CSE 291
INTERNET DATA SCIENCE FOR CYBERSECURITY

1 February 2023

Router-level Internet Topology

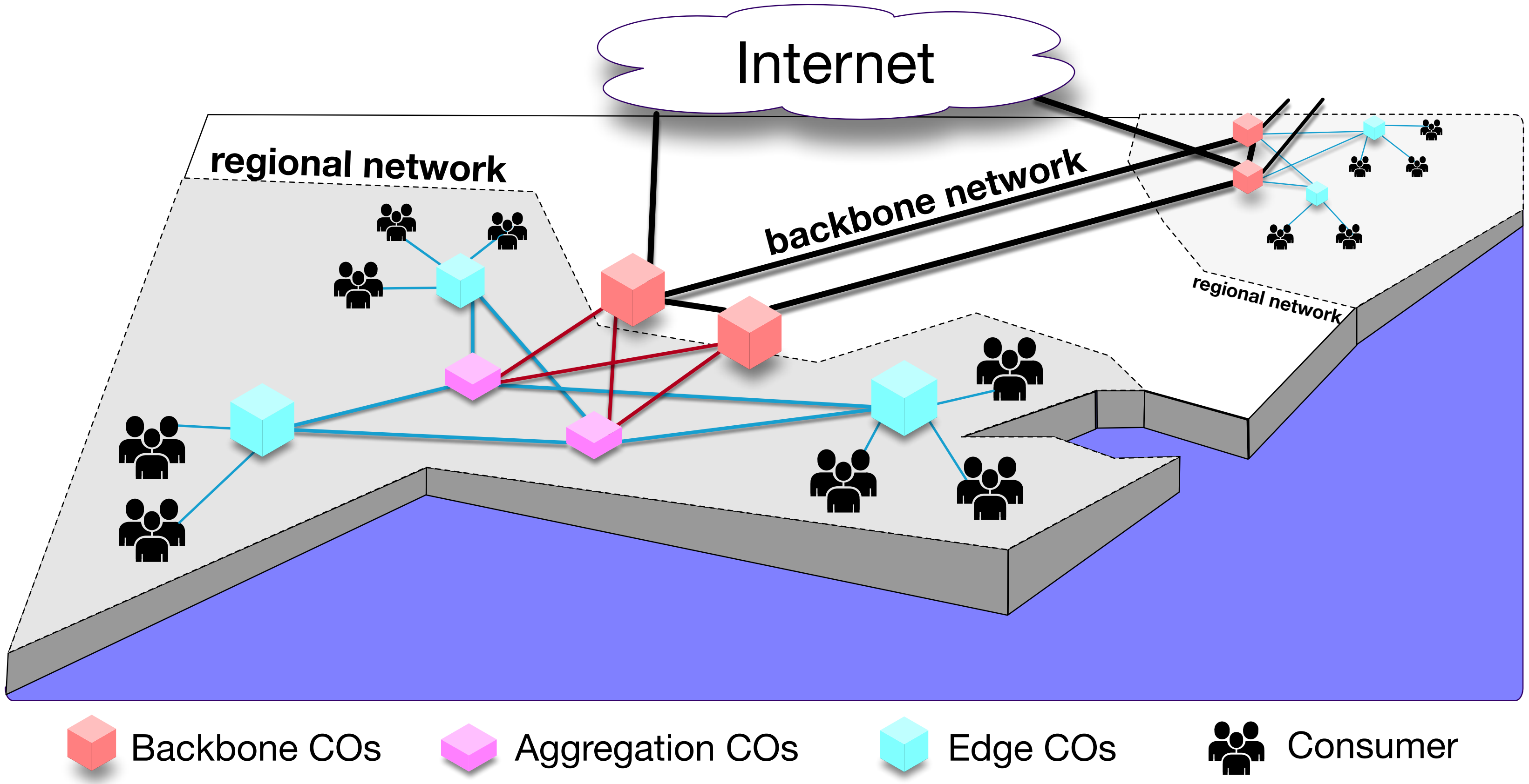
OVERVIEW

1. Background to support future work in this class, without an assigned paper (but possible projects)

LEARNING OBJECTIVES

1. Be able to describe how traceroute works
2. Appreciate how operators assign IP addresses to router interfaces
3. Be able to reason about router properties based on hostname annotations

MOTIVATION

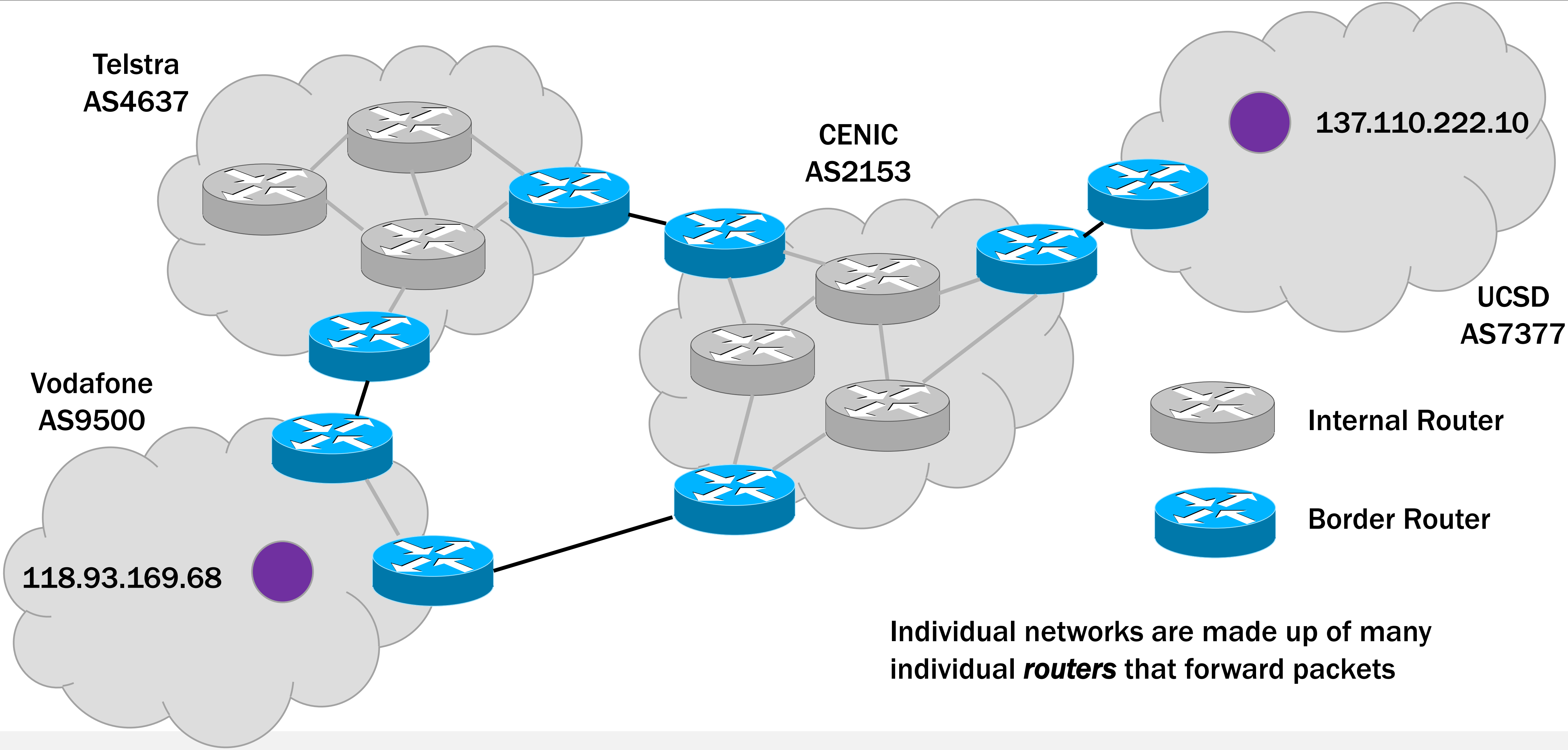


Are networks designed to withstand damage?

Where is attacker infrastructure located?

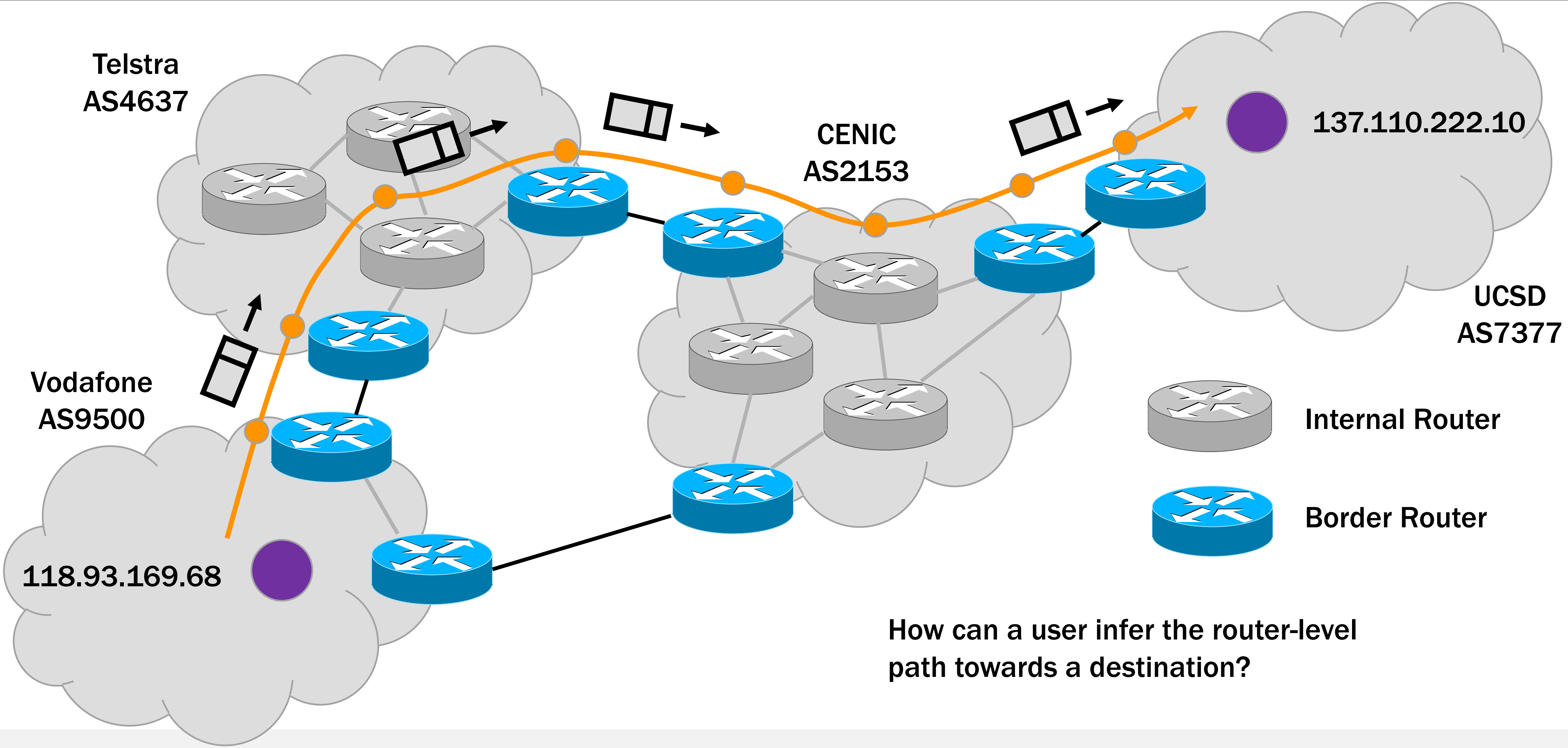
What would be affected if a network segment was damaged?

