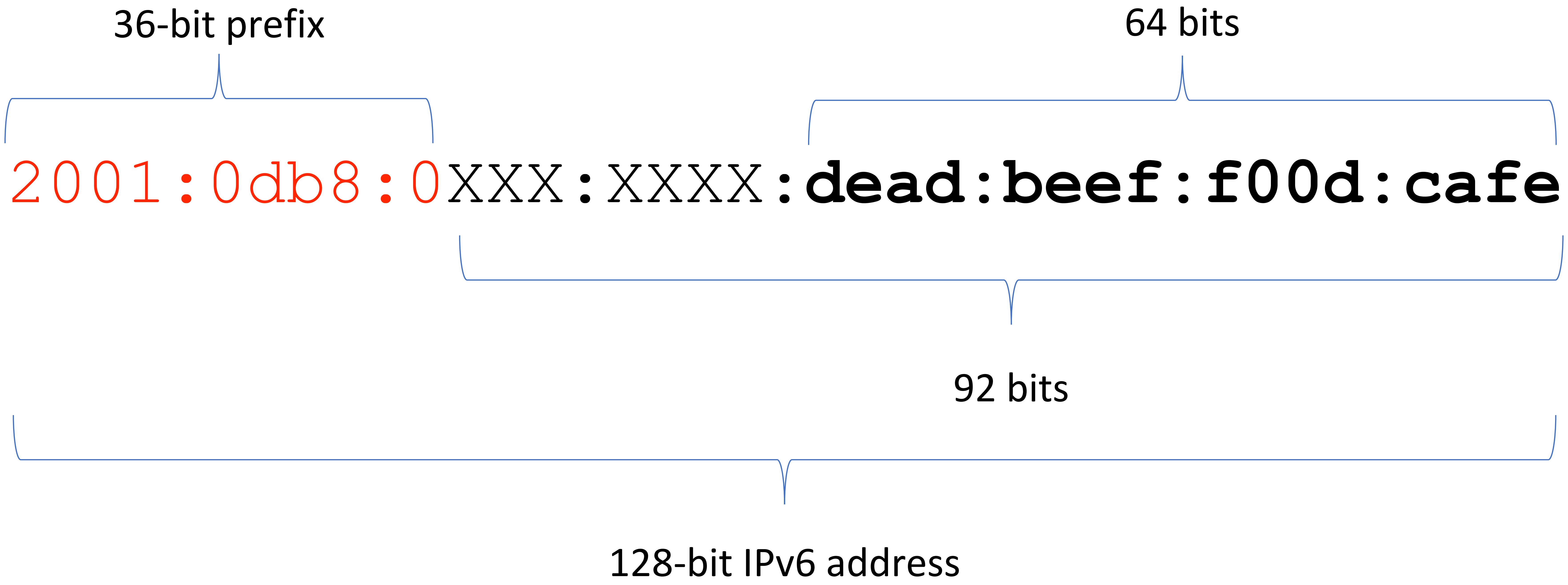
How are we able to collect responses to our traceroute probes given that the source addresses are forged?

- We limit our forged sources to an IPv6 address block (/36) completely under our control and forward all packets destined to addresses within that block to the NOISE source host


Traceroute from *X* to *Y*
Source IP: *NY1*; TTL: 1
Source IP: *NY2*; TTL
Our router
NOISE source host
Set up static route in our router to forward all addresses within our /36 to our NOISE source host
Target host *Y*
Source IP: *NZ1;* TTL: 1
Source IP: *NZ2;* TTL: 2
Target host *Z*
Traceroute from *X* to *Z*

Let’s take a closer look at the /36 IPv6 address block that’s under our control.

- The NOISE address block is an IPv6 /36 prefix that has 2^{92} possible addresses, each of which can contain any of 2^{64} possible nonces



36-bit prefix

64 bits

2001:0db8:0XXX:XXXX:dead:beef:f00d:cafe

92 bits

128-bit IPv6 address

Our Strategy - The Active Component

- In our experiments, we ran yarrp on a computer dedicated to NOISE—this is our trace source host
- We traced from nonced IPv6 source addresses to the approximately 15.2M target addresses used in prior work[1] which is to the best of our knowledge the largest IPv6 topology survey to date
- We are disseminating our nonces while getting a sense of the topology so we can know where the monitoring happened

[1] "In the IP of the Beholder: Strategies for Active IPv6 Topology Discovery" by Beverly et al. (IMC 2018) https://arxiv.org/abs/1805.11308

Our Strategy - The Passive Component

- After disseminating our nonces via this special yarrp-based traceroute survey, we then wait to see who or what reacts with interest to our nonced source addresses

- An example of “interest” could be the receipt of a packet destined for a nonce-laden address from a host that was not a target of our traceroutes, and we capture all such unsolicited packets on our machine. We call these “**pcap**” reactions.

Our Strategy - The Passive Component

- We know from experience that a common reaction to unsolicited traffic from an unfamiliar address (from our /36) is to perform a reverse DNS query on it
- We capture this traffic at our NOISE DNS server, which is NSD (open-source DNS server) running on a virtual machine (VM) that was made to be the authoritative reverse DNS nameserver for NOISE’S /36 IPv6 address block
- This way, we’re able to capture DNS queries involving any of our nonced source addresses ourselves
- We refer to these as “**rdns**” reactions

