Routing of Outgoing Packets for MP-TCP
draft-handley-mptcp-routing-00

Mark Handley

Costin Raiciu

Marcelo Bagnulo

Multiaddressed MP-TCP

- Host is connected to the Internet via more than one path.
  - Site where host resides is multihomed.
  - Host (eg phone) is multihomed.

- Host gets an IP address for each path it wishes to use.
  - IP addresses control incoming traffic via route advertisements, allowing load balancing.
  - By default, outgoing traffic would be routed based on destination. **Doesn't allow outgoing load balancing**.

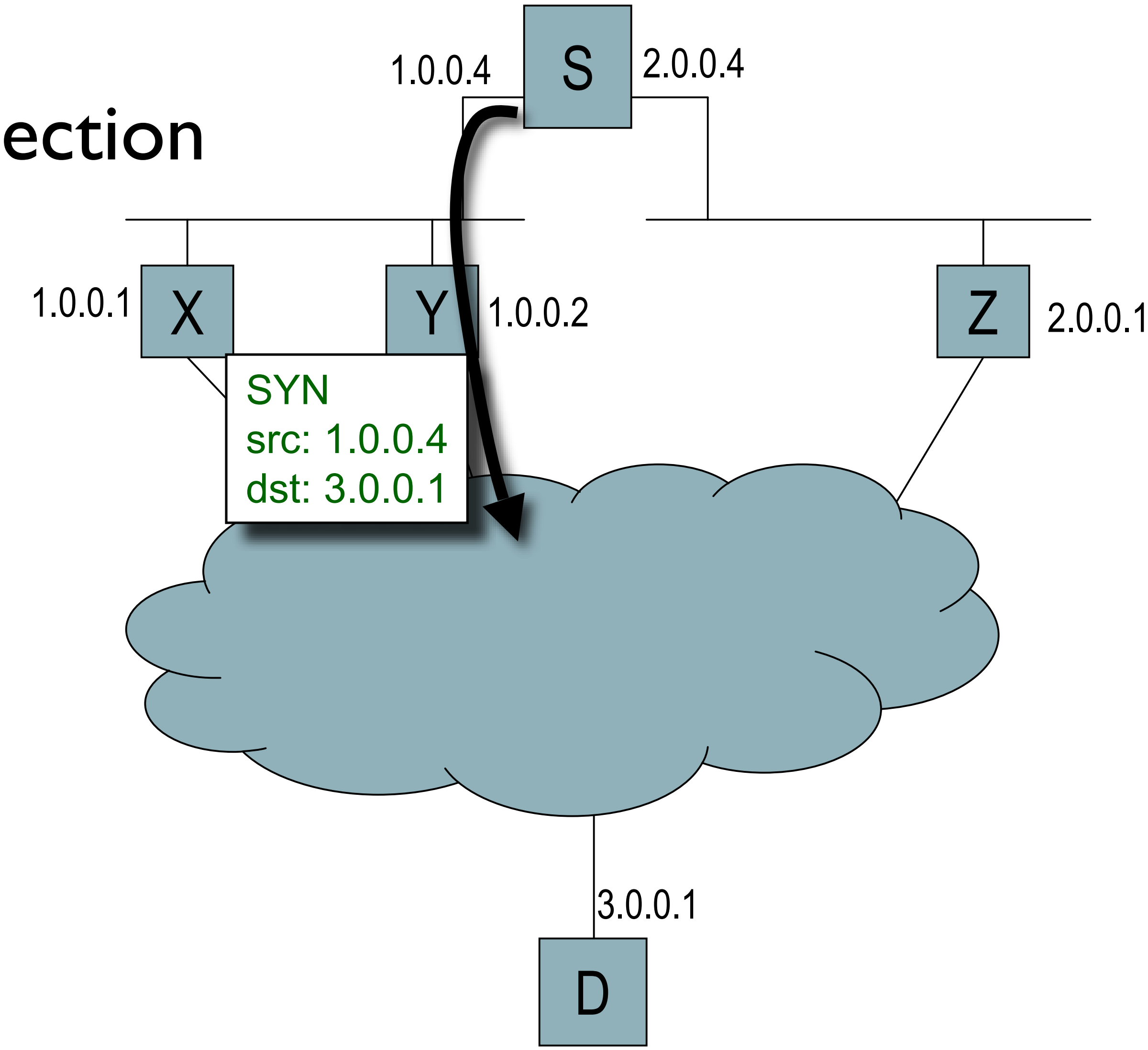
Example:
Outgoing Connection

New TCP connection **from S to D.**

In S's host routing table, longest prefix match for 3.0.0.1 is via 1.0.0.2.

TCP then binds the connection to 1.0.0.4.