AUTONOMOUS SYSTEMS



Individual networks are made up of many individual ***routers*** that forward packets

UNCOVERING PATHS



How can a user infer the router-level path towards a destination?

TRACEROUTE

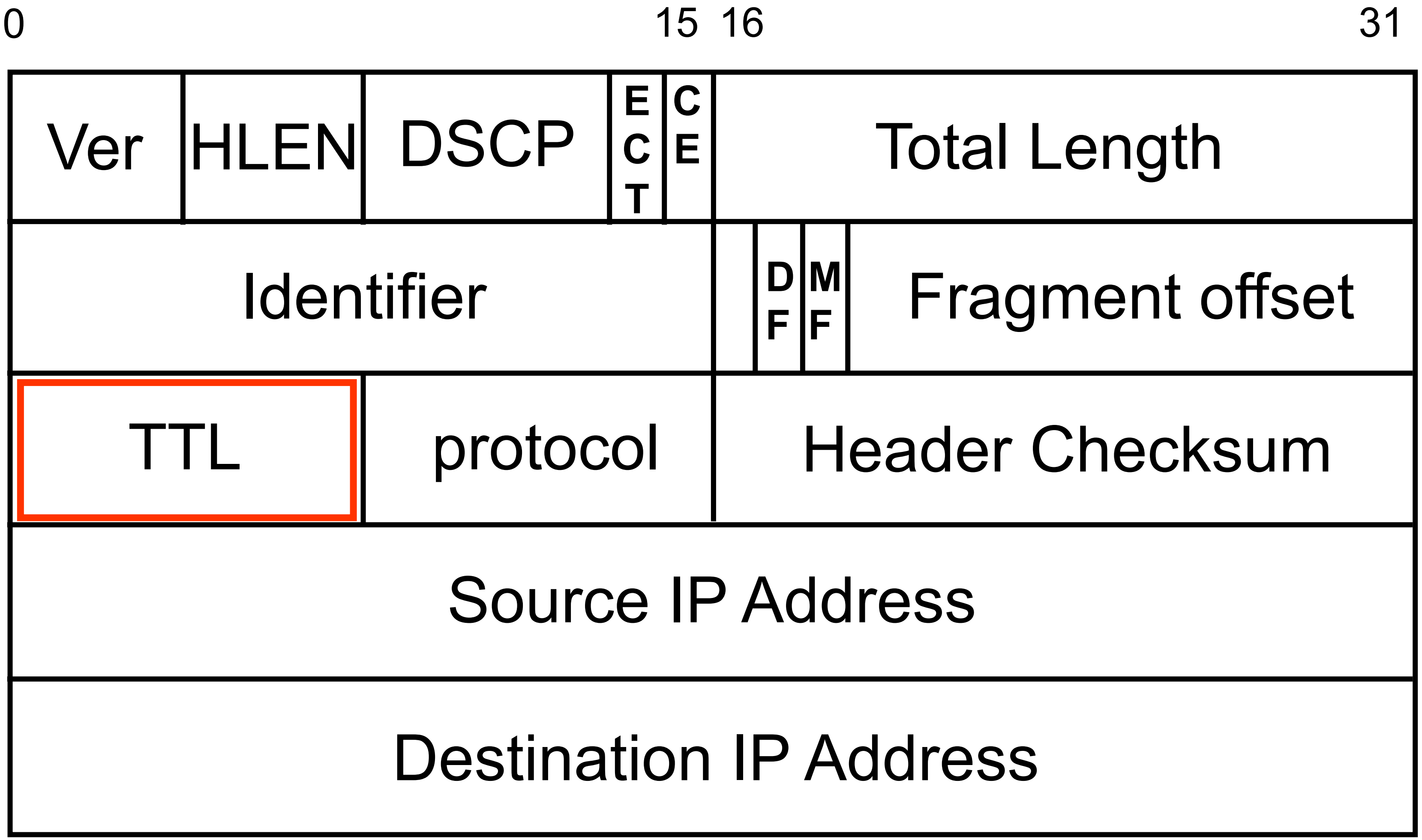
- The Internet does not have built-in path tracing facility
- Instead, we use a *hack* to infer the path, known as *traceroute*

IPV4 HEADER

TTL: Time-to-Live

Decremented by each router in the path to prevent a packet circling indefinitely in a loop.

When TTL reaches 1, router sends an error message to the source of the packet.

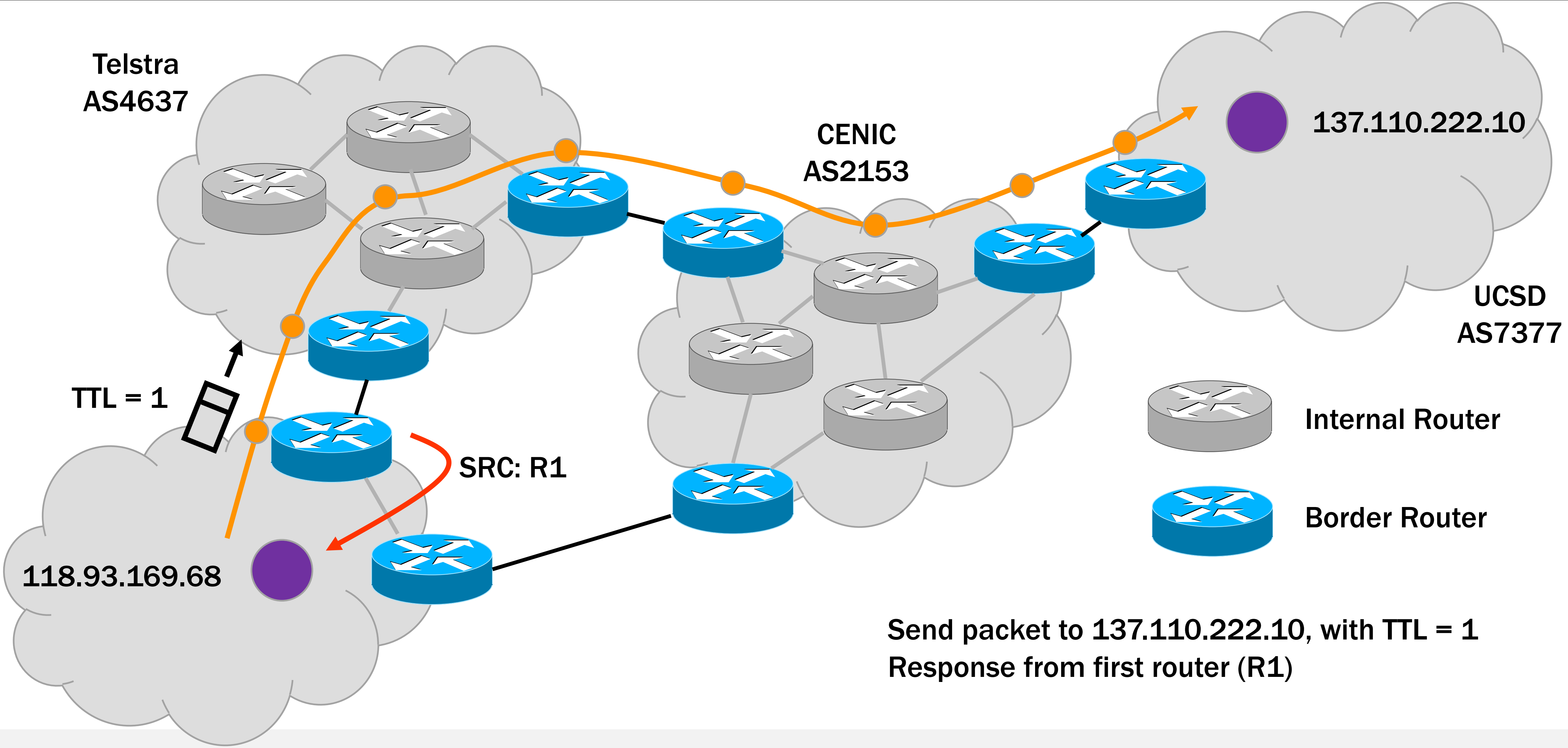


TRACEROUTE

- Send packets toward a destination, starting with TTL=1 to obtain a response from the first router
- Increment the TTL to obtain a response from the second router, and so on.

