On the use of TCP's Initial Congestion Window in IPv4 and by Content Delivery Networks

Jan Rüth, Christian Bormann, Oliver Hohlfeld

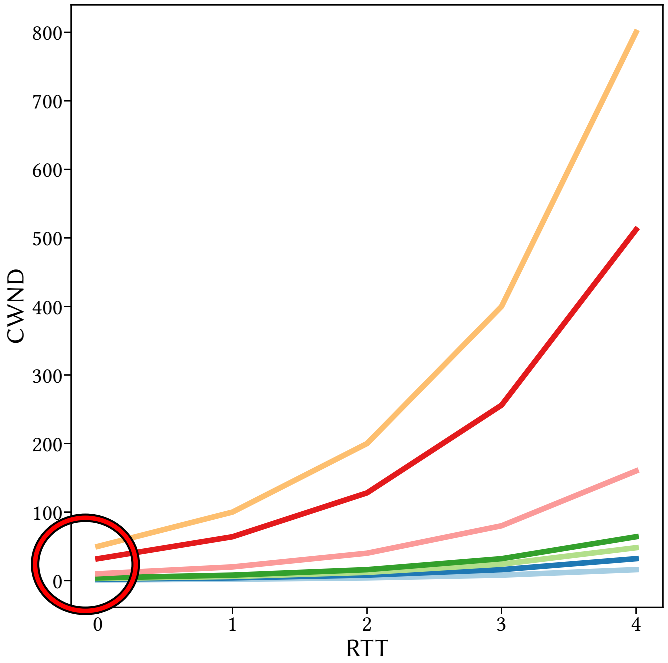
http://comsys.rwth-aachen.de/

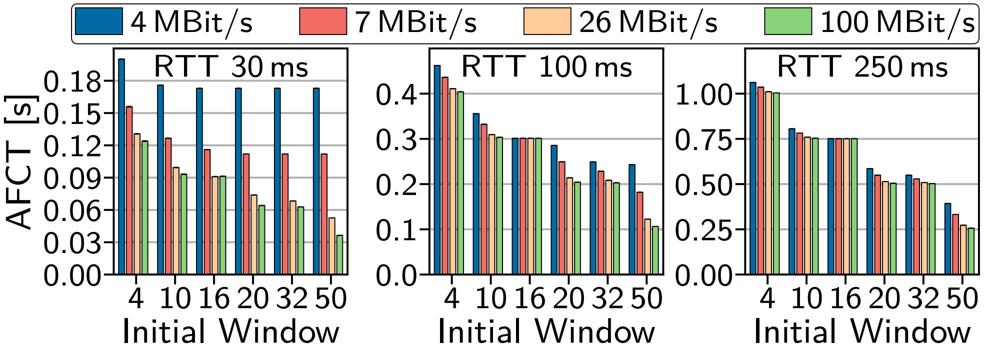
London / IETF-101, March 2018

COM SYS | RWTH AACHEN UNIVERSITY

Why look at Initial Windows?

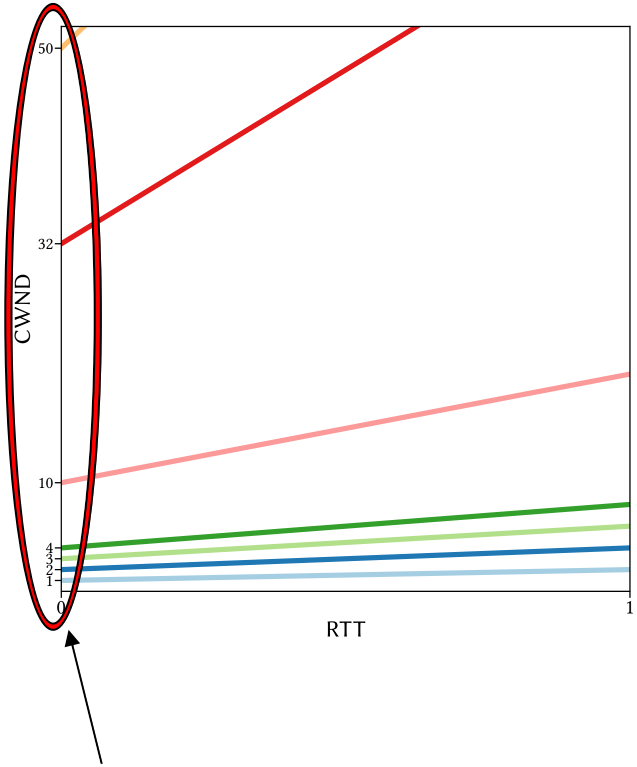
- **Initial Window = bootstrap value for cwnd in slowstart**
  - Number of unacknowledged bytes in the first round trip
  - Typically a multiple of the MSS