Packets are routed via 1.0.0.2 - **no problem.**

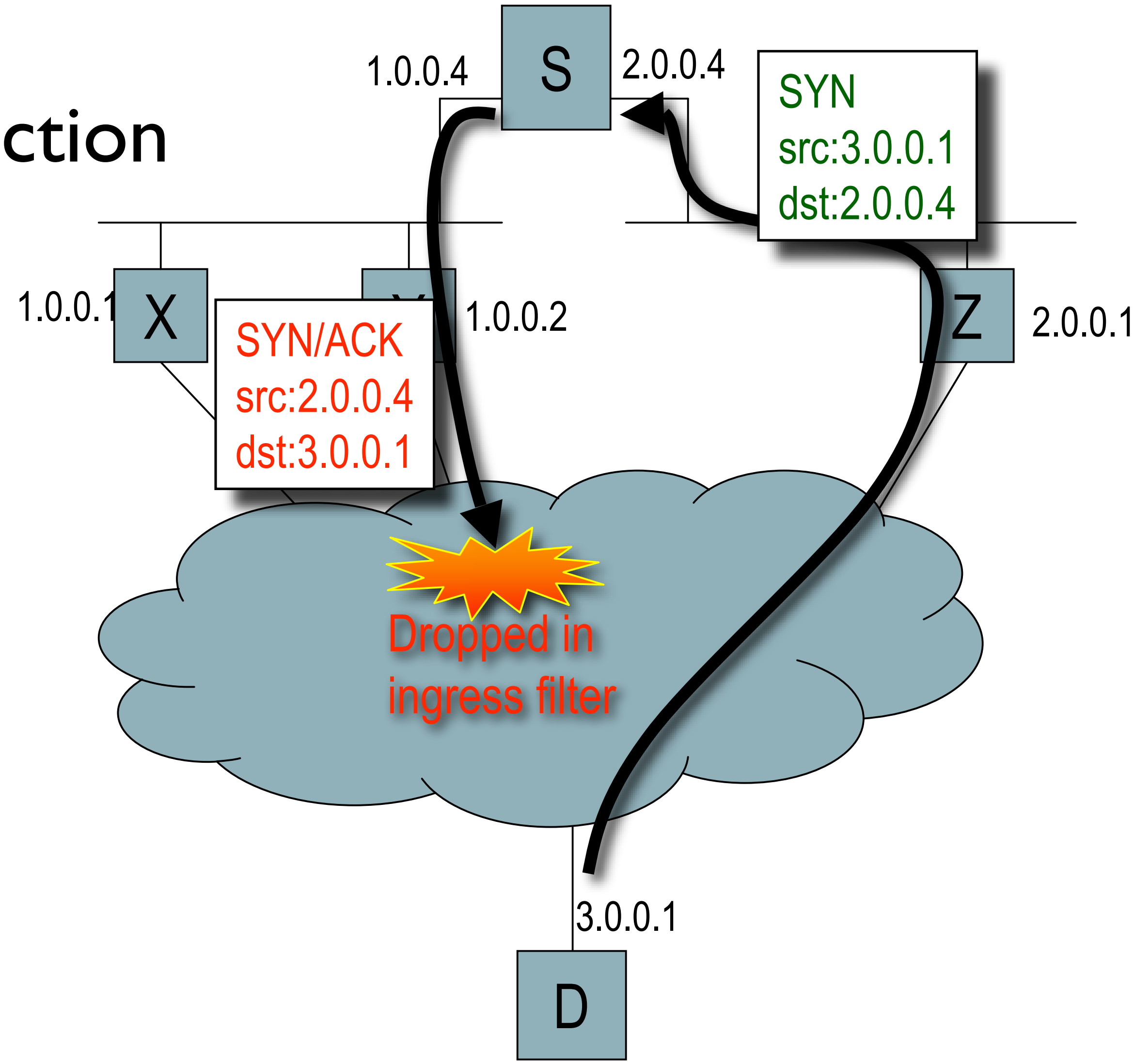


Example:
Incoming Connection

New TCP connection **from D to S.**

SYN sent to 2.0.0.4, so connection is bound to 2.0.0.4

In S's host routing table, longest prefix match for 3.0.0.1 is via 1.0.0.2. **Problem!**



Multi-addressing

- Because of the problems with incoming connections and ingress filtering, sites rarely configure addresses in this way.
- But we need multi-addressing for MP-TCP to work.
  - And an MP-TCP host has to fall back to regular TCP, so TCP needs to work too.
- Conclusion:
  - We need to revisit **host routing** to get most of the benefits of MP-TCP.

Traditional host routing

- Actually quite a wide range of different behaviours.
  - "strong" host vs "weak" host, etc.

- General idea:
  - OS has one best route to a particular prefix.
    - All packets to that destination are sent using this route.

MP-TCP Host Routing Prerequisites

- To use an outgoing subnet, a host must have a route to the destination via a next-hop router on that subnet.
- We do longest prefix match:
  - All routes actively used for subflows to the same destination must have the same prefix length.
- **Implication**:
  - To use multiple local addresses to the same destination address, there must be **multiple routes to the same prefix** via different next-hop routers.

New host forwarding rules

To send to a destination address from a source address:

1. Do **longest prefix match**.
   - This can give multiple routes with different metrics via different nexthop routers.
   - If no route exists, send fails.
2. If there are any routes via a next hop router on the same subnet as the source address:
   - Use the route via this subnet that has the lowest metric
3. Otherwise, send using the route with the lowest metric.
   - Even though it's via the wrong subnet.