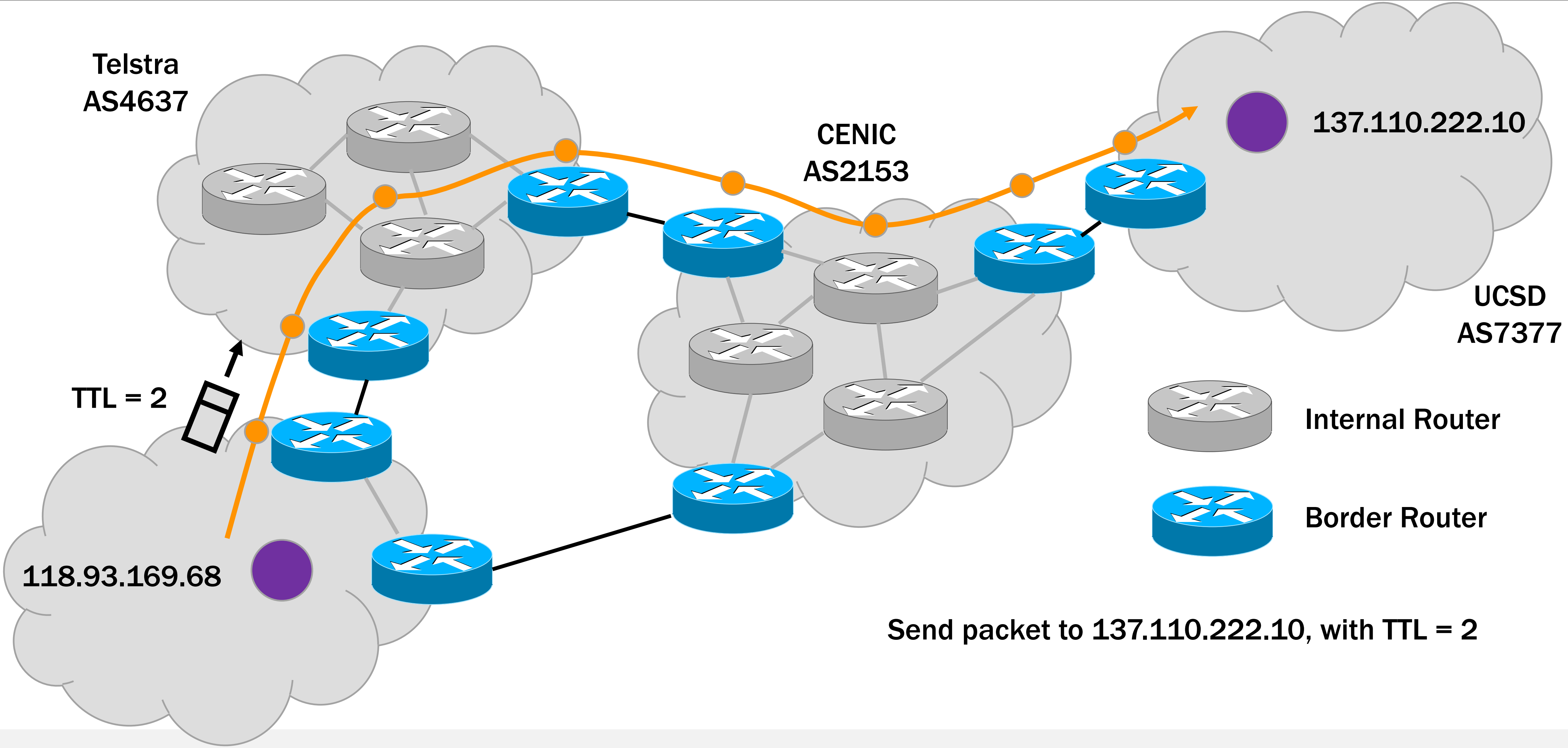
- Issue: we can only observe the forward path
- Issue: routers do not have to reply to probes

UNCOVERING PATHS



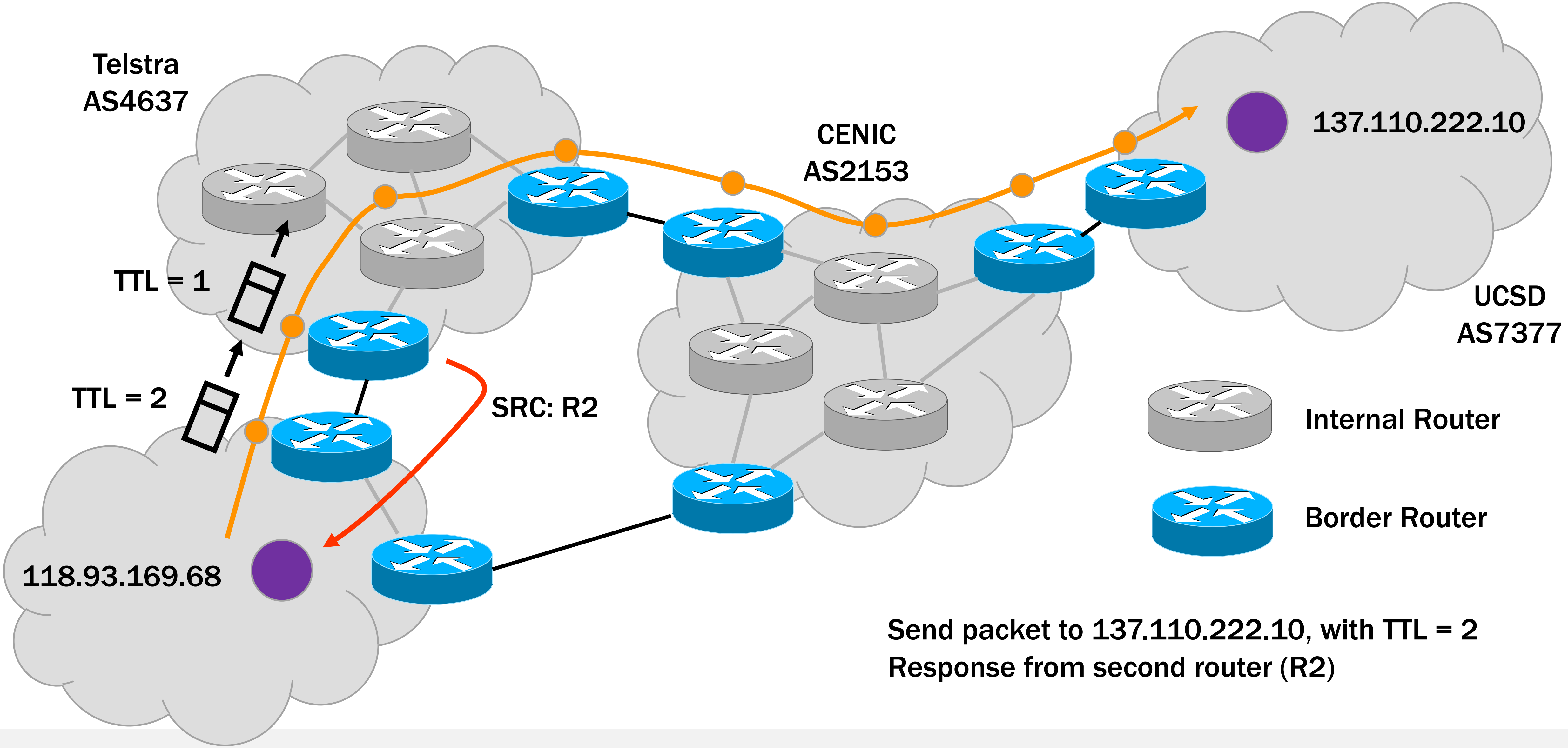
Send packet to 137.110.222.10, with TTL = 1
Response from first router (R1)

UNCOVERING PATHS



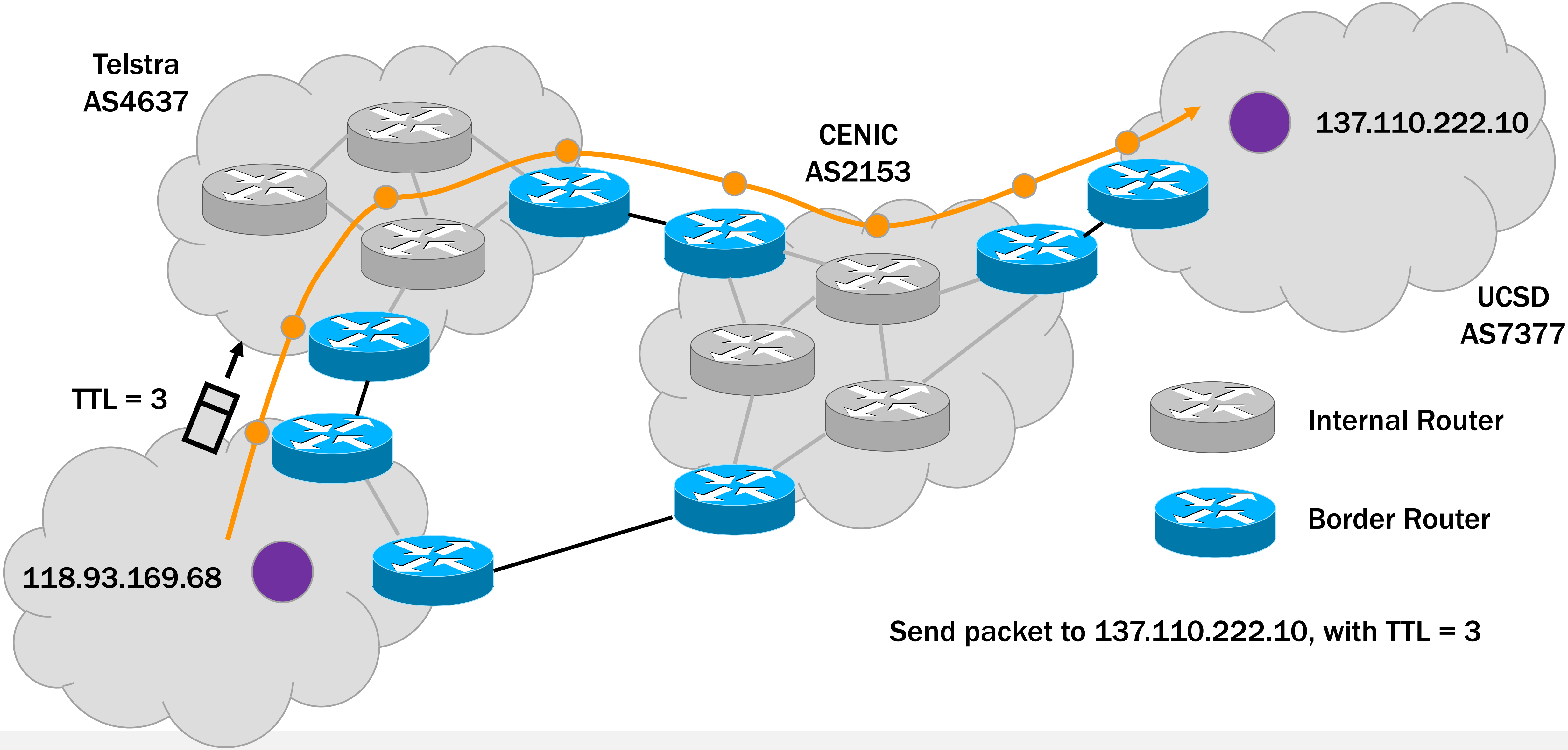
Send packet to 137.110.222.10, with TTL = 2

