Network Protocols

Routing

IP routing

- Performed by routers
- Table (information base) driven
- Forwarding decision on a hop-by-hop basis
- Route determined by destination IP address

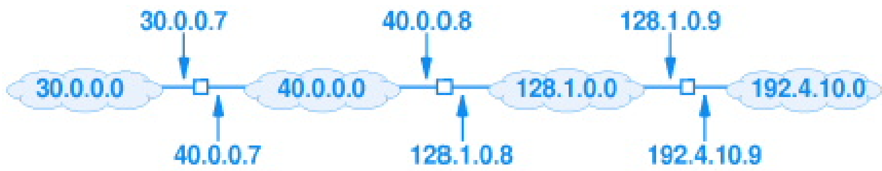
Basic IP forwarding process

- For an IP datagram received on an interface
- Remove layer 2 information
- Extract destination IP address *(D)*
- Find best match for *(D)* in the routing table
- Extract fowarding address *(F)* for next hop
- Create layer 2 information
- Send datagram to *(F)*

IP routing tables

Since each entry in a routing table represents an IP network, the size of the routing table is proportional to the number of IP networks known throughout the entire internetwork.

IP routing table illustrated



(a)

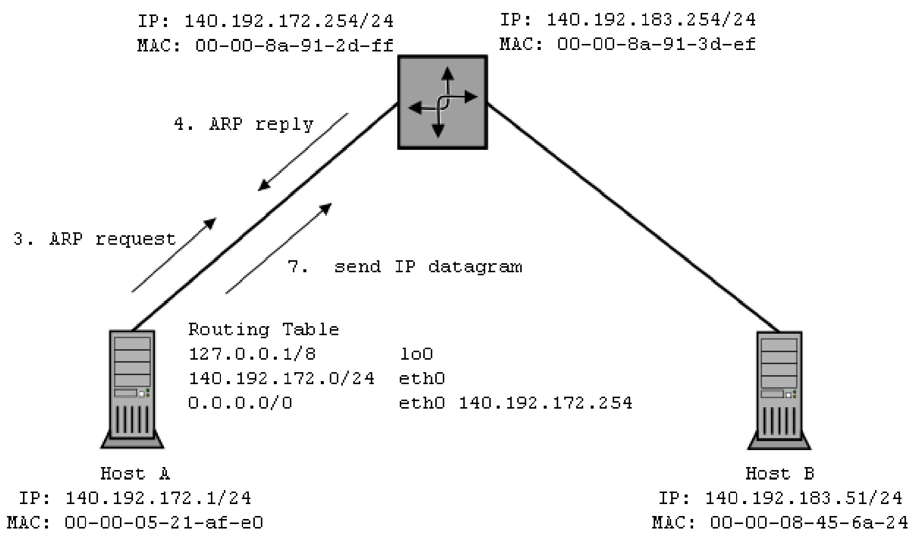
Destination	Mask	Next Hop
30.0.0.0	255.0.0.0	40.0.0.7
40.0.0.0	255.0.0.0	deliver direct
128.1.0.0	255.255.0.0	deliver direct
192.4.10.0	255.255.255.0	128.1.0.9

(b)

Generating routing tables

- Manually
  - Simple for small, single path networks
  - Does not scale well
  - Useful for permanent route entries
- Dynamically
  - Allows quick re-routing around failed nodes/links
  - Useful for large multi-path networks
  - Catastrophic, distributed failures are possible

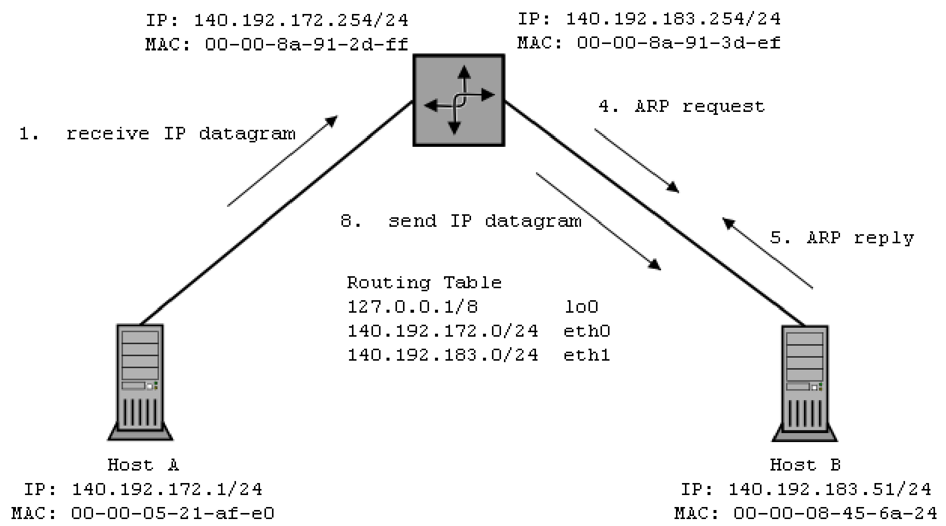
IP routing illustrated



Host A sends an IP datagram to Host B

1. Host A examines destination IP network
2. No match in its routing table, chooses default gateway
3. ARPs for default gateway's MAC address
4. Receives ARP reply from 00-00-8a-91-2d-ff
5. Builds layer 2 frame with destination to 00-00-8a-91-2d-ff
6. Adds IP info with destination IP of 140.192.183.51
7. Sends datagram on local link