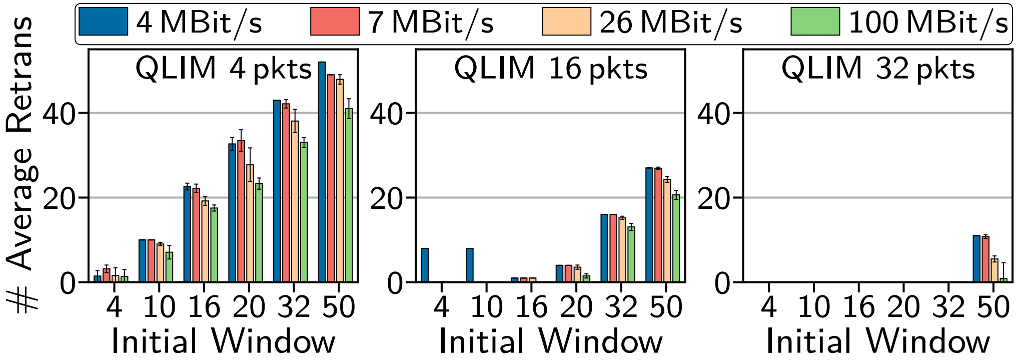


Why look at Initial Windows?

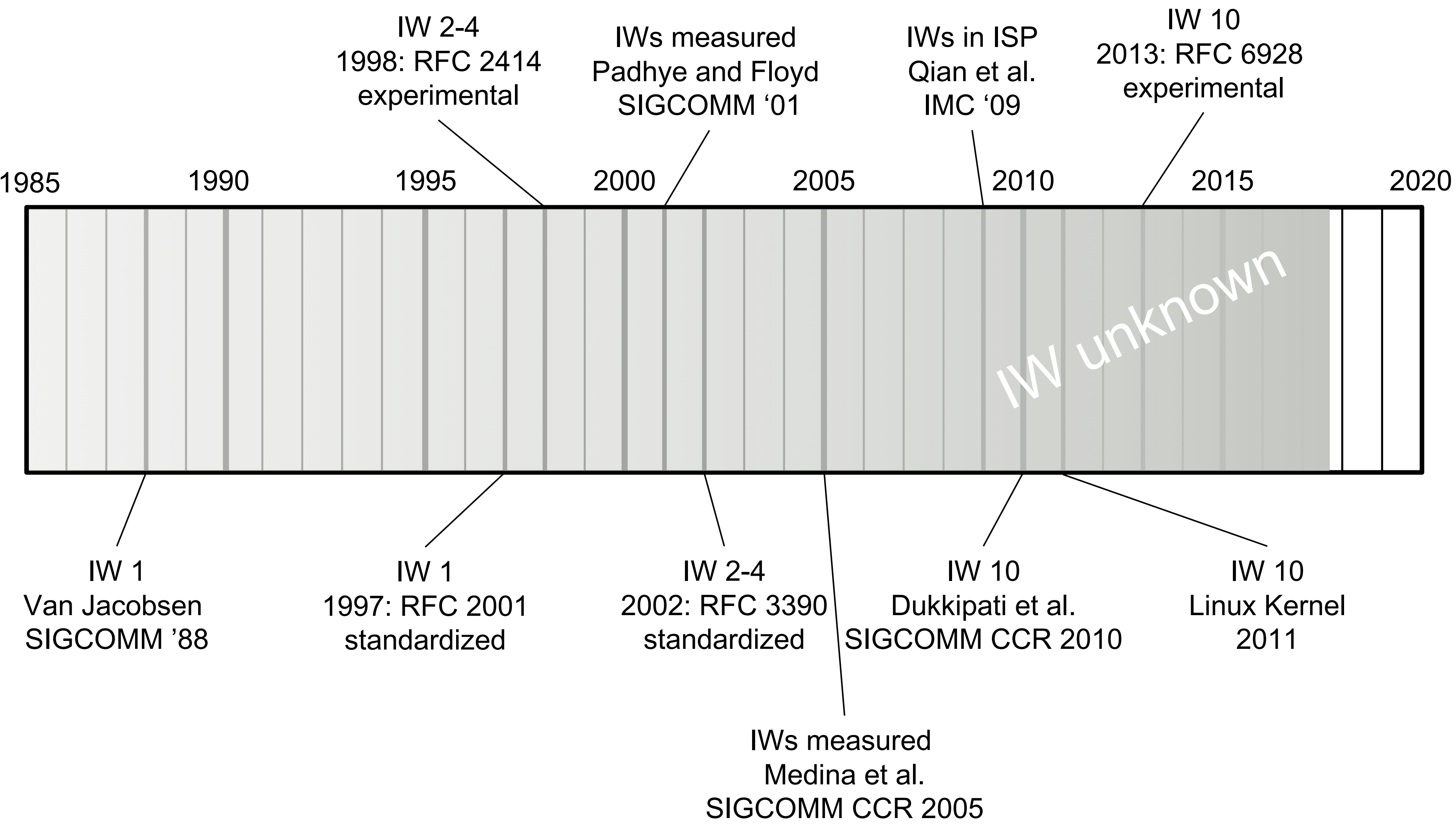
- **TCP bursts the IW in an unprobed network**
  - Can lead to loss at the bottleneck → bad