UNCOVERING PATHS



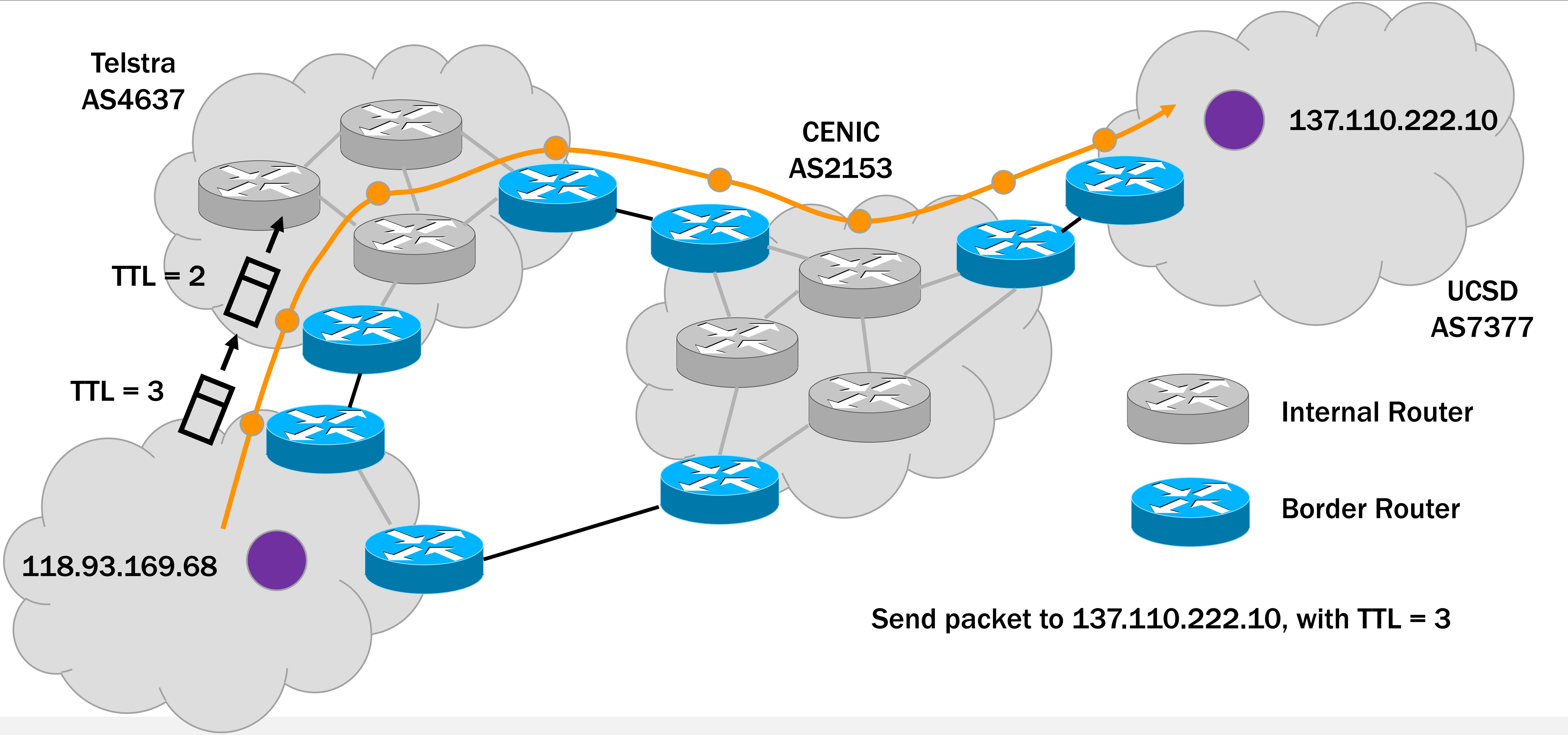
Send packet to 137.110.222.10, with TTL = 2
Response from second router (R2)

UNCOVERING PATHS



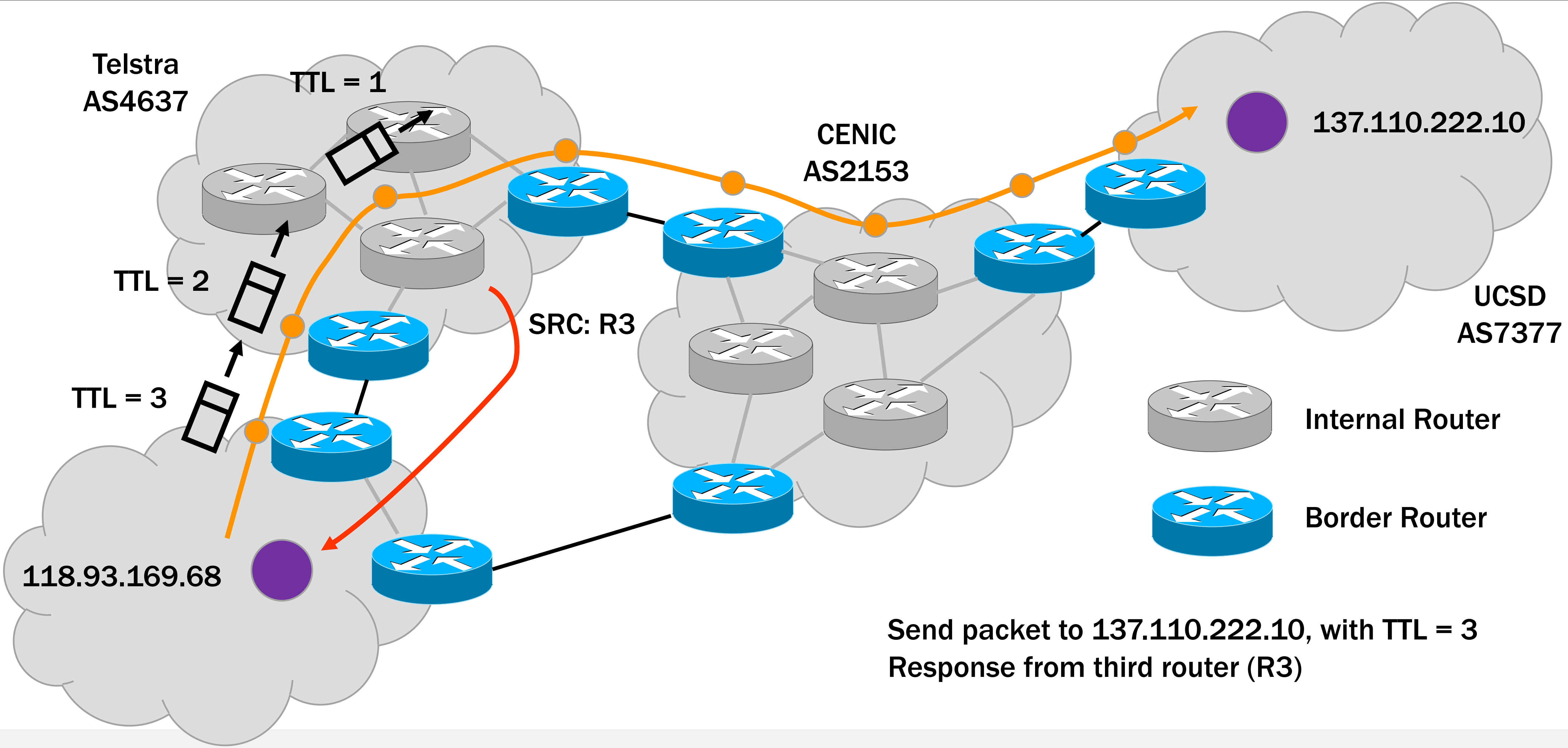
Send packet to 137.110.222.10, with TTL = 3

UNCOVERING PATHS



Send packet to 137.110.222.10, with TTL = 3

UNCOVERING PATHS



Send packet to 137.110.222.10, with TTL = 3
Response from third router (R3)

TRACEROUTE OUTPUT

```
$ traceroute -q 1 sysnet.ucsd.edu
traceroute to sysnet.ucsd.edu (137.110.222.10), 64 hops max, 40 byte packets
 1  118-93-171-254.dsl.dyn.ihug.co.nz (118.93.171.254)  11.731 ms
 2  134.159.174.38 (134.159.174.38)  13.757 ms
 3  134.159.174.41 (134.159.174.41)  14.591 ms
 4  i-93.tauc-core02.telstraglobal.net (202.84.227.53)  14.666 ms
 5  202.84.138.82 (202.84.138.82)  139.923 ms
 6  202.84.138.82 (202.84.138.82)  139.267 ms
 7  i-91.tlot02.telstraglobal.net (202.40.149.177)  137.899 ms
 8  lsan0.tr-cps.internet2.edu (206.223.123.199)  139.149 ms
 9  fourhundredge-0-0-0-0.4079.core2.losa.net.internet2.edu (163.253.1.198) 138.479 ms
10  64.57.20.83 (64.57.20.83)  136.953 ms
11  dc-tus-agg8--lax-agg8-300g.cenic.net (137.164.11.83)  139.387 ms
12  dc-sdg-agg4--tus-agg8-300g.cenic.net (137.164.11.85)  138.764 ms
13  dc-ucsd-100ge--sdg-agg4.cenic.net (137.164.23.177)  138.881 ms
14  nodem-core-6807-vlan2761-gw.ucsd.edu (132.239.254.163)  138.417 ms
15  agg-b-mcore.ucsd.edu (132.239.254.44)  138.709 ms
16  sysnet.sysnet.ucsd.edu (137.110.222.10)  138.334 ms
```