IP routing illustrated (continued)



Host A sends an IP datagram to Host B

1. Router receives frame with IP datagram inside
2. Examines layer 3 destination address/network
3. Matches destination network to attached link's network
4. ARPs for destination 140.192.183.51 on local network link
5. Receives ARP reply from 00-00-08-45-6a-24
6. Builds layer 2 frame with destination 00-00-08-45-6a-24
7. Adds IP info with destination IP of 140.192.183.51
8. Sends datagram on local network link

Routing metrics

- Shortest/longest hop path
- Lowest/highest cost path
- Lowest/highest reliability
- Best/worst latency
- Policy decisions

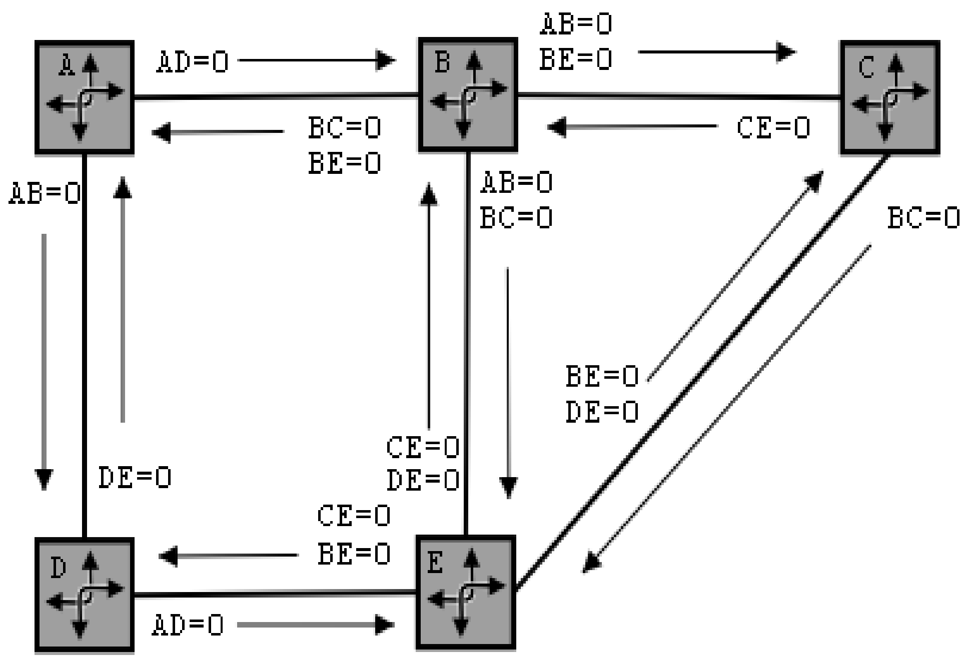
Some terminology

- Autonomous system (AS)
  - A network or set of networks that is administrated by a single entity
- Interior gateway protocols (IGP)
  - Routing protocol used within an AS
- Exterior gateway protocols (EGP)
  - Routing protocol used between ASes