Our Strategy - The Passive Component

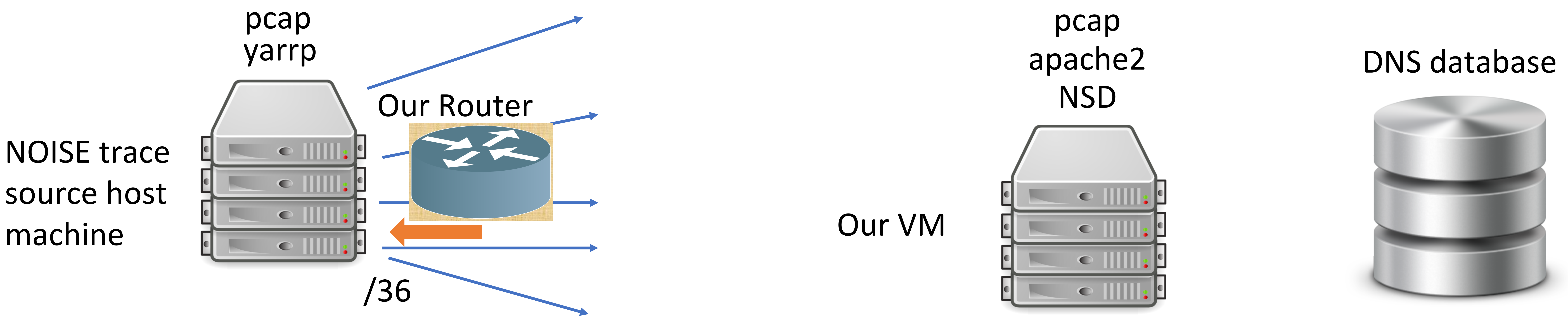
- Our nameserver is also authoritative for forward queries in two NOISE project domains, which enables us to capture “**fdns**” reactions

- And we have access to DNSDB, a passive DNS database, which allows us to determine when queries for our nonced addresses or project domains were shared with this third-party commercial database, and we refer to these as “**pdns**” reactions

We employ all of these components in our NOISE experiments to evaluate its performance in detecting monitoring.

2001:0db8:0XXX:XXXX:**dead:beef:f00d:cafe**

something1.noise.example.com
something2.noise.example.com



pcap
yarrp

NOISE trace source host machine

Our Router

/36

pcap
apache2
NSD

Our VM

DNS database

Let’s discuss our results.

Our results come from three experiments.

Exp. Name	Description	Maximum TTL	Dates, 2019	Traces Performed
UDP:443c	**UDP probes sent TO port 443**	**32**	Jan 4 –10	15.2M
UDP:443s	**UDP probes sent FROM port 443**	**24**	Apr 10 –14	15.2M
Ping	**ICMPv6 Echo Request probes**	**16 +**	Apr 15 –18	15.2M

Macroscopic View

- Across three experiments, NOISE detected monitoring more than 200k times, ostensibly in 268 networks, for probes destined for 437 networks.

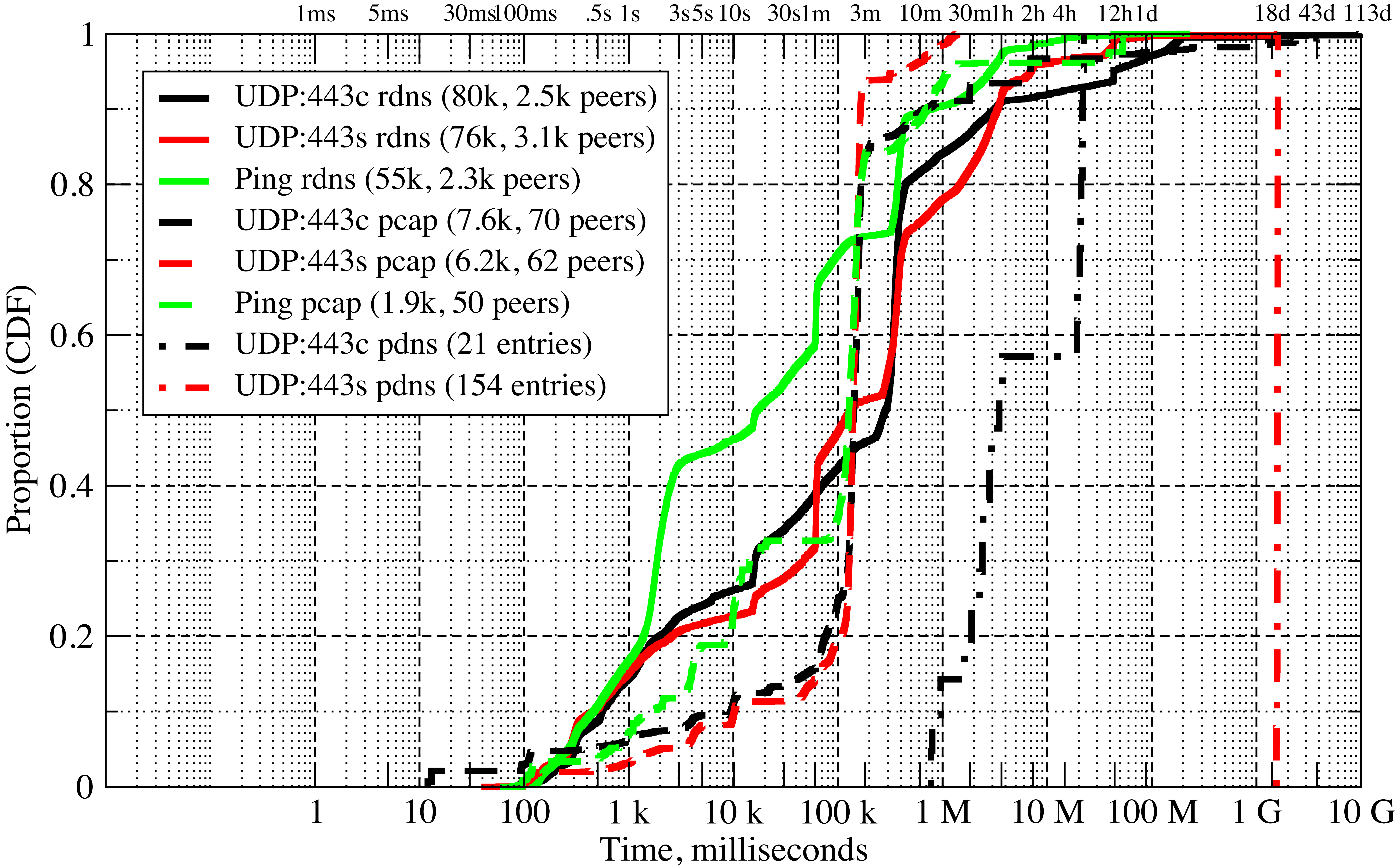
- We are particularly interested in the following types of evidence of monitoring:

  - **rdns**: reverse lookups

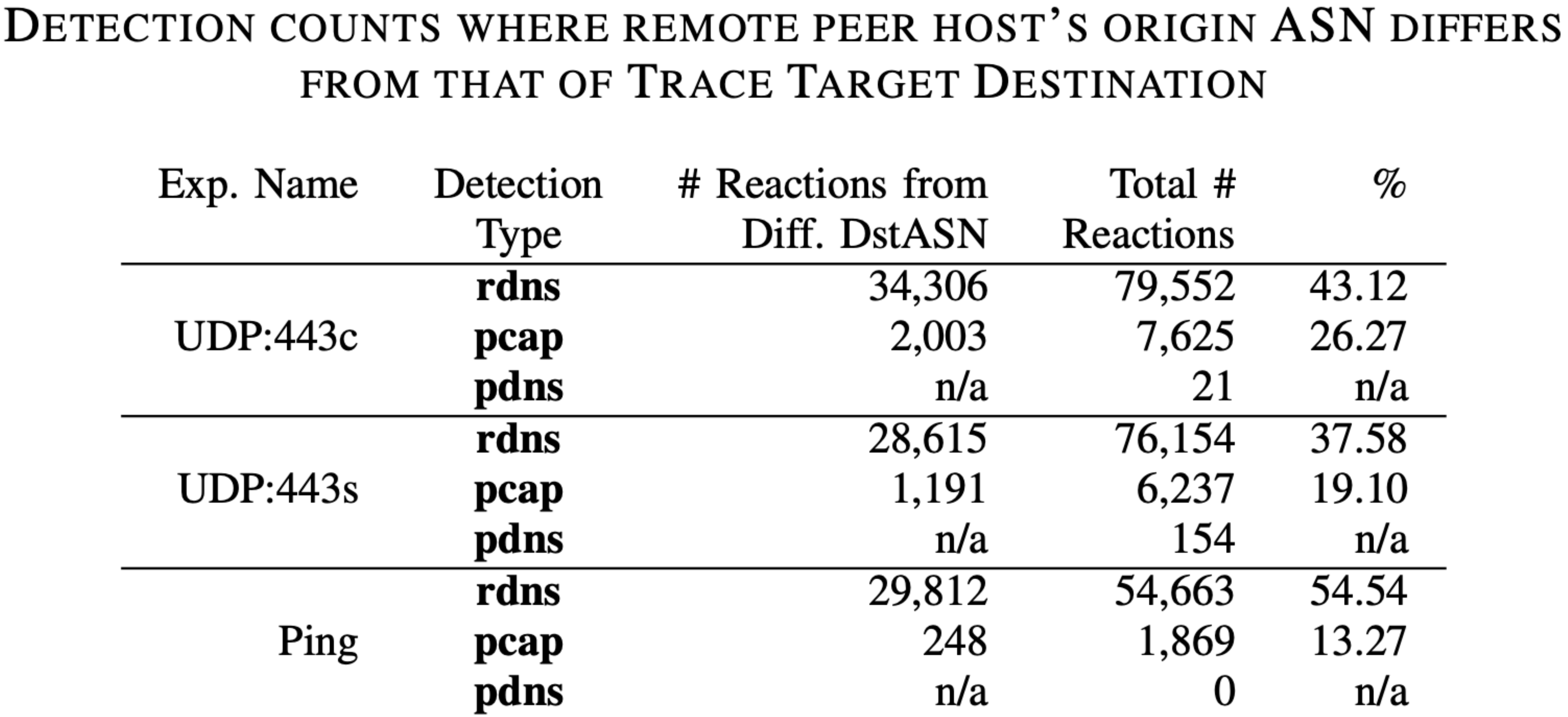
  - **pcap**: unexpected packets that talk back to our nonced source addresses