TRACEROUTE OUTPUT

caida

```
$ traceroute -q 1 sysnet.ucsd.edu
traceroute to sysnet.ucsd.edu (137.110.222.10), 64 hops max, 40 byte packets
 1  118-93-171-254.dsl.dyn.ihug.co.nz (118.93.171.254)  11.731 ms
 2  134.159.174.38 (134.159.174.38)  13.757 ms
 3  134.159.174.41 (134.159.174.41)  14.591 ms
 4  i-93.tauc-core02.telstraglobal.net (202.84.227.53)  14.666 ms
 5  202.84.138.82 (202.84.138.82)  139.923 ms
 6  202.84.138.82 (202.84.138.82)  139.267 ms
 7  i-91.tlot02.telstraglobal.net (202.40.149.177)  137.899 ms
 8  lsan0.tr-cps.internet2.edu (206.223.123.199)  139.149 ms
 9  fourhundredge-0-0-0-0.4079.core2.losa.net.internet2.edu (163.253.1.198) 138.479 ms
10  64.57.20.83 (64.57.20.83)  136.953 ms
11  dc-tus-agg8--lax-agg8-300g.cenic.net (137.164.11.83)  139.387 ms
12  dc-sdg-agg4--tus-agg8-300g.cenic.net (137.164.11.85)  138.764 ms
13  dc-ucsd-100ge--sdg-agg4.cenic.net (137.164.23.177)  138.881 ms
14  nodem-core-6807-vlan2761-gw.ucsd.edu (132.239.254.163)  138.417 ms
15  agg-b-mcore.ucsd.edu (132.239.254.44)  138.709 ms
16  sysnet.sysnet.ucsd.edu (137.110.222.10)  138.334 ms
```

Auckland

???

Los Angeles

Tustin

San Diego

TRACEROUTE OUTPUT

```
$ traceroute -q 1 sysnet.ucsd.edu
traceroute to sysnet.ucsd.edu (137.110.222.10), 64 hops max, 40 byte packets
 1  118-93-171-254.dsl.dyn.ihug.co.nz (118.93.171.254)  11.731 ms
 2  134.159.174.38 (134.159.174.38)  13.757 ms
 3  134.159.174.41 (134.159.174.41)  14.591 ms
 4  i-93.tauc-core02.telstraglobal.net (202.84.227.53)  14.666 ms
 5  202.84.138.82 (202.84.138.82)  139.923 ms
 6  202.84.138.82 (202.84.138.82)  139.267 ms
 7  i-91.tlot02.telstraglobal.net (202.40.149.177)  137.899 ms
 8  lsan0.tr-cps.internet2.edu (206.223.123.199)  139.149 ms
 9  fourhundredge-0-0-0-0.4079.core2.losa.net.internet2.edu (163.253.1.198) 138.479 ms
10  64.57.20.83 (64.57.20.83)  136.953 ms
11  dc-tus-agg8--lax-agg8-300g.cenic.net (137.164.11.83)  139.387 ms
12  dc-sdg-agg4--tus-agg8-300g.cenic.net (137.164.11.85)  138.764 ms
13  dc-ucsd-100ge--sdg-agg4.cenic.net (137.164.23.177)  138.881 ms
14  nodem-core-6807-vlan2761-gw.ucsd.edu (132.239.254.163)  138.417 ms
15  agg-b-mcore.ucsd.edu (132.239.254.44)  138.709 ms
16  sysnet.sysnet.ucsd.edu (137.110.222.10)  138.334 ms
```

400Gbps Ethernet

300Gbps

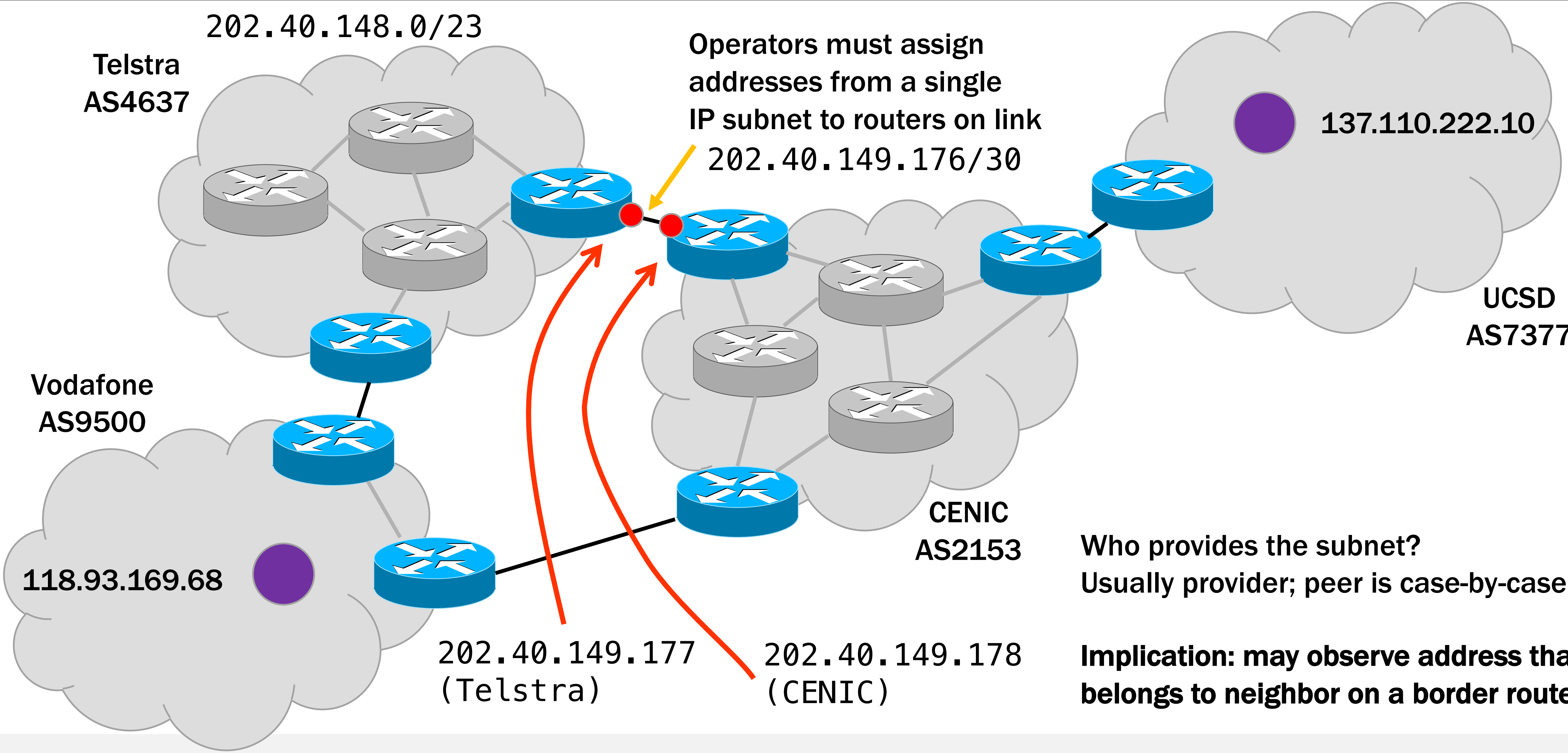
300Gbps

100Gbps Ethernet

HOIHO

- Many operators use internal naming conventions when they assign hostnames to router IP addresses
- Each operator can have a different convention
- *Hoiho* automatically infers regular expressions to extract and decode information in hostnames
- M. Luckie, B. Huffaker, A. Marder, Z. Bischof, M. Fletcher, and k. claffy, **Learning to Extract Geographic Information from Internet Router Hostnames**, ACM CoNEXT, pp. 440-453, Virtual Event, Germany, December 2021.

AUTONOMOUS SYSTEMS



Operators must assign addresses from a single IP subnet to routers on link

Who provides the subnet?
Usually provider; peer is case-by-case.

Implication: may observe address that belongs to neighbor on a border router

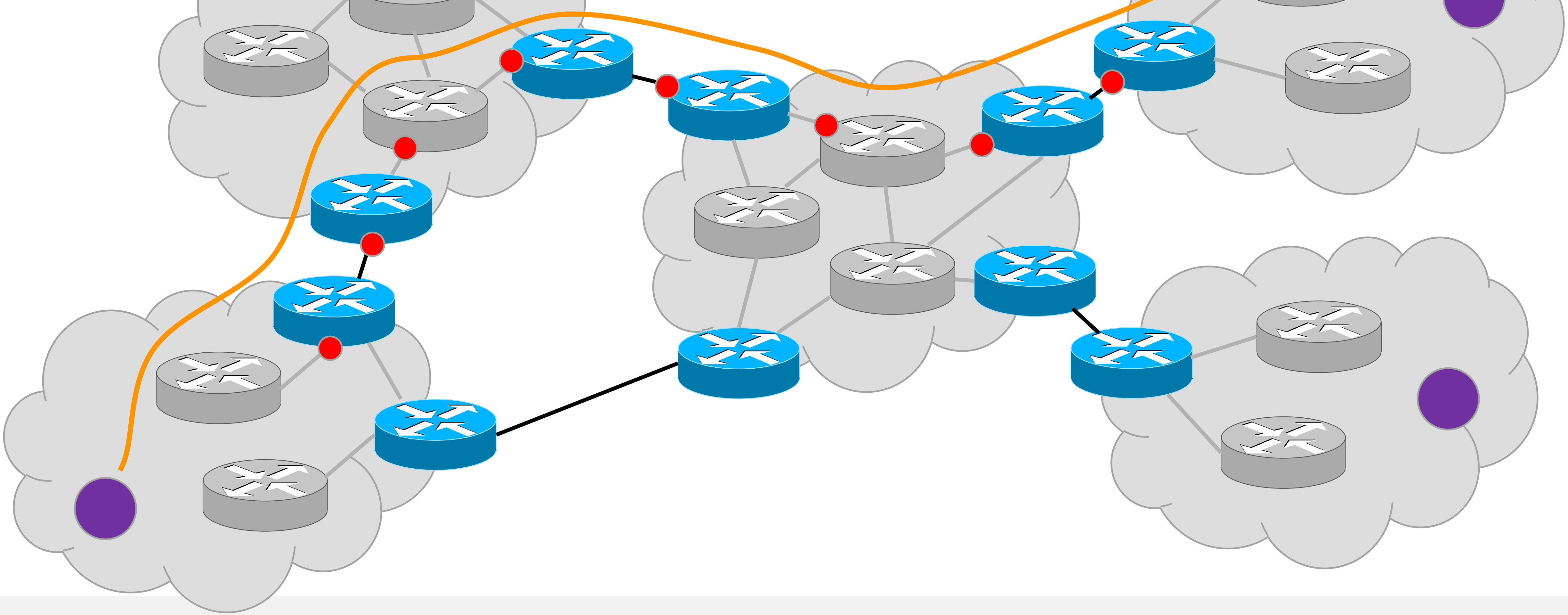
BDRMAPIT

- Using longest matching prefix to infer which AS operates a given router can give misleading results
- *bdrmapIT* uses heuristics to infer router operators