Motivation

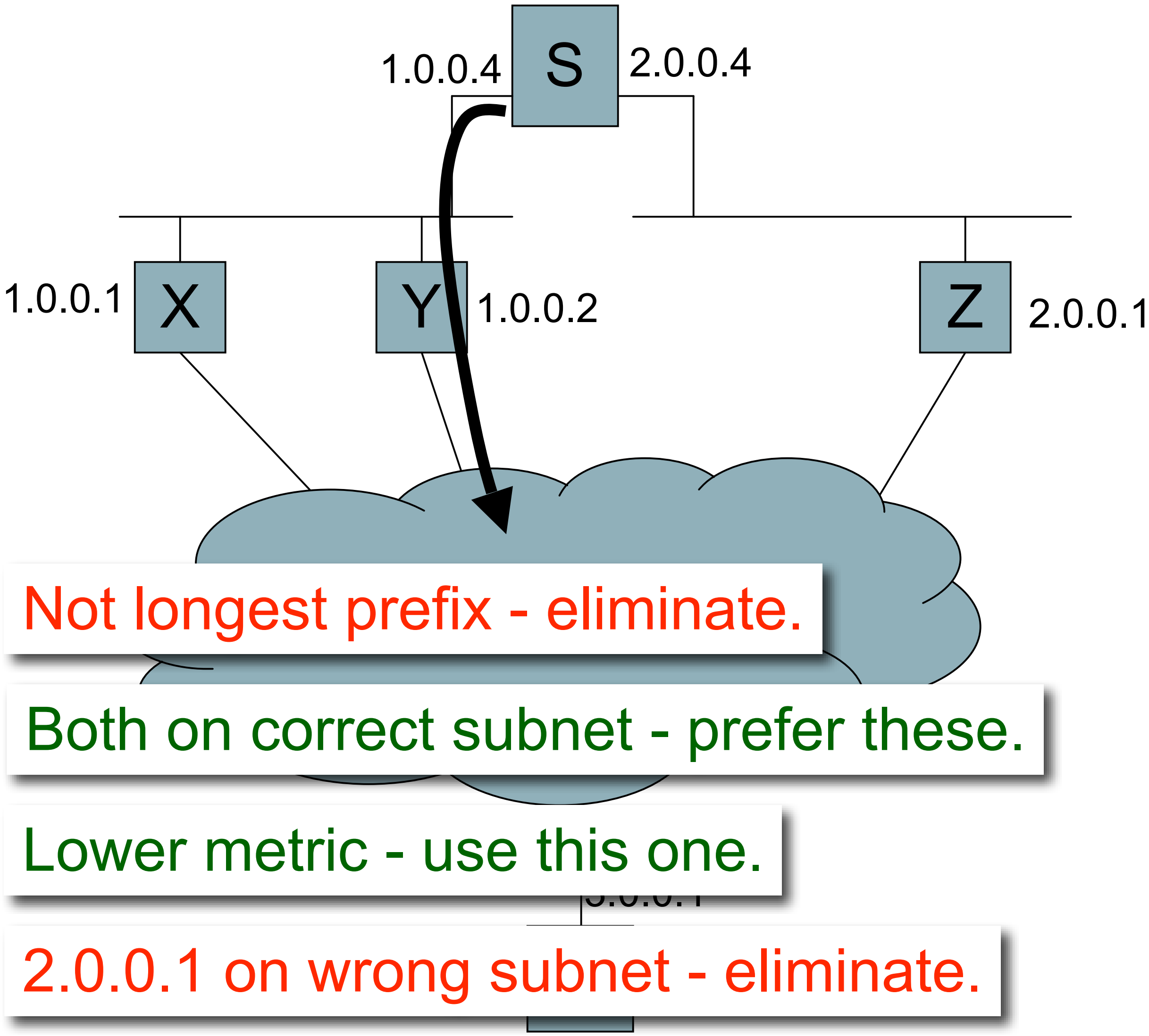
- We need to make outgoing routing match addressing to the extent it's possible
  - Even for regular TCP and UDP.
- For a multipath, we also need to force the use of multiple routes.
  - Normally only the lowest metric route would be used which gives no diversity.
- To achieve this we must override the route's metric in favour of the source address choosing the outgoing subnet.
  - But only where such a route exists.
  - If no such route exists, do the best we can.

Example 1: Active Opener

MPTCP packet from 1.0.0.4 to 3.0.0.1

Routes at S:

- ~~3.0.0.0/16 via 1.0.0.1 metric 1~~
- 3.0.0.0/24 via 1.0.0.1 metric 10
- 3.0.0.0/24 via 1.0.0.2 metric 5
- ~~3.0.0.0/24 via 2.0.0.1 metric 2~~