  - **pdns**: entries in DNSDB, a commercial passive DNS database

Macroscopic View: times to detection of nonce propagation



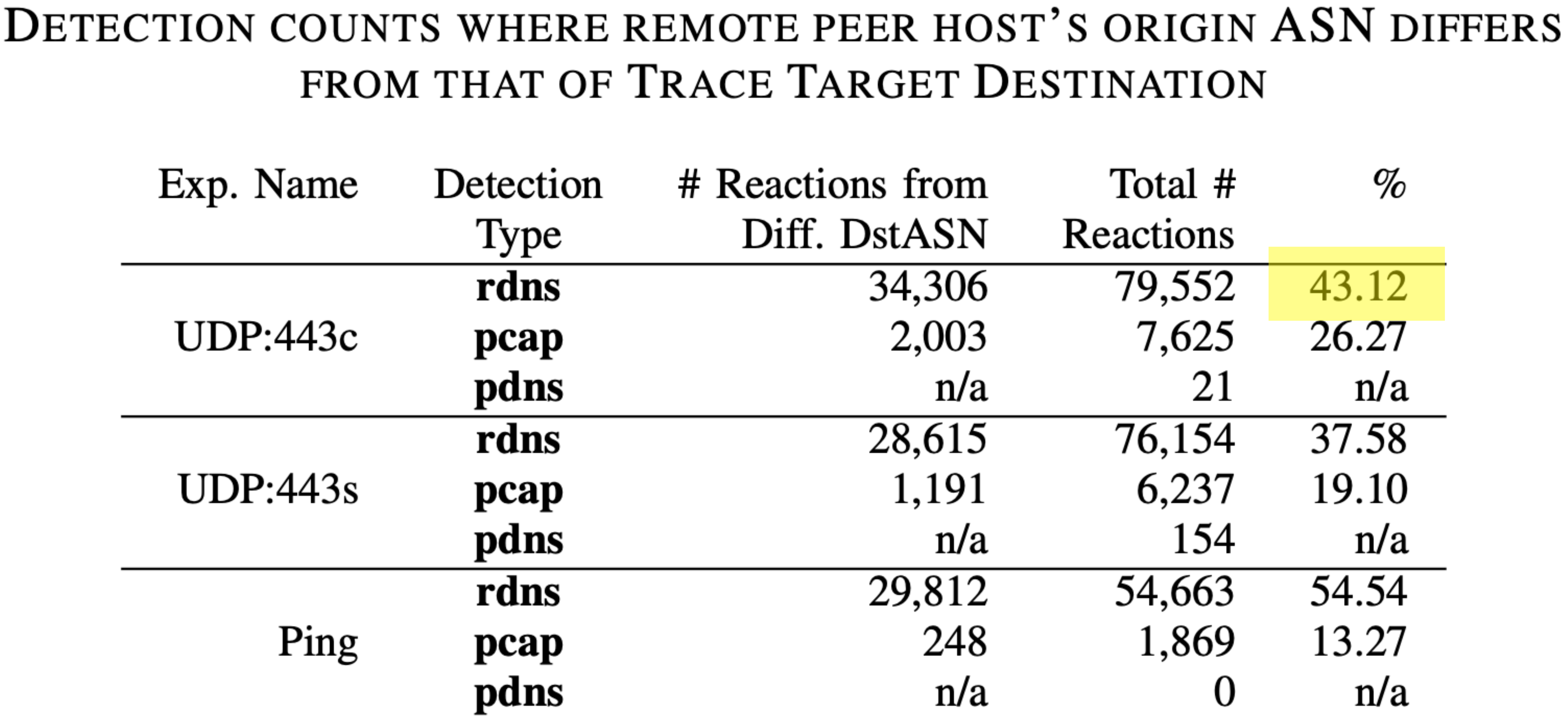
Macroscopic View



Detection counts where remote peer host's origin ASN differs from that of Trace Target Destination

Exp. Name	Detection Type	# Reactions from Diff. DstASN	Total # Reactions	%
UDP:443c	**rdns**	34,306	79,552	43.12
	pcap	2,003	7,625	26.27
	pdns	n/a	21	n/a
UDP:443s	**rdns**	28,615	76,154	37.58
	pcap	1,191	6,237	19.10
	pdns	n/a	154	n/a
Ping	**rdns**	29,812	54,663	54.54
	pcap	248	1,869	13.27
	pdns	n/a	0	n/a