- A. Marder, M. Luckie, A. Dhamdhere, B. Huffaker, J. Smith, and k. claffy, **Pushing the Boundaries with bdrmapIT: Mapping Router Ownership at Internet Scale**, ACM IMC, pp. 56-69, Boston, MA, USA, November 2018.

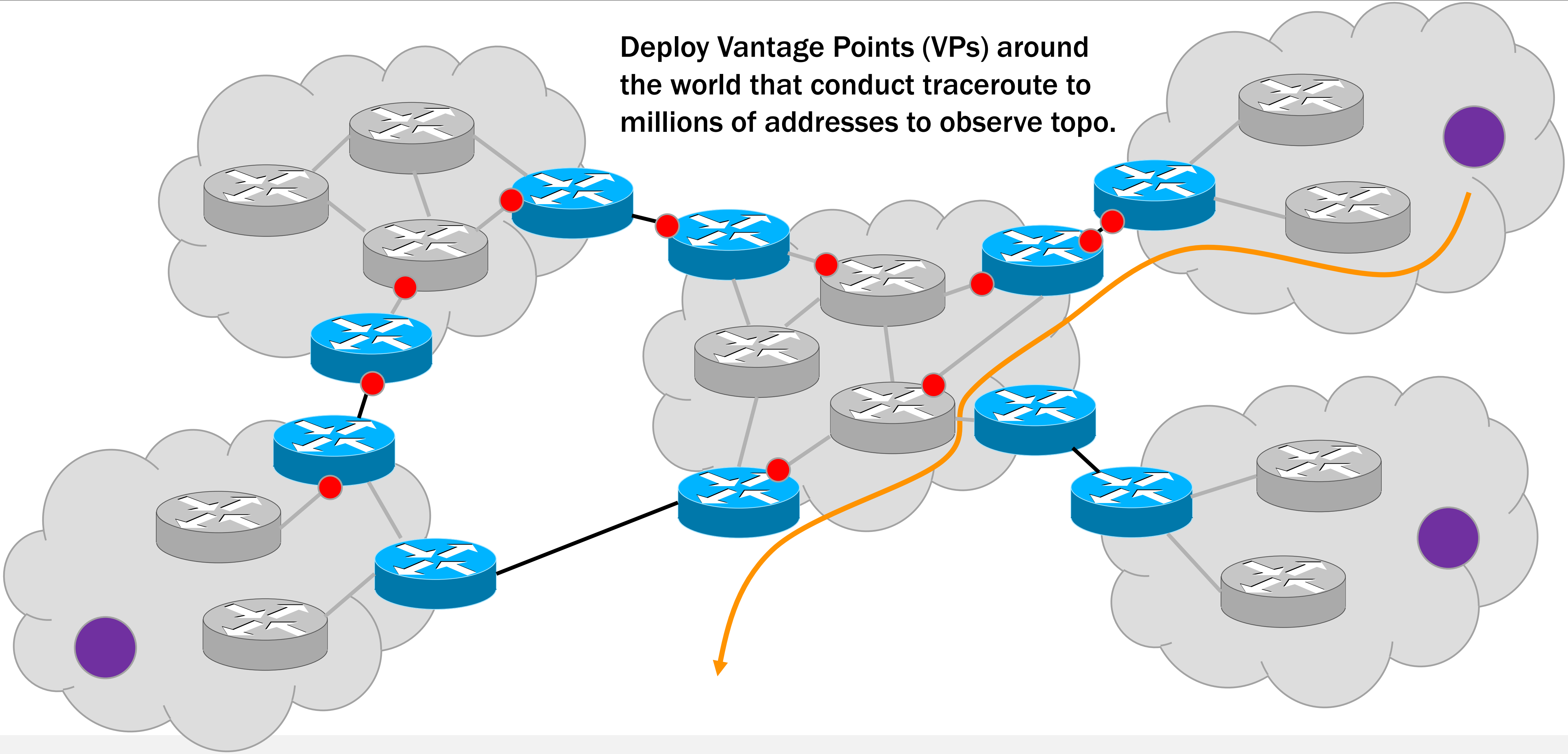
MACROSCOPIC TOPO DISCOVERY

Deploy Vantage Points (VPs) around the world that conduct traceroute to millions of addresses to observe topo.



MACROSCOPIC TOPO DISCOVERY

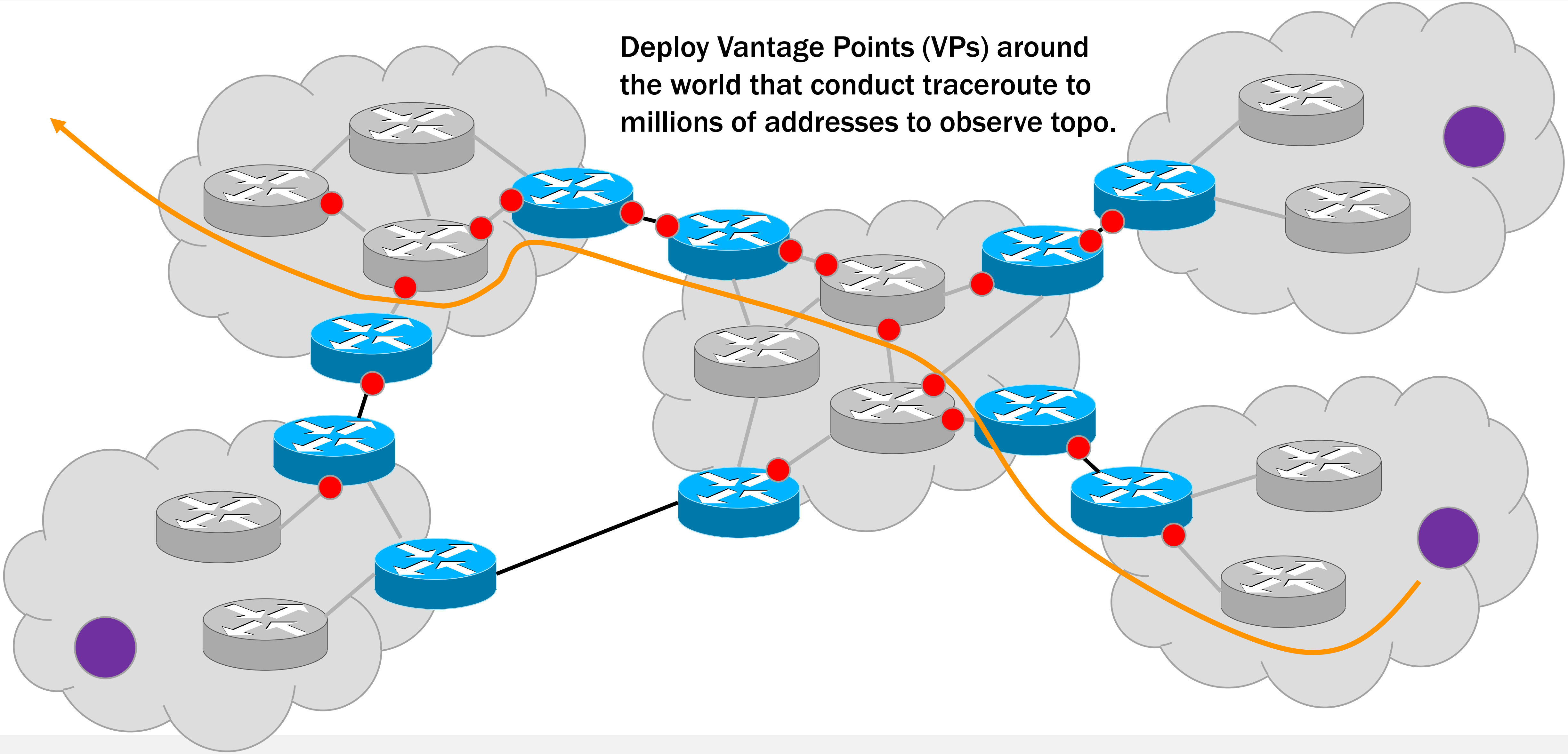
Deploy Vantage Points (VPs) around the world that conduct traceroute to millions of addresses to observe topo.