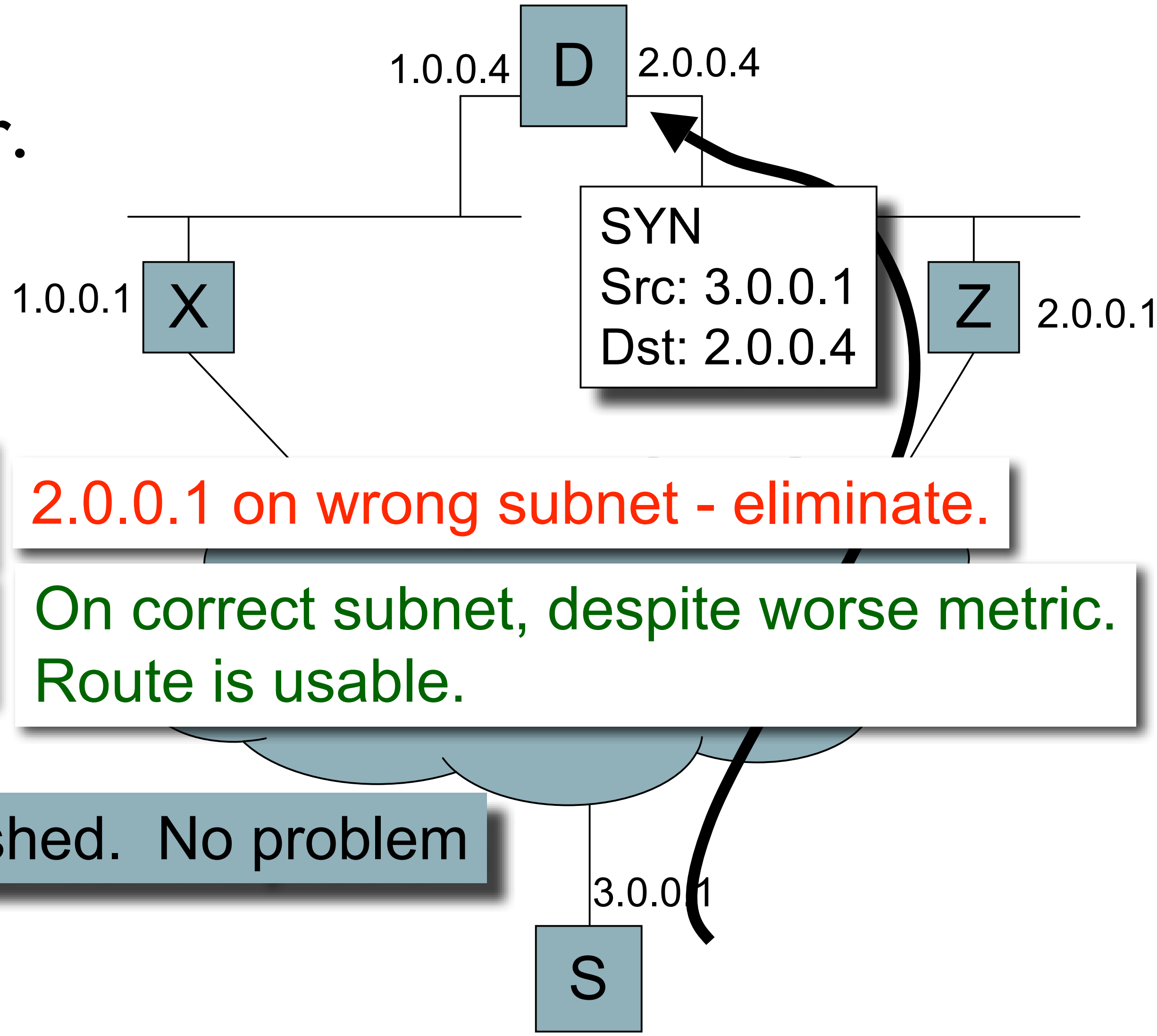
Not longest prefix - eliminate.

Both on correct subnet - prefer these.

Lower metric - use this one.

2.0.0.1 on wrong subnet - eliminate.

Example 2:
Passive Listener.



Routes at D:

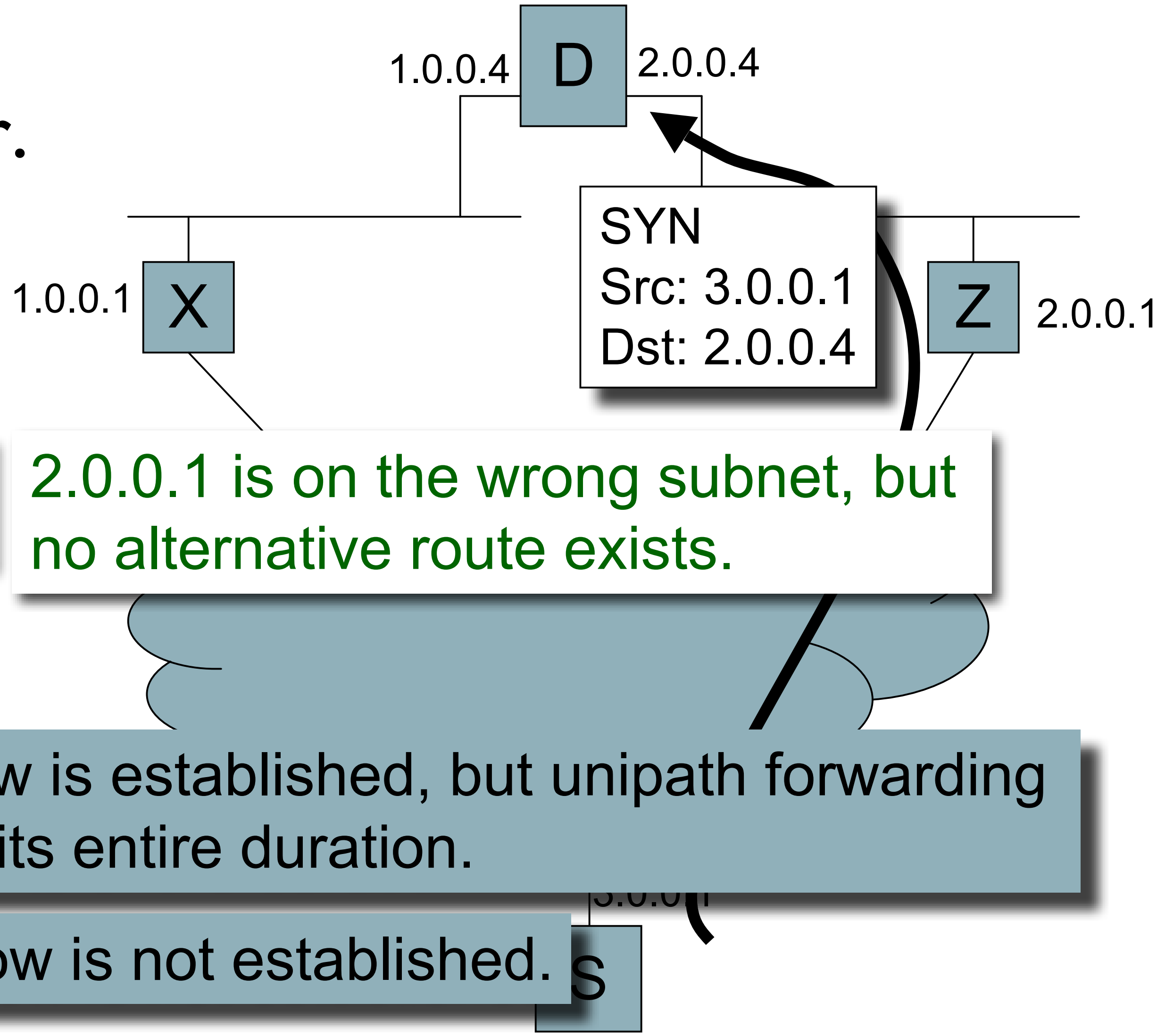
- 3.0.0.0/24 via 1.0.0.1 metric 1
- 3.0.0.0/24 via 2.0.0.1 metric 10

2.0.0.1 on wrong subnet - eliminate.

On correct subnet, despite worse metric. Route is usable.

Subflow is established. No problem

Example 3:
Passive Listener.



Routes at D:

- 3.0.0.0/24 via 1.0.0.1 metric 1

2.0.0.1 is on the wrong subnet, but no alternative route exists.

Weak host: subflow is established, but unipath forwarding rules are used for its entire duration.

Strong host: subflow is not established.

Usage examples.

1. Multi-interface host, directly connected to two (or more) ISPs.
   - Eg. smartphone.

2. Single-interface host at multi-homed site.
   - Eg. web server.

Multi-interface host.