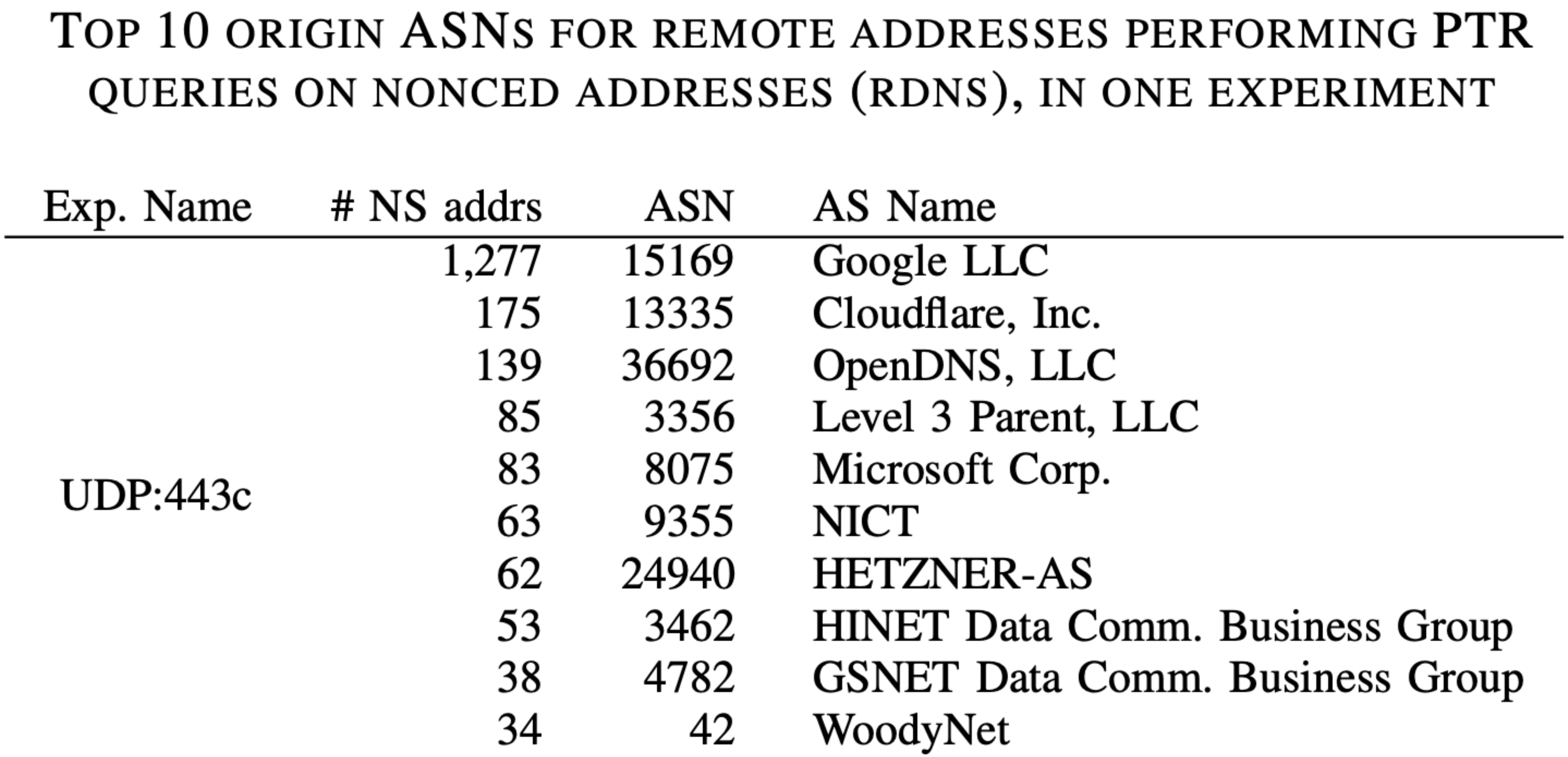
Macroscopic View



Detection counts where remote peer host's origin ASN differs from that of Trace Target Destination

Exp. Name	Detection Type	# Reactions from Diff. DstASN	Total # Reactions	%
UDP:443c	**rdns**	34,306	79,552	43.12
	pcap	2,003	7,625	26.27
	pdns	n/a	21	n/a
UDP:443s	**rdns**	28,615	76,154	37.58
	pcap	1,191	6,237	19.10
	pdns	n/a	154	n/a
Ping	**rdns**	29,812	54,663	54.54
	pcap	248	1,869	13.27
	pdns	n/a	0	n/a