At the start,
we don't know
the bottleneck capacity



How large is it?



IW 1
Van Jacobsen
SIGCOMM '88

IW 1
1997: RFC 2001
standardized

IW 2-4
1998: RFC 2414
experimental

IWs measured
Padhye and Floyd
SIGCOMM '01

IW 2-4
2002: RFC 3390
standardized

IWs measured
Medina et al.
SIGCOMM CCR 2005

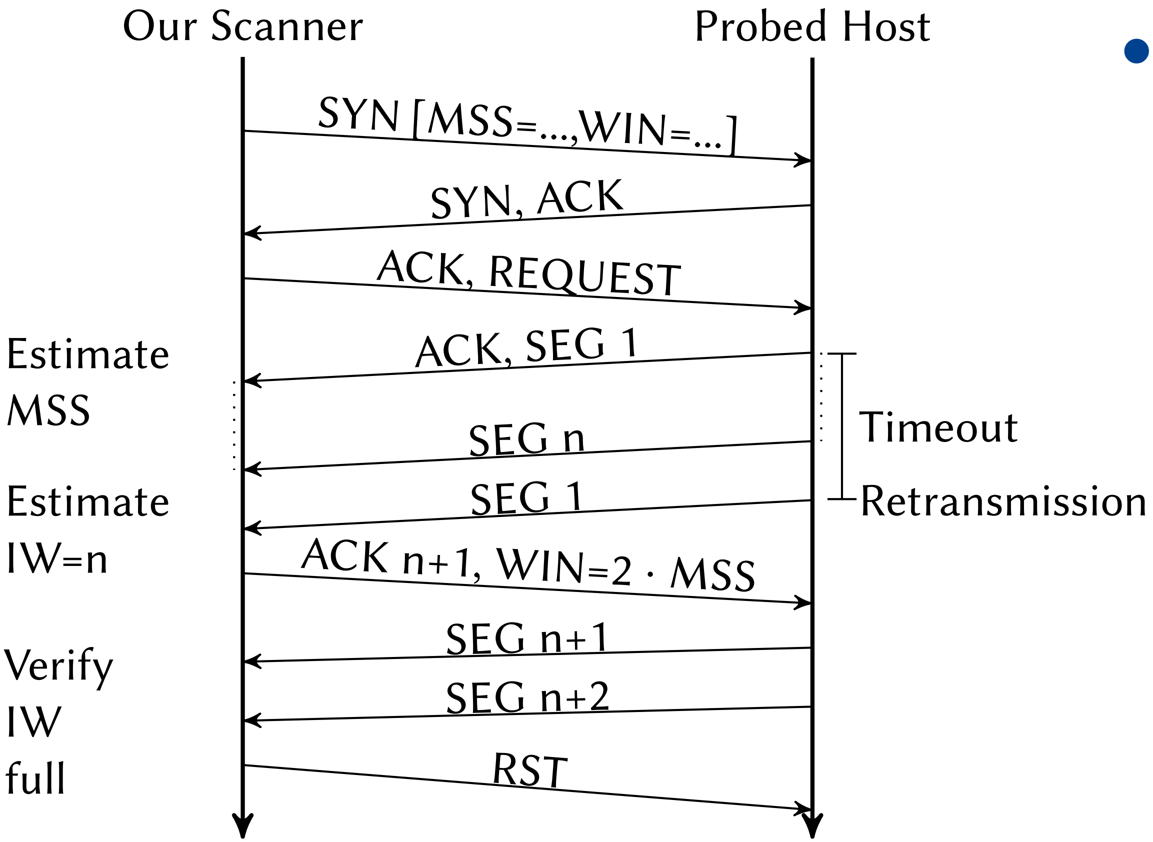
IWs in ISP
Qian et al.
IMC '09

IW 10
Dukkipati et al.
SIGCOMM CCR 2010

IW 10
Linux Kernel
2011

IW 10
2013: RFC 6928
experimental

Measuring IWs



- **Loss is a problem**
  - Actually tail-loss
  - **Do multiple scans**
  - Disable tail-loss probes
    - Do not enable SACK

- **Announce small MSS and large receive window**
- **Use ACK to test for more data**
  - Was the host out of data or was the IW actually full?