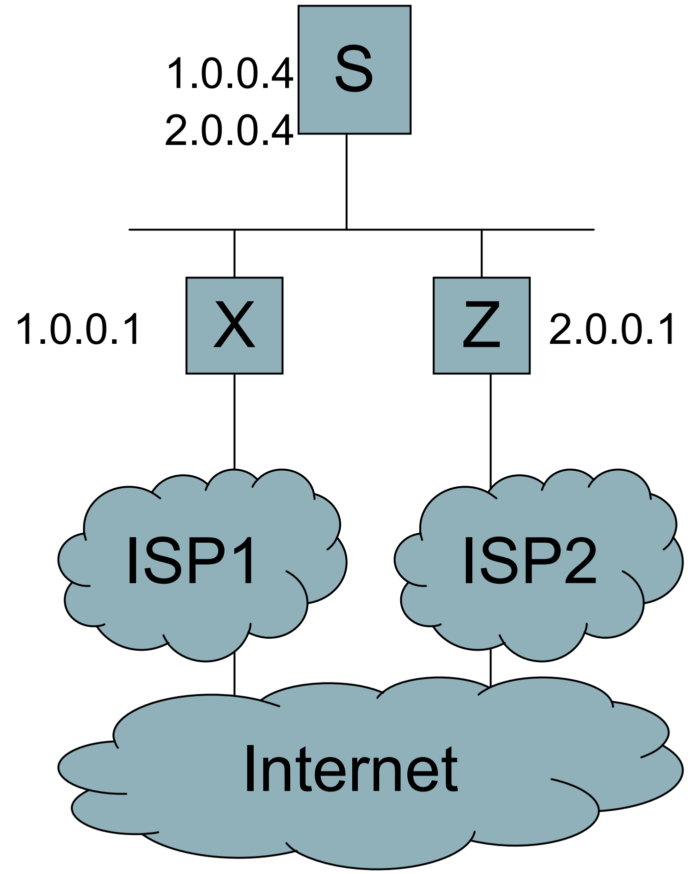
- Directly connected to ISPs.
- Has complete control over which packet leaves via which link.
  - Host multipath forwarding rules are sufficient.

Single-interface host at multihomed site.

- Site has one address prefix per provider.
- Host gets one address from each prefix.

Multihoming: Case 1

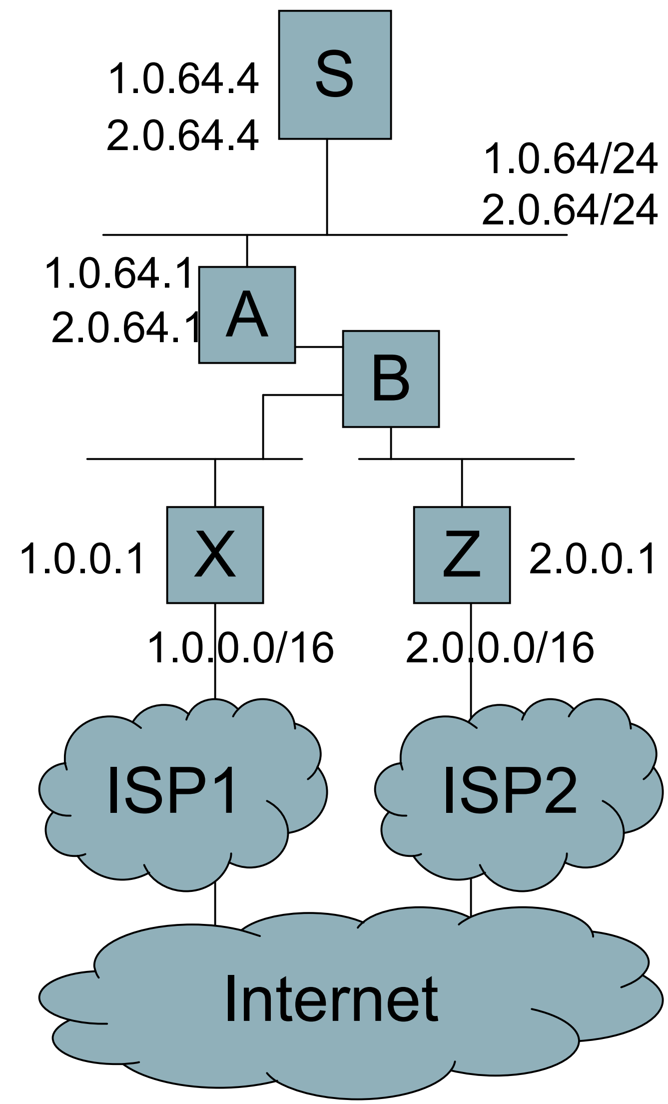
- Multihomed host is on the same L2 infrastructure as site exit routers.
  - Common in datacenters.
- Host multipath forwarding rules are sufficient.