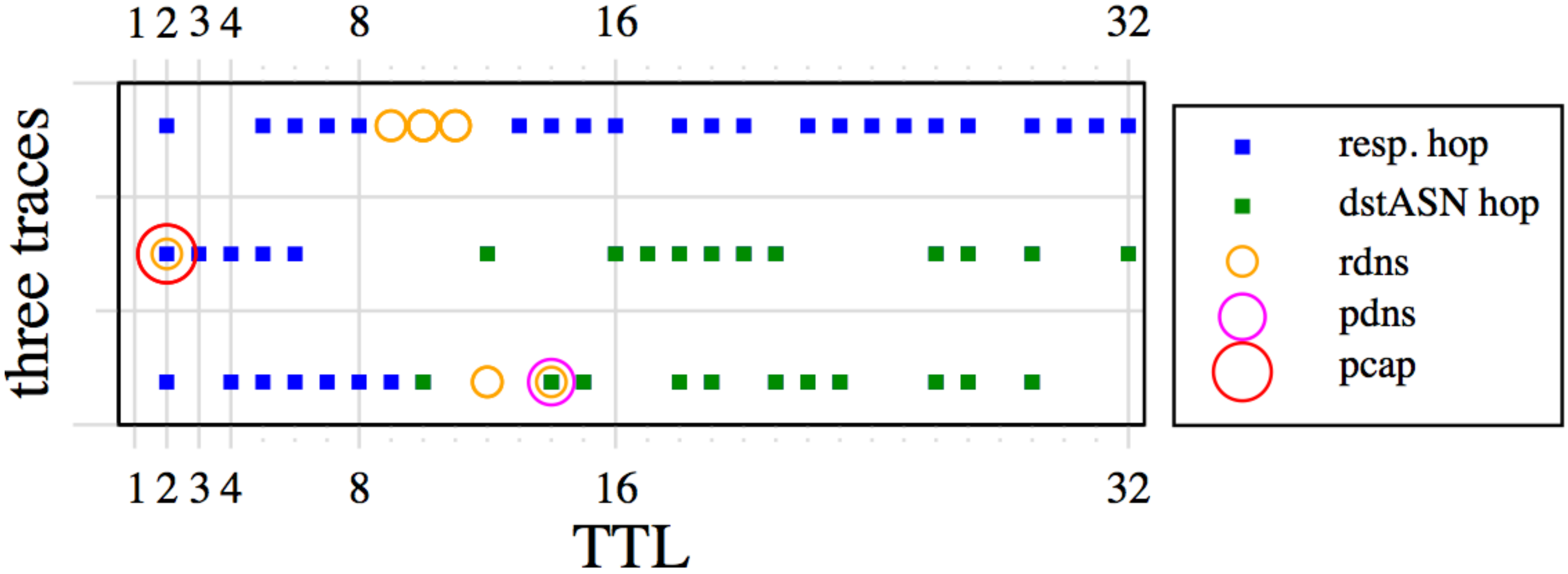
Macroscopic View



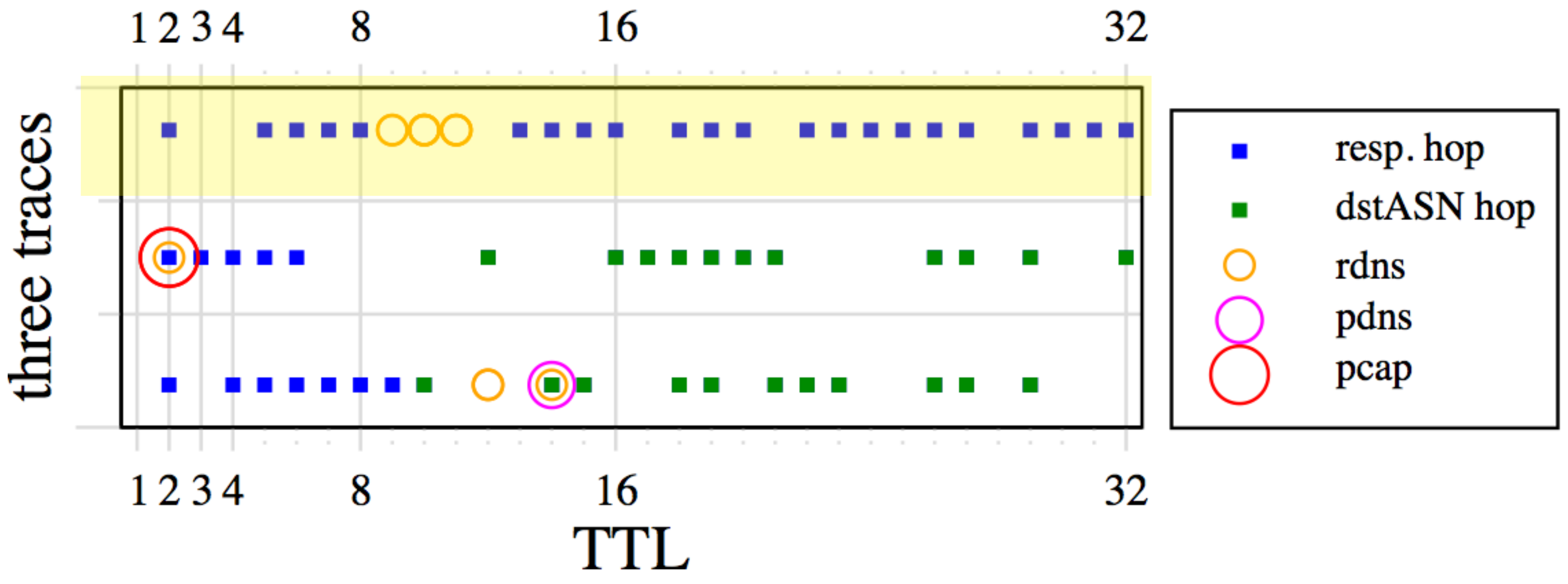
Top 10 origin ASNs for remote addresses performing PTR queries on nonced addresses (rDNS), in one experiment

Exp. Name	# NS addrs	ASN	AS Name
UDP:443c	1,277	15169	Google LLC
	175	13335	Cloudflare, Inc.
	139	36692	OpenDNS, LLC
	85	3356	Level 3 Parent, LLC
	83	8075	Microsoft Corp.
	63	9355	NICT
	62	24940	HETZNER-AS
	53	3462	HINET Data Comm. Business Group
	38	4782	GSNET Data Comm. Business Group
	34	42	WoodyNet

Microscopic View of NOISE Capabilities and Results Validation


1
2
3
4
8
16
32
three traces
TTL
resp. hop
dstASN hop
rdns
pdns
pcap

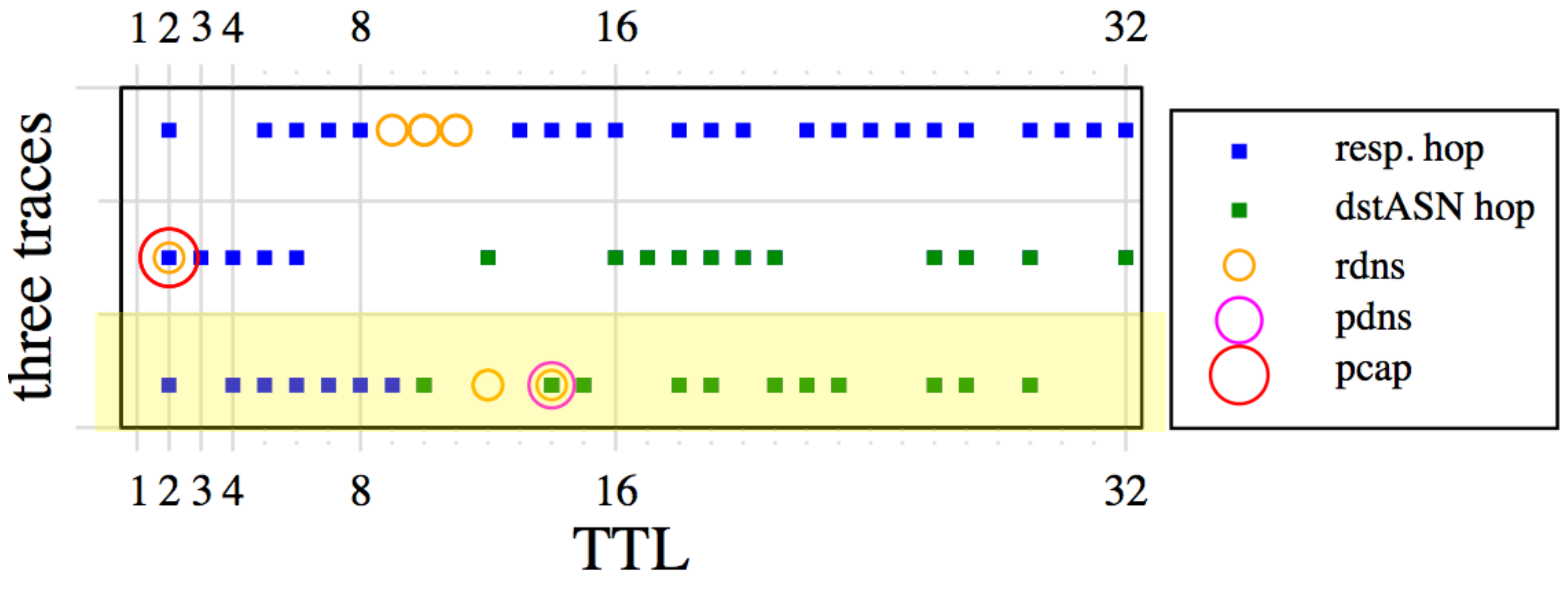
NOISE Capability 1: Detection of Curious Queries and Improved Reachability Measurements