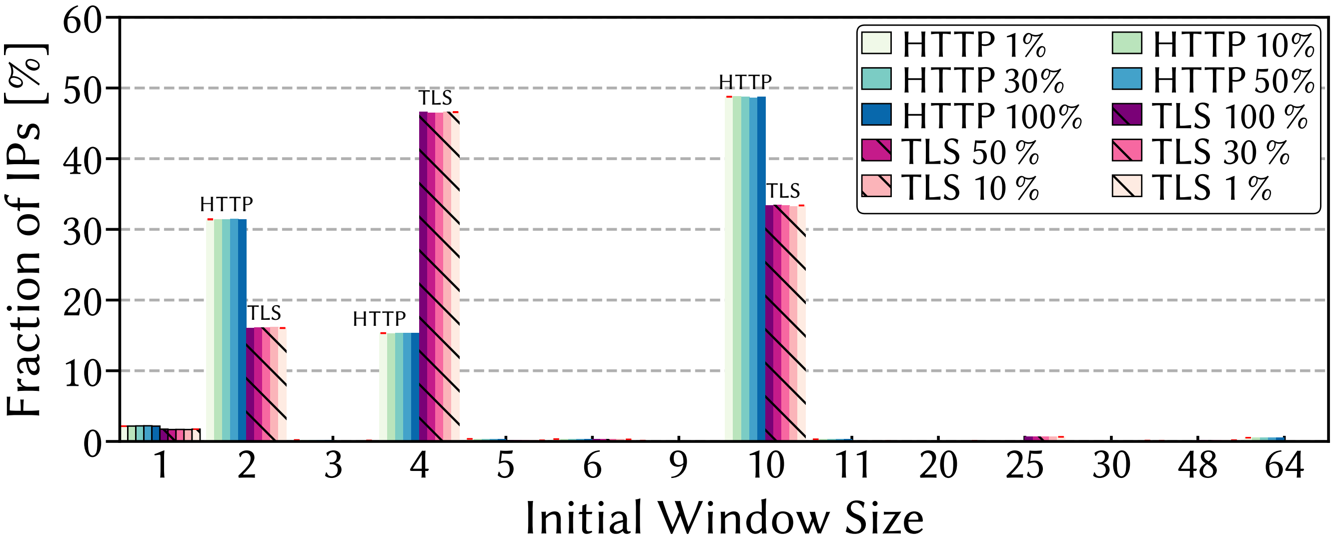
Scanner implementation

- **We want to probe all reachable IPv4 HTTP/TLS hosts**

- **We implement the methodology in ZMap**
  - Bypasses the kernel stack
  - Typically only used for enumeration
  - We enable Zmap to send multiple packets
  - We can manually craft connections and manipulate them

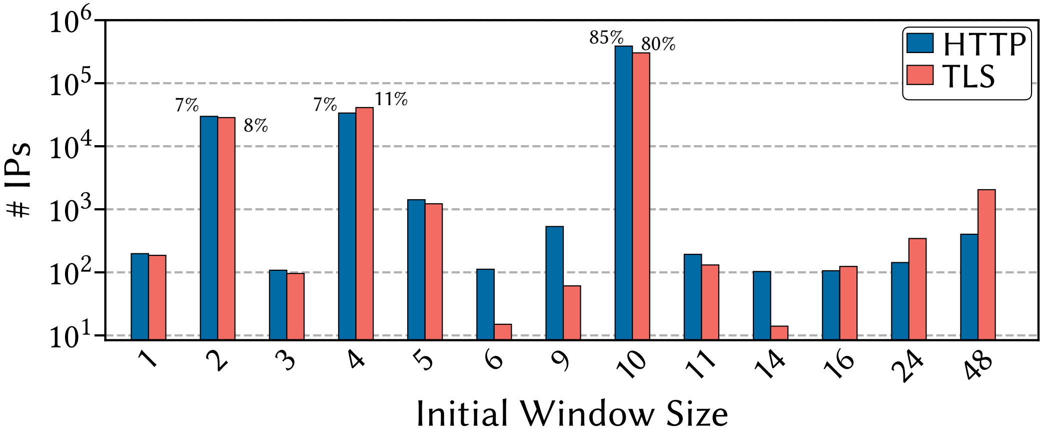
- **Modified ZMap, HTTP/TLS scanners available on Github**
  - https://github.com/COMSYS/zmap