Multihoming: Case 2

- Multihomed host is several IP hops from site exit routers.
  - E.g, UCL, organizations with lots of internal structure.

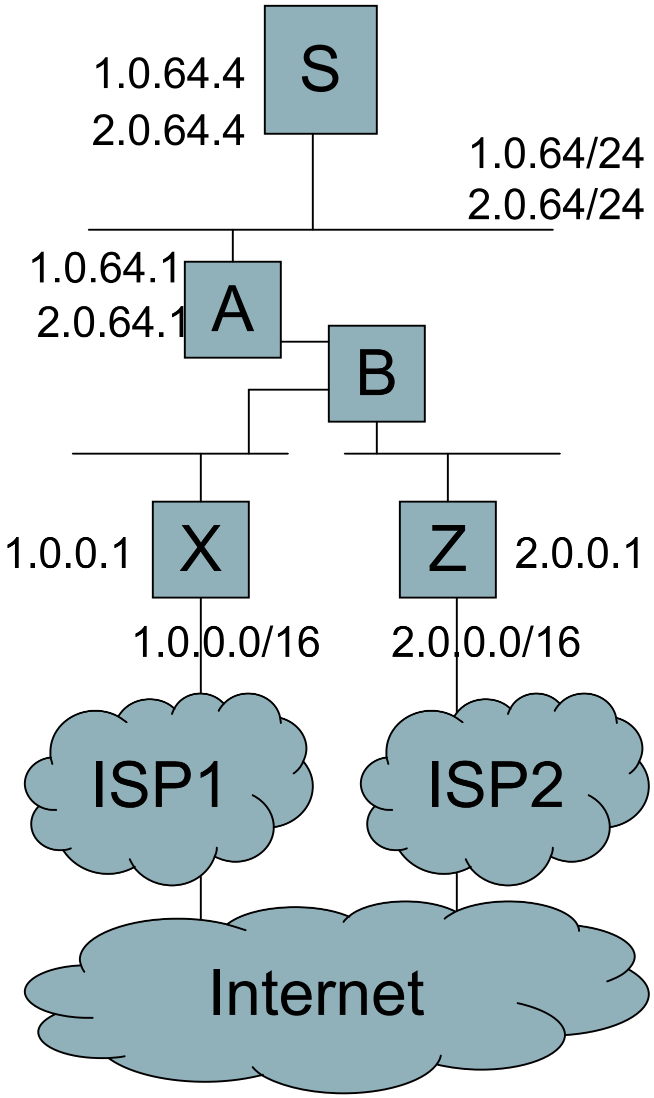
- Host multipath forwarding rules will allow multiple subflows to be set up, but host cannot ensure routing congruence.



Multihoming: Case 2

Many possible solutions:

- Tunnel from S to X and Z.
- Source-address routing.
  - In this case, at B.
- MPLS from S.
- Virtual routers on A, then MPLS to X, Y.
- Loose-source-route from S via X or Z.



Summary

- Important to specify how MP-TCP interacts with host routing.
  - New host forwarding rules cover what seem to be the most common cases for MP-TCP.
- Additional network mechanisms needed for full generality.
  - Existing mechanisms seem to suffice.
  - Not clear there's a need to standardize these, or to choose just one mechanism.

Extra slides

What about route changes?

- For a directly connected interface.
  - If the interface goes down, the address is removed.
  - Subflows using that interface are paused (killed?).
- Only on hosts using a dynamic routing protocol can routes disappear.
  - Might then switch to an incongruent path.
  - Is this a problem?
    - Worst case is that subflow stalls due to NAT or ingress filtering?
    - Same problem with current forwarding rules.