

Ah, this is how packets get here!

Measuring reverse paths on the public internet

Valentin Heinrich





Traceroute (TR)

- Traceroute (TR) is sometimes referred to as “the number one go-to tool for troubleshooting problems on the Internet”¹
- While it appears simple, it can be challenging to interpret its results
- This talk is about **reverse** traceroute
 - A protocol extension of ICMP:
<https://datatracker.ietf.org/doc/html/draft-heiwin-intarea-reverse-traceroute>
 - An implementation:
<https://github.com/HSAnet/reverse-traceroute>
(also available as Debian package)

¹ A Practical Guide to (Correctly) Troubleshooting with Traceroute, Richard Steenbergen, NANOG 80, <https://youtu.be/L0RUI5kHzeQ>



Analyse this!

You suspect a problem. You run traceroute. You get the following output.

```
1  routerA.aug.net-a.com (10.10.10.10)  1ms    2ms    1ms
2  routerB.muc.net-a.com (20.20.20.20)  5ms    6ms    12ms
3  routerC.fra.net-a.com (30.30.30.30)  11ms   21ms   14ms
4  routerD.fra.net-b.com (40.40.40.40)  340ms  320ms  350ms
5  www.example.com      (50.50.50.50)  345ms  310ms  360ms
```

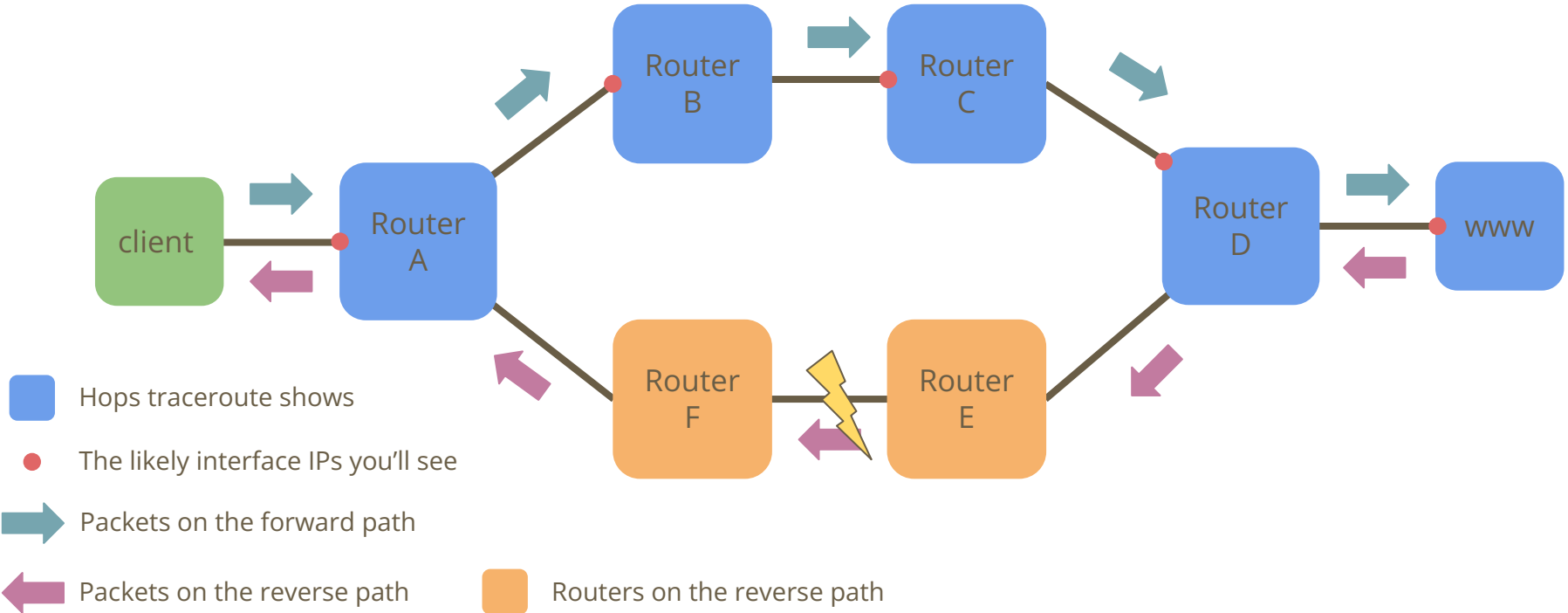
What is your conclusion?