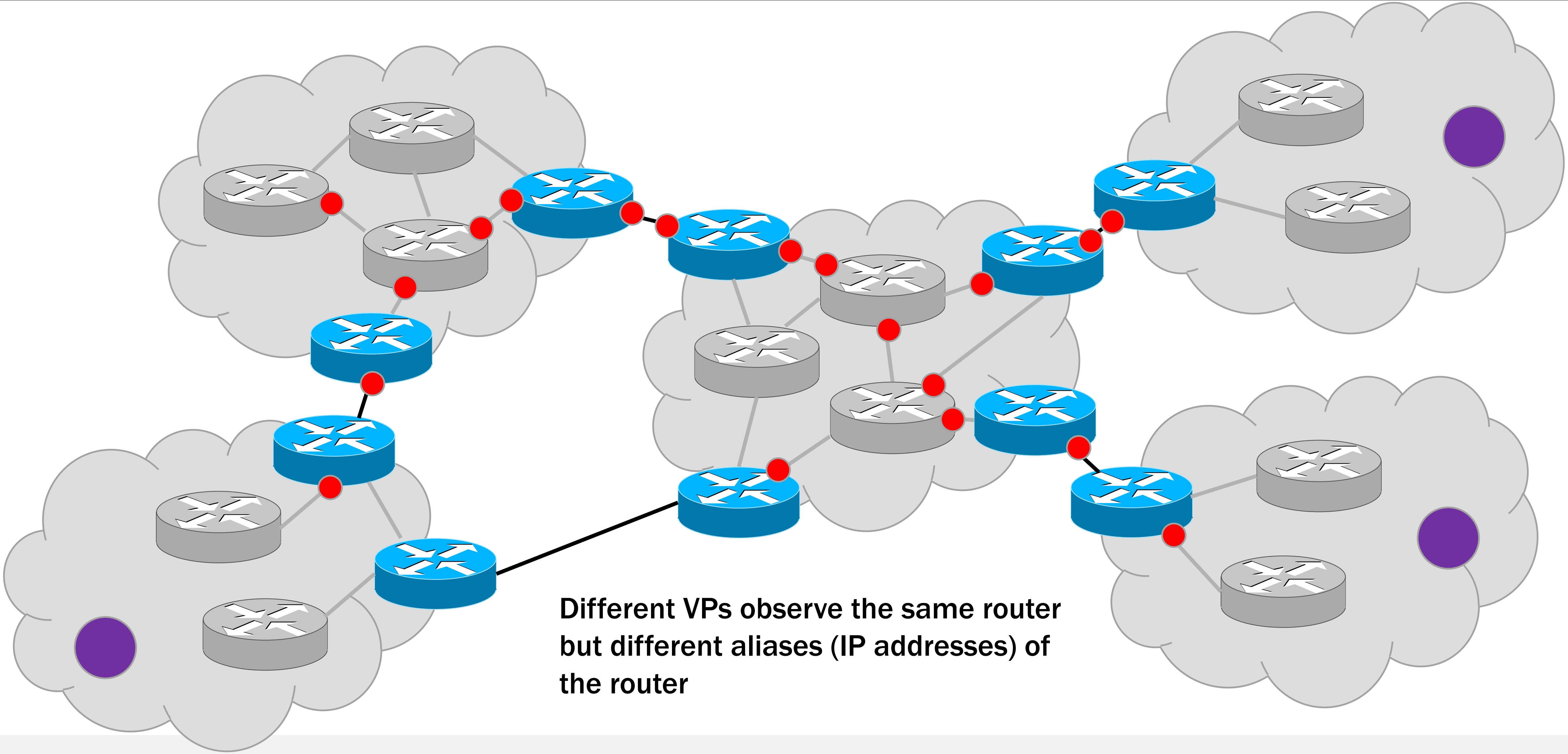
MACROSCOPIC TOPO DISCOVERY

caida

Deploy Vantage Points (VPs) around the world that conduct traceroute to millions of addresses to observe topo.



MACROSCOPIC TOPO DISCOVERY

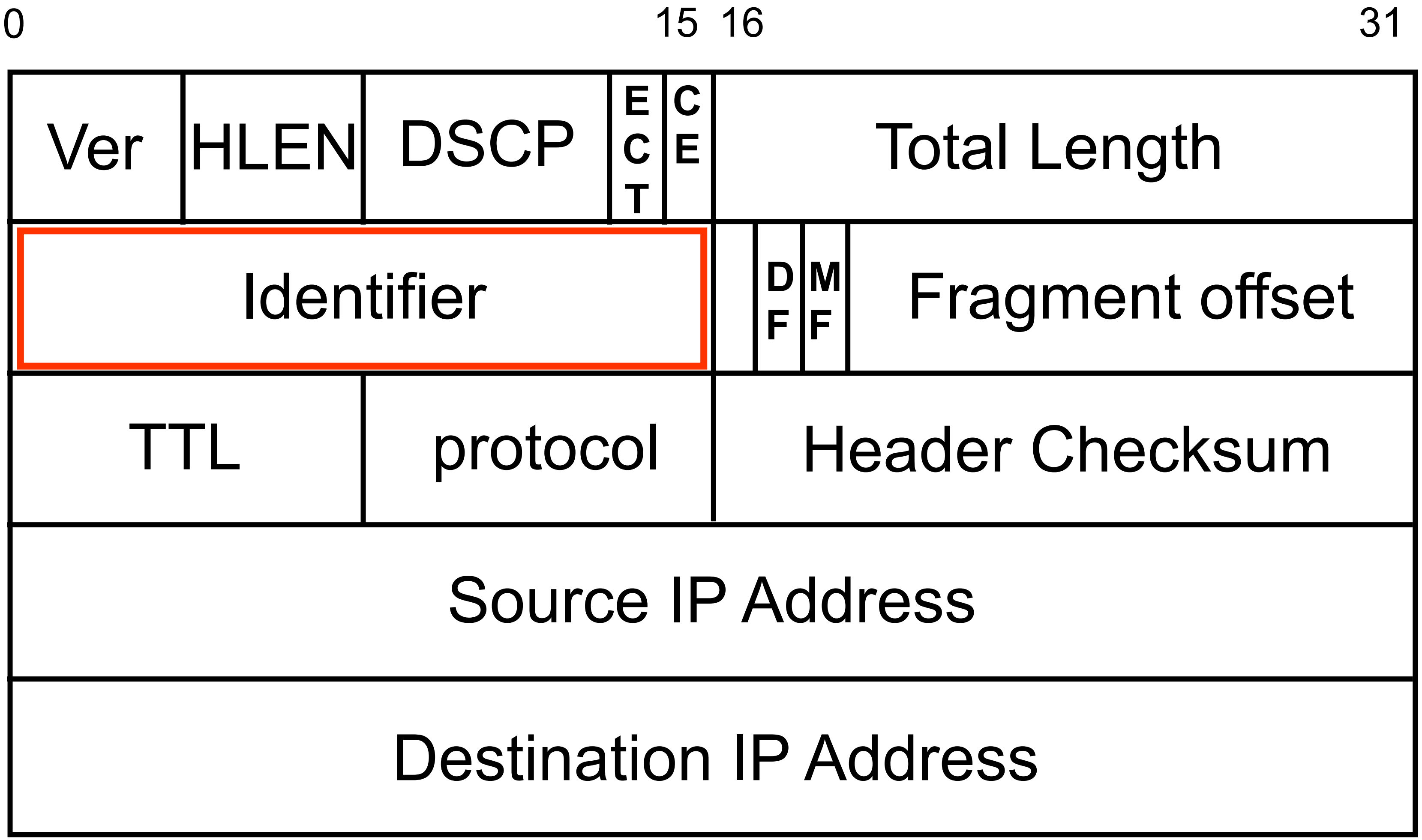


Different VPs observe the same router but different aliases (IP addresses) of the router

ALIAS RESOLUTION

- The Internet does not have built-in method to determine which IP addresses belong to the same router
- Instead, we use *hacks* to infer this, a process known as *alias resolution*
- Many different hacks available, we will talk about one

IPV4 HEADER

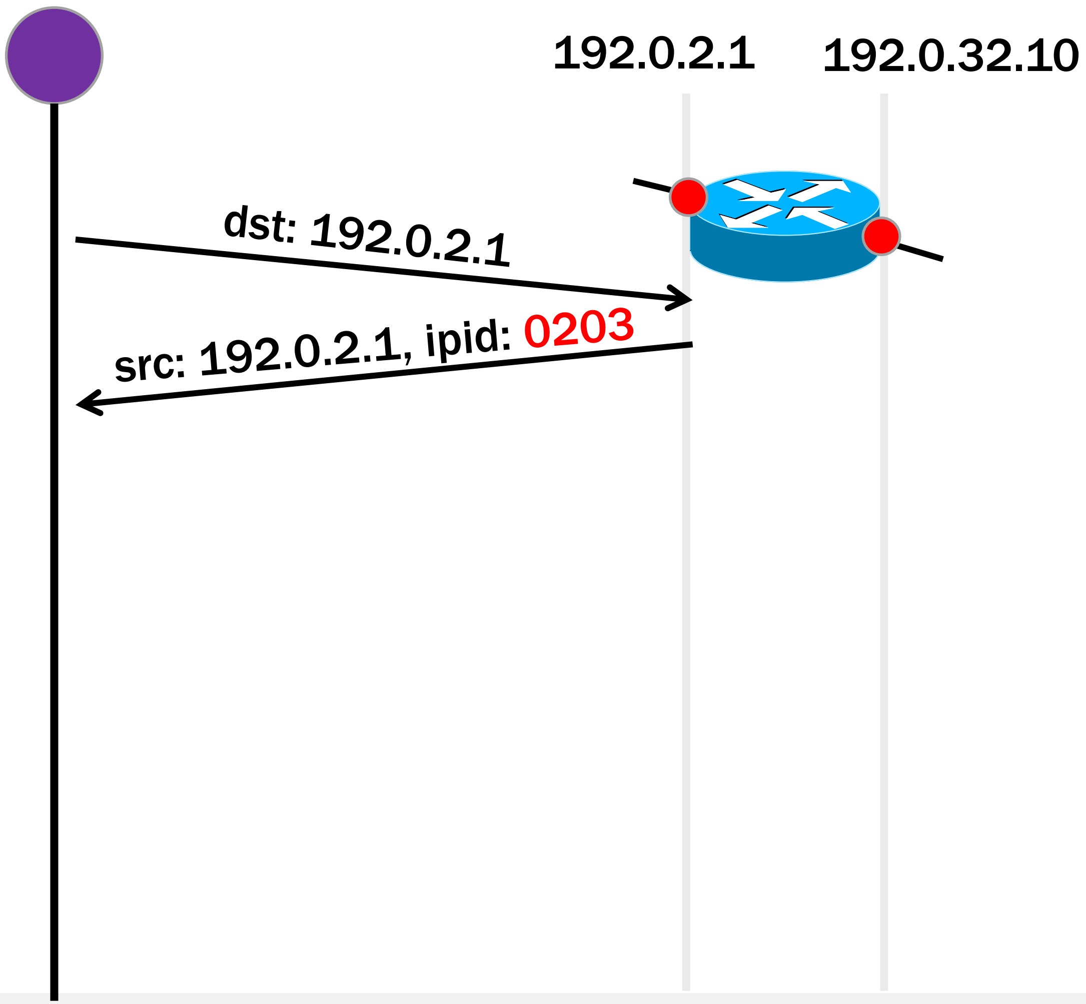


IP Identifier (IPID)

Unique value assigned by sender; if the packet is too large for a subsequent router to forward, it breaks the packet into fragments, assigning the same ID value to each fragment so that the receiver can reassemble.