Results – IPv4 HTTP/TLS



- **TLS and HTTP do not agree**
  - Many TLS hosts still use IW 4
- **HTTP scan triggers many abuse mails**
  - In contrast to TLS, this appears in access logs
- **How much scanning is enough?**

Results – Alexa 1M

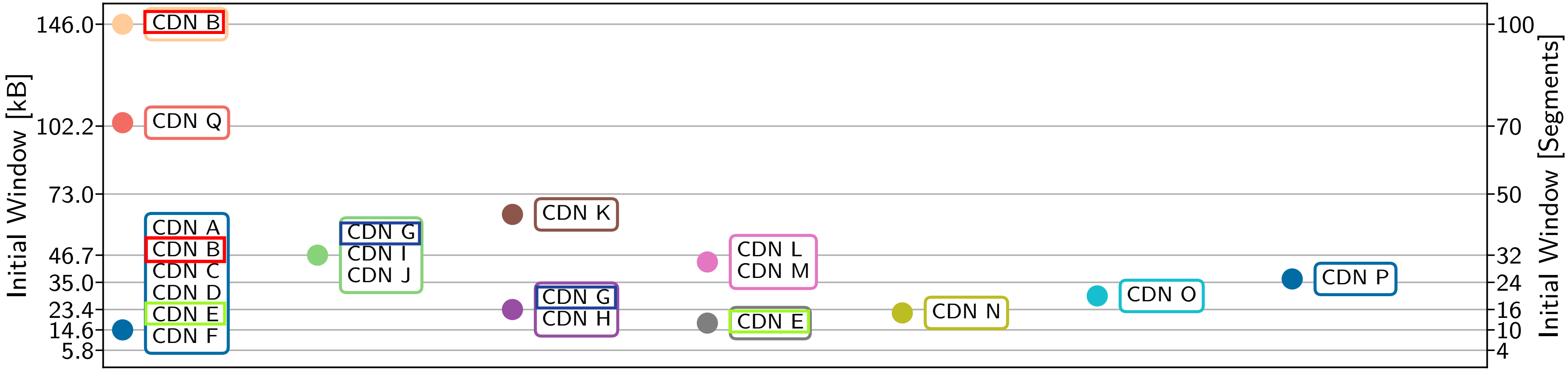


- **Most people in the Alexa list follow current RFCs**
  - Here: similar distribution for HTTP and TLS

- **Generally, we see older IWs in Access Networks**
- **What about CNDs?**

Content Delivery Networks

- **Get large URLs from HTTPArchive for each CDN**
  - ▶ Use regular-sized MSS (enough data)
  - ▶ Use HTTP to request resources
  - ▶ Also announce Window Scaling Option

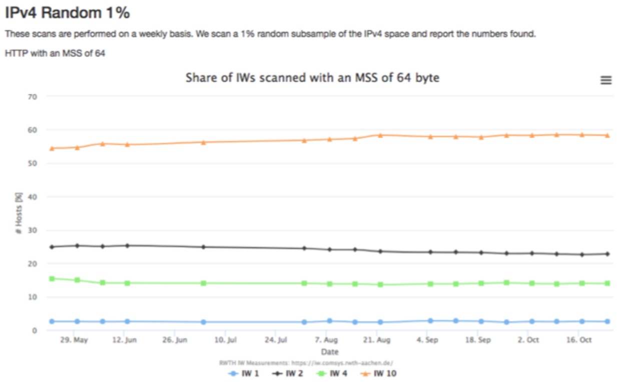


  - ▶ CDN B is 10x over current IETF standard, most are under IW 50
  - ▶ CDNs customize IWs for different customers

Conclusion

- **Distributions dominated by RFC-recommended values**
  - Still a lot of IW 2 and IW 4
  - Popular hosts seem to be on IW 10
- **We also find some customization**
  - Some hosts have very large IWs
  - CDNs are far beyond current standards
    - Some even customize for different networks

- **Periodic 1% scans are available at https://iw.netray.io**
- **Source code available at https://github.com/COMSYS/zmap**



IPv4 Random 1%

These scans are performed on a weekly basis. We scan a 1% random subsample of the IPv4 space and report the numbers found.

HTTP with an MSS of 64

Thank you!

Any questions?

Thanks to RWTH Aachen ITC for enabling our measurements