EVENTS THAT OCCURRED DURING TRACE TO AN ASIAN NETWORK IN EXPERIMENT UDP:443C

Delta time	Event	ProbeTTL
0s	tr probe sent to target	26
0.24s	tr hop response	26
8m 56s	tr probe sent to target	10
9m 7s	**RDNS query on *noncedAddr* by target's network**	**10**
9m 10s	**RDNS query on *noncedAddr* by target's network**	**10**
3h 6m	tr probe sent to target	14
3h 6m	tr hop response	14
3h 38m	tr probe sent to target	32
3h 38m	tr hop response	32
⋮	⋮	⋮
1d 15h	last tr probe sent to target	29
1d 15h	tr hop response	29

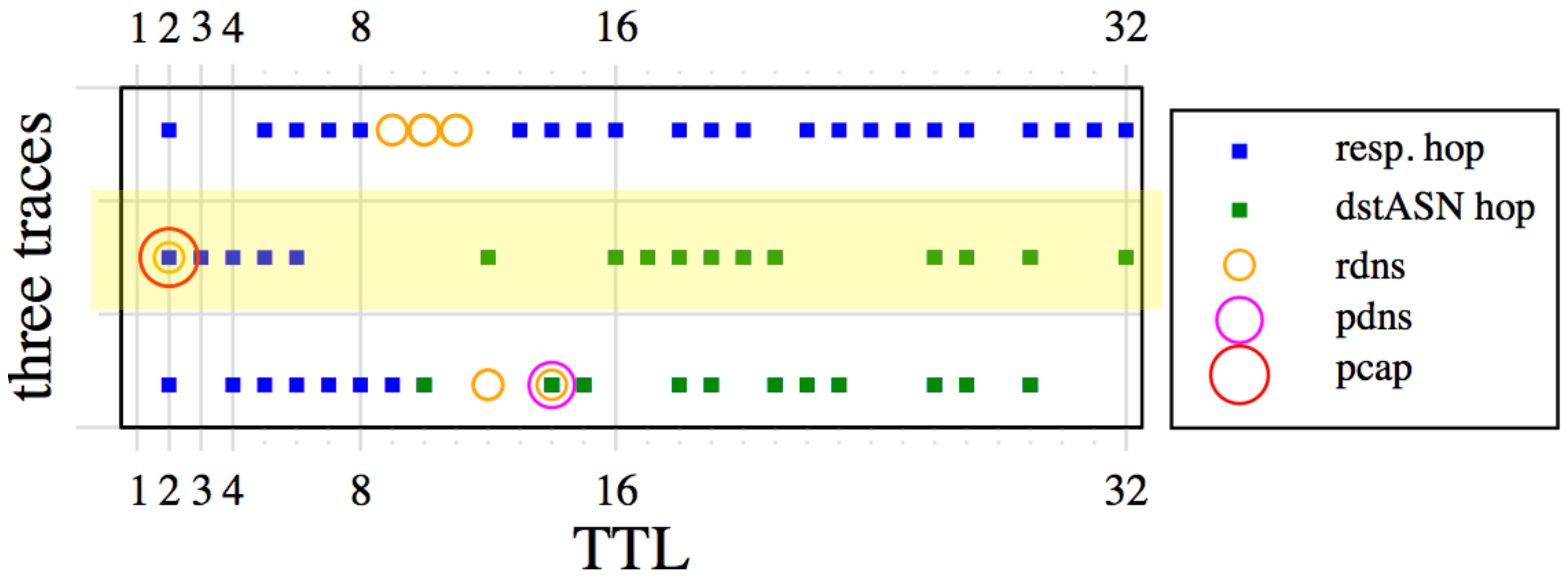
NOISE Capability 2: Detection of Sharing Passive DNS Data



Events that occurred during and after trace to a university in Experiment UDP:443s