- A. Problem? What problem? This is how I would expect the output to be.
- B. There is something wrong between routers C and D (hops 3 and 4).
- C. You cannot really tell given this output alone.

Well...



1	routerA.aug.net-a.com	(10.10.10.10)	1ms	2ms	1ms
2	routerB.muc.net-a.com	(20.20.20.20)	5ms	6ms	12ms
3	routerC.fra.net-a.com	(30.30.30.30)	11ms	21ms	14ms
4	routerD.fra.net-b.com	(40.40.40.40)	340ms	320ms	350ms
5	www.example.com	(50.50.50.50)	345ms	310ms	360ms





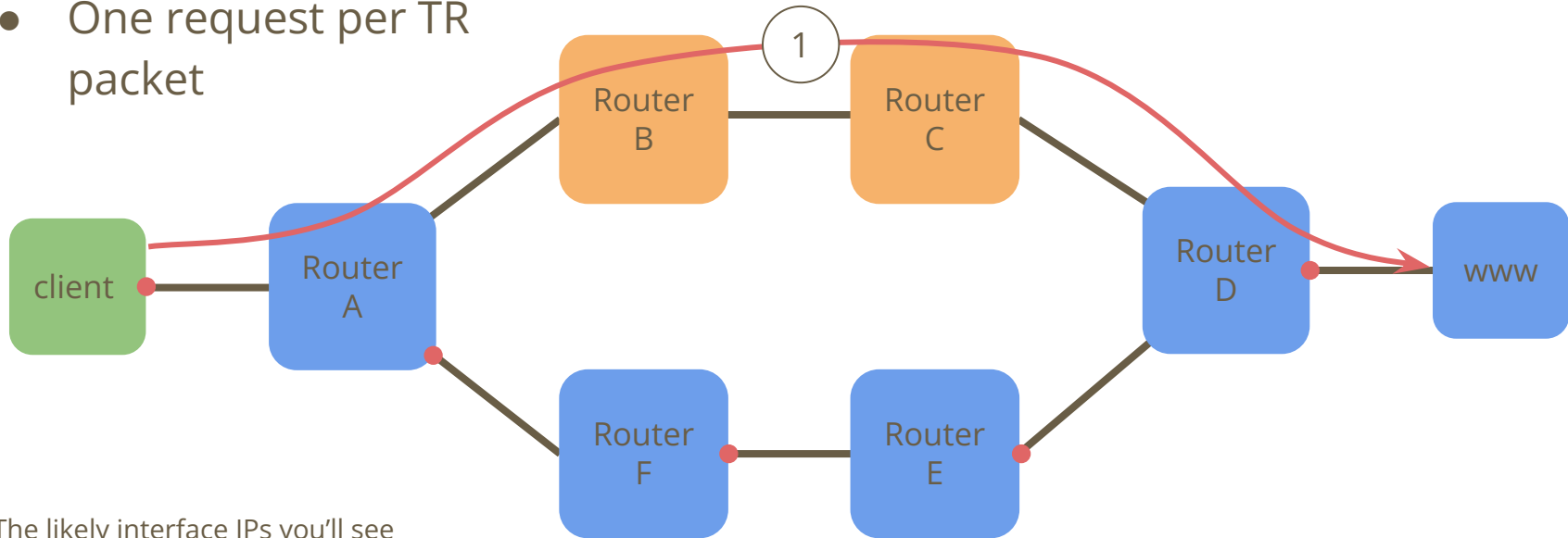
One past attempt

- "Traceroute Using an IP Option", RFC 1393, January 1993
 - A special IPv4 option is added to TR packets (incl. the IP address of the originator)
 - Causes a router to send a special TR message to the originator
 - Packet with the option is simply forwarded
 - The receiver also sends a packet incl. above option with the originators address
- Why don't we have this yet?
 - Well, likely the need for router support and the use of IP options
 - Could spoof originator's IP / Amplification attack vector
 - It teaches us to be careful with design choices
 - RFC 1393 was obsoleted in 2012



Meet reverse traceroute

- Uses a new ICMP request to trigger a reverse traceroute
- One request per TR packet