Implementations often assign from a counter.

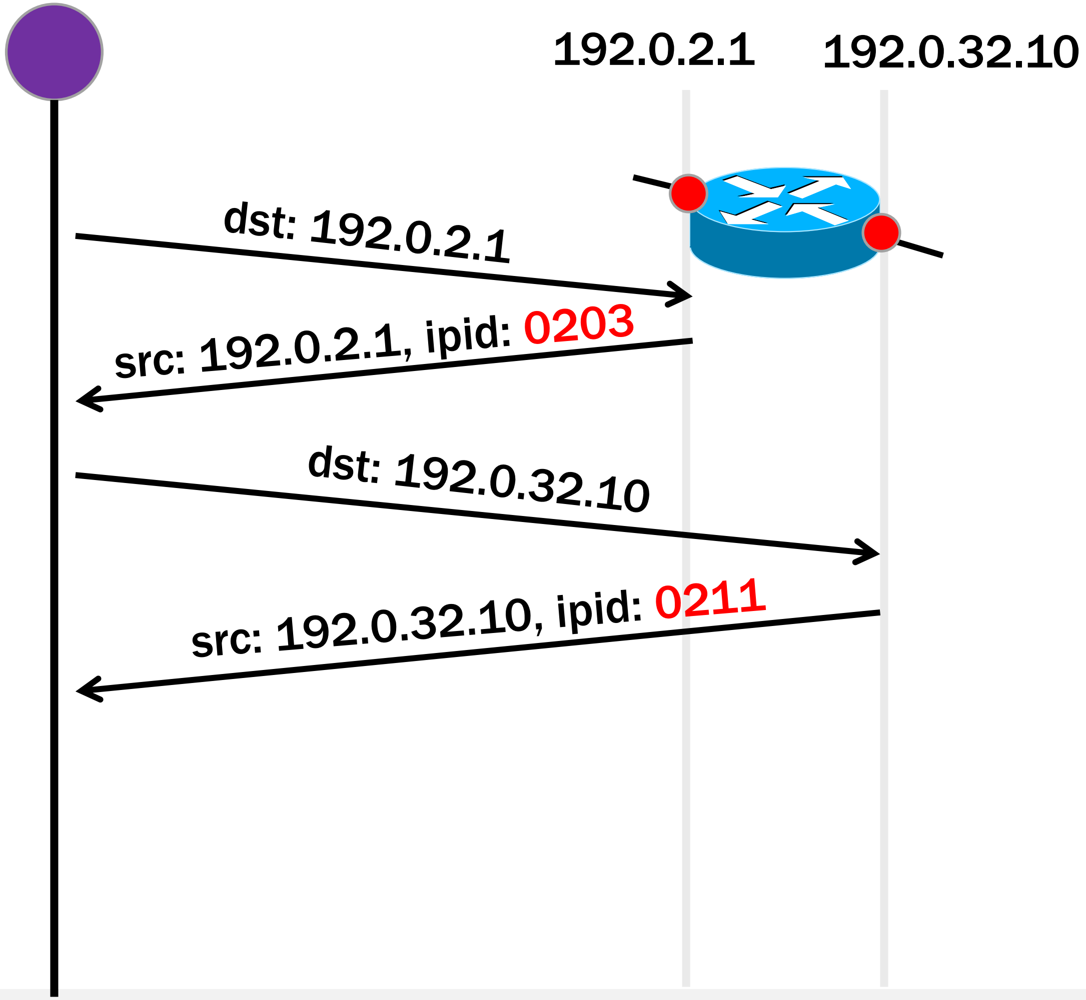
IPID ALIAS RESOLUTION



Idea: Infer that two addresses belong to the same router based on side channels, e.g., IP-ID

Assumption: router assigns IP-ID values from a single central counter.

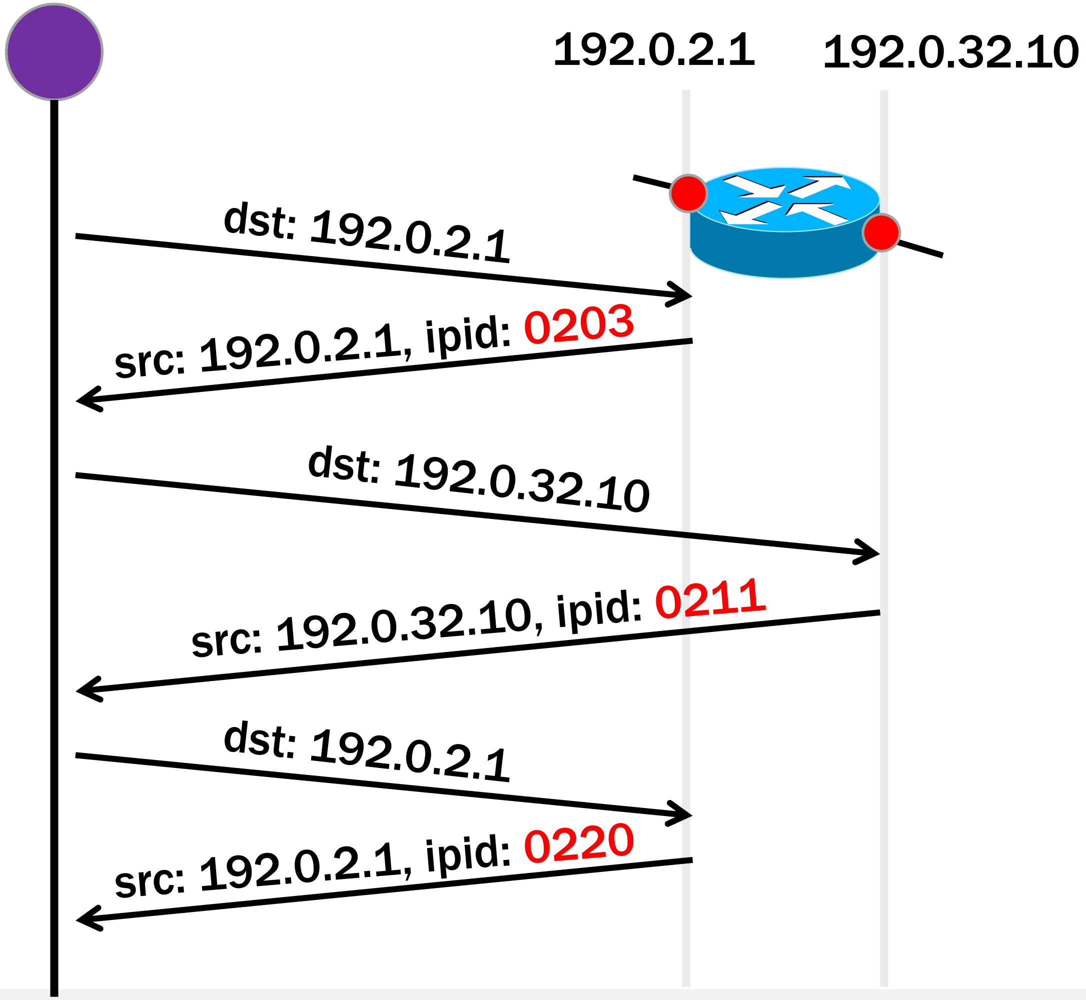
IPID ALIAS RESOLUTION



Idea: Infer that two addresses belong to the same router based on side channels, e.g., IP-ID

Assumption: router assigns IP-ID values from a single central counter.

IPID ALIAS RESOLUTION



Idea: Infer that two addresses belong to the same router based on side channels, e.g., IP-ID

Assumption: router assigns IP-ID values from a single central counter.

MIDAR

- Monotonic ID-based Alias Resolution
- Probes >2M addresses using a sliding-window, and multiple (more than 40) vantage points
- Highly precise

- K. Keys, Y. Hyun, M. Luckie, and k. claffy, **Internet-Scale IPv4 Alias Resolution with MIDAR**, IEEE Transactions on Networking, 21 (2), pp. 383-399, April 2013.

ITDK

- Internet Topology Data Kit
- Heavily annotated router-level graph, produced 1-2 times per year
- Alias Resolution: MIDAR; >2 million router interfaces
- Inference of Router Operator: bdrmapIT
- Inference of Location with
  - Hoiho to extract geohints in hostnames,
  - and from location databases.

OPTIONAL READING

- K. Keys, Y. Hyun, M. Luckie, and k. claffy, **Internet-Scale IPv4 Alias Resolution with MIDAR**, IEEE Transactions on Networking, 21 (2), pp. 383-399, April 2013.
- A. Marder, M. Luckie, A. Dhamdhere, B. Huffaker, J. Smith, and k. claffy, **Pushing the Boundaries with bdrmapIT: Mapping Router Ownership at Internet Scale**, ACM IMC, pp. 56-69, Boston, MA, USA, November 2018.
- M. Luckie, B. Huffaker, A. Marder, Z. Bischof, M. Fletcher, and k. claffy, **Learning to Extract Geographic Information from Internet Router Hostnames**, ACM CoNEXT, pp. 440-453, Virtual Event, Germany, December 2021.

LEARNING OBJECTIVES

1. Be able to describe how traceroute works
2. Appreciate how operators assign IP addresses to router interfaces
3. Be able to reason about router properties based on hostname annotations