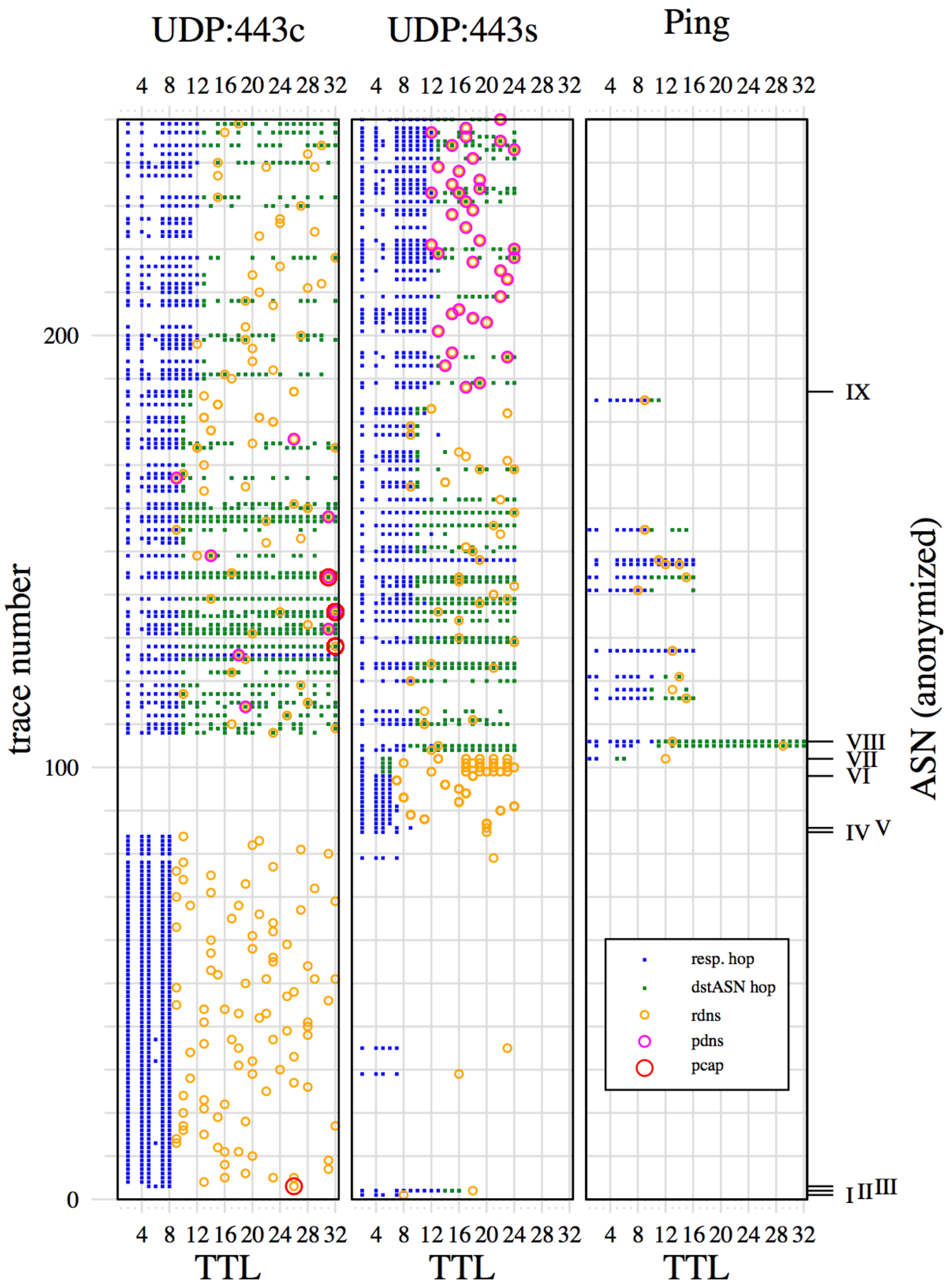
Delta time	Event	ProbeTTL
0s	tr probe sent to target	15
2m 32s	tr probe sent to target	17
16m 14s	tr probe sent to target	7
16m 14s	tr hop response	7
⋮	⋮	⋮
1h 47s	tr probe sent to target	14
⋮	⋮	⋮
4h 44m	last tr probe sent to target	4
4h 44m	tr hop response	4
18d 5h	**RDNS query on *noncedAddr* by university**	**14**
18d 6h	***noncedAddr* appears in passive DNS database**	**14**

NOISE Capability 3: Detection of Eavesdropping



EVENTS THAT OCCURRED DURING TRACE DETECTING EAVESDROPPING IN EXPERIMENT UDP:443C

Delta time	Event	ProbeTTL
0s	tr probe sent to target	2
0.0005s	tr hop response	2
9m 58s	**TCP SYN :20 → *noncedAddr*:80 by cloud Provider**	**2**
10m 25s	**TCP SYN :20 → *noncedAddr*:443 by cloud Provider**	**2**
10m 43s	**RDNS query on *noncedAddr* by cloud DNS Provider**	**2**
22m 26s	tr probe sent to target	24
⋮	⋮	⋮
11h 51m	last tr probe sent to target	15


UDP:443c
UDP:443s
Ping
4 8 12 16 20 24 28 32
trace number
0
100
200
TTL
ASN (anonymized)
I II III
IV V
VI
VII
VIII
IX
resp. hop
dstASN hop
rdns
pdns
pcap

Conclusion

- We have presented NOISE, the Nonce Observatory for Inverse Surveillance of Eavesdroppers, a novel way to detect monitors of Internet traffic remotely.
- While NOISE currently implements one mode of nonce dissemination, many others are possible, e.g., in the WWW
- And we envision a system that is so pervasive, surveillants would have no choice but to observe our nonce-laden traffic, improving detection of surveillants whenever they act on their observations

Presented at TMA 2020 (June): https://tma.ifip.org/2020/main-conference/
Open-access preprint: https://arxiv.org/abs/2005.07641

Conclusion

- We have presented NOISE, the Nonce Observatory for Inverse Surveillance of Eavesdroppers, a novel way to detect monitors of Internet traffic remotely.

- While NOISE currently implements one mode of nonce dissemination, many others are possible, e.g., in the WWW

- And we envision a system that is so pervasive, surveillants would have no choice but to observe our nonce-laden traffic, improving detection of surveillants whenever they act on their observations

Acknowledgments: Niels Bakker, Arthur Berger, Robert Beverly, Aaron Block, David Choffnes, David Duff, Jared Mauch, Suzanne Pan, Philipp Richter, Kyle Rose, Steven Schecter, Chris Schill, Jon Thompson, and Rick Weber

Presented at TMA 2020 (June): https://tma.ifip.org/2020/main-conference/
Open-access preprint: https://arxiv.org/abs/2005.07641