● The likely interface IPs you'll see

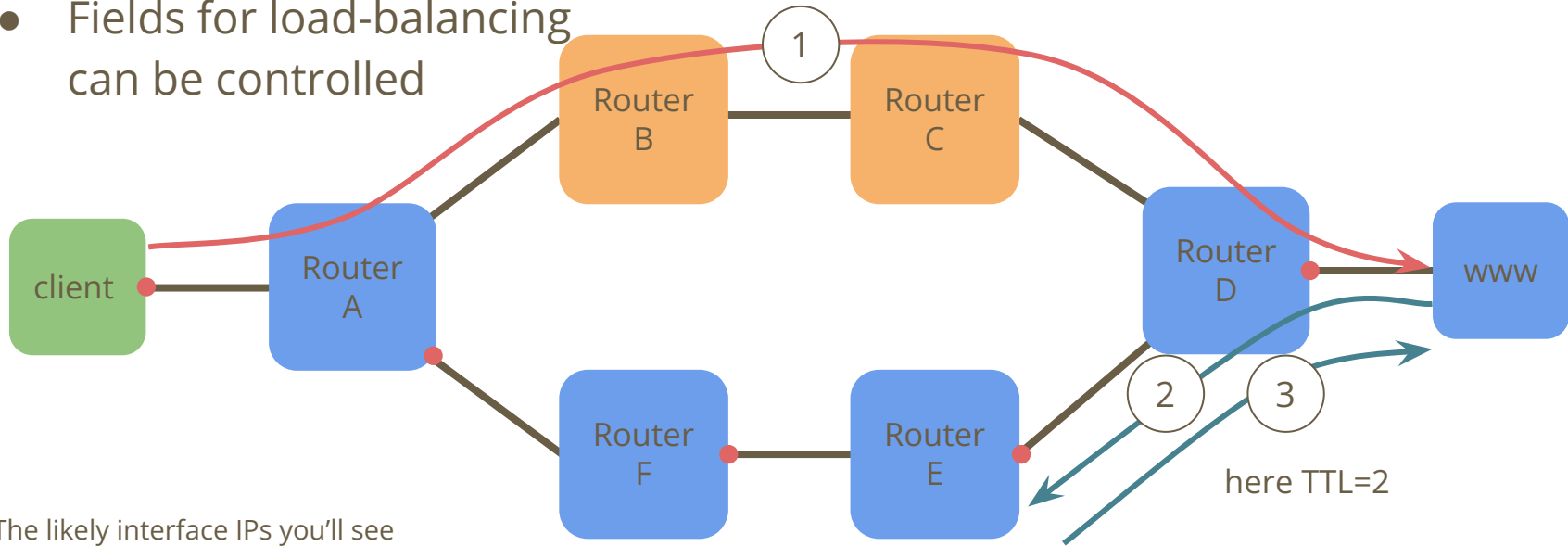
■ Routers reverse traceroute shows

■ Routers on the forward path



Meet reverse traceroute

- A regular TR packet is sent (UDP, ICMP or TCP)
- Fields for load-balancing can be controlled



• The likely interface IPs you'll see

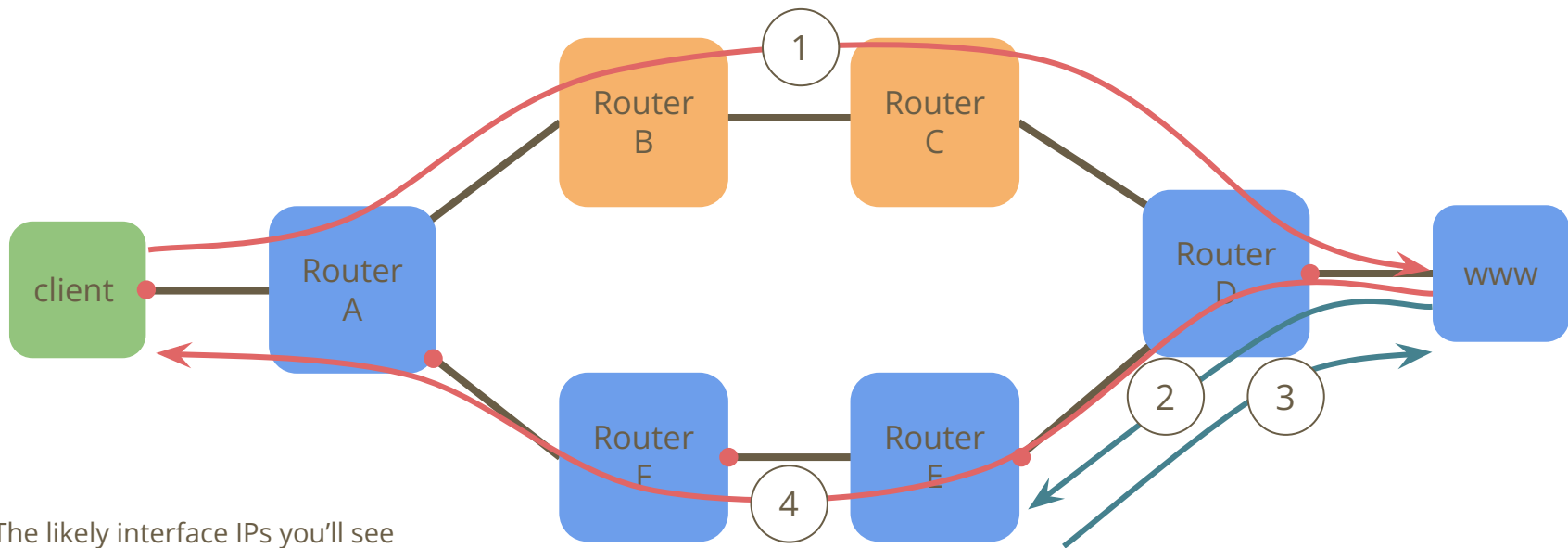
• Routers reverse traceroute shows

• Routers on the forward path



Meet reverse traceroute

- For that single probe, an ICMP response is sent back



● The likely interface IPs you'll see

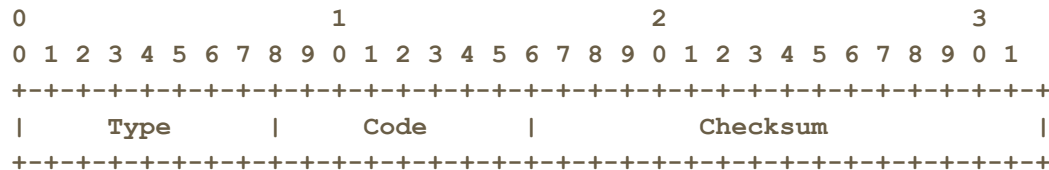
■ Routers reverse traceroute shows

■ Routers on the forward path



Headers, code points ... oh my

- Reverse Traceroute is defined for both ICMP and ICMPv6
- ICMP messages typically start like this:



- Question, which `Type` and `Code` to use:
 - Option A: New types and codes
 - Option B: Existing type and new codes
- Real question: which ones work on today's internet



What about middleboxes?

- The internet is ossified, mainly thanks to middleboxes
 - NATs e.g., are a pretty common middlebox
- Question: which packets go through NATs
- Tested 12 NAT implementation:
 - We sent two packets with type 8 (used by ping request) and codes 1 and 2 (standard ping uses 0), replies matched the code but used type 0
 - And two unassigned types (7 and 252) with code 0 each

ICMP request	forwarded	filtered	bypassed
Type 8, code 1	11	1 ^{a)}	0
Type 8, code 2	11	1 ^{a)}	0
Type 7, code 0	1	7	4
Type 252, code 0	1	6	5

^{a)} Response dropped

But what happens to those packets on the internet?



- We picked ten million IPv4 addresses at random and send an ICMP Echo request there (good old Ping)
- For each host that responded, we sent an ICMP Packet with the Echo type but a different code (code 1)

Filtered	Reflective	Unreflective	Erroneous
39.993	931.427	32.478	659 ^{a)}

^{a)} mostly dest. unreach.



Conclusion

- Call for action
 - Read the draft and join the discussion at the IntArea WG (IETF)
 - Offer to host a reverse traceroute end-point
 - Use our reverse traceroute client and send us the output (or maybe offer reverse traceroute as part of **RIPE Atlas**)
- Website: <https://net.hs-augsburg.de/en/project/reverse-traceroute/>
- Github: <https://github.com/HSAnet/reverse-traceroute>
- Contact: valentin.heinrich@hs-augsburg.de