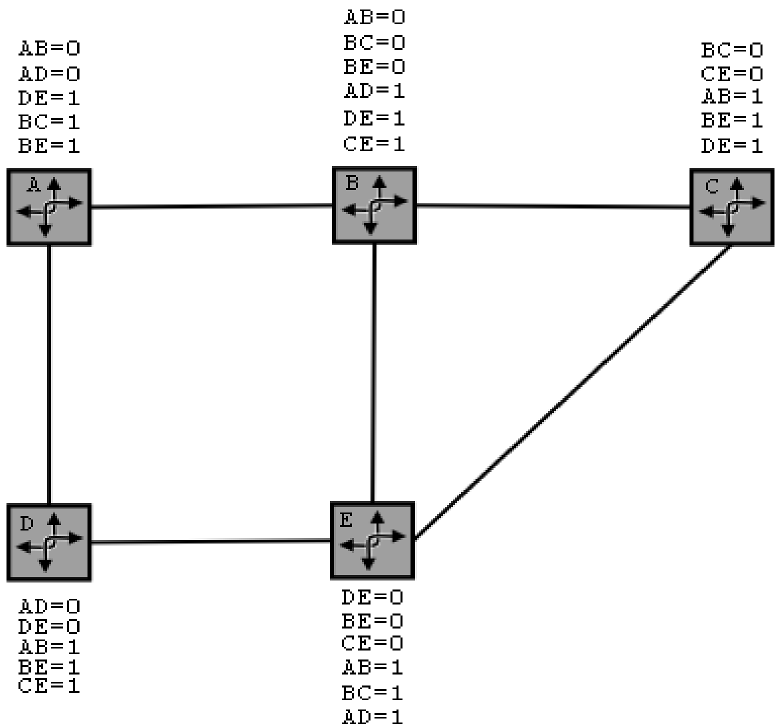
Distance vector routing

- Each node maintains distance to destination
  - e.g. 4 hops to network XYZ, 2 hops to ABC
- Periodically advertise attached networks out each link
- Learn from other router advertisements
- Advertise learned routes
- Also known as Bellman-Ford after inventors of the algorithm

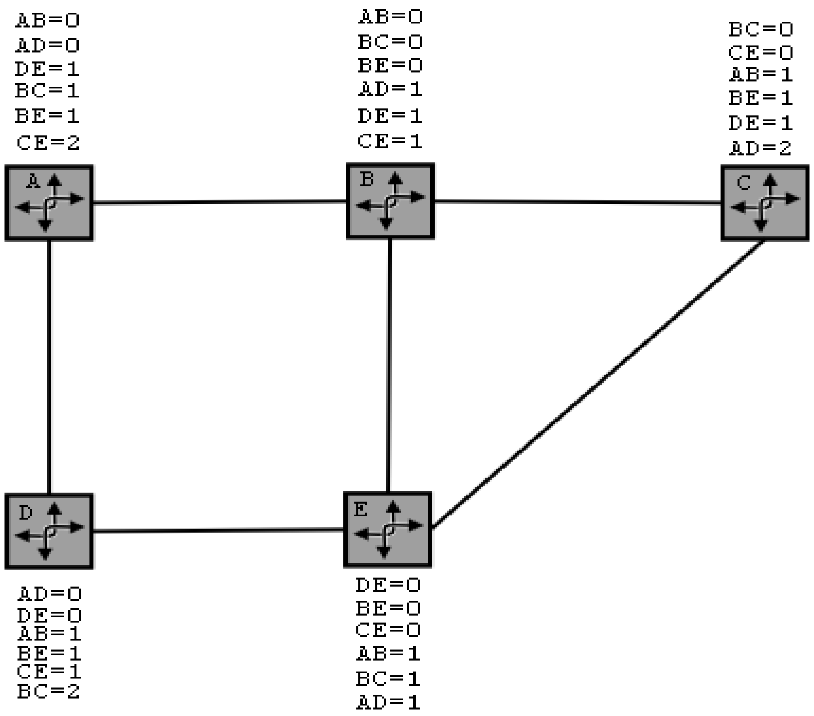
Distance vector illustrated



Distance vector illustrated (continued)

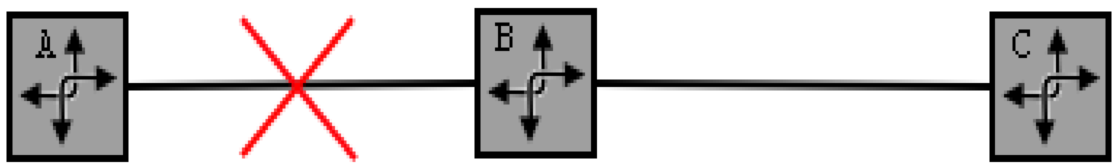


Distance vector illustrated (converged)



Problems with distance vector

- Convergence time can be slow
- Also known as the *count to infinity* problem
- What happens when link to A fails?



Solving count to infinity

- Hold down
  - Advertise infinity for route and wait a period of time before switching routes. Hope that news of the downed link will spread fast enough. Kludge.
- Report the entire path
  - Guarantees no loops, but expensive.
- Split horizon
  - Do not advertise route to a neighbor if you received route from that neighbor. Not foolproof.

Other distance vector improvements

- Triggered updates
  - Advertise changes immediately. May cause *route flapping*, but generally a good thing to do.
- Poison reverse
  - Used with split horizon, advertise infinity rather than nothing at all.
- DUAL
  - Somewhat like hold down. Can switch paths if new distance is lower. Sufficiently complex.

Routing information protocol (RIP)

- Standardized in RFC 1058 and 2453
  - The later defines RIPv2 for improvements
- Very simple
- Slow convergence time
- UDP broadcast every 30 seconds (default)
- Route times out after 180 seconds (default)
- Widely used as an IGP (RIPv2 particulary)
- 15 hop limit (any greater equals infinity)

RIP version 2 (RIPv2)

- Most important new feature was to include the subnet mask with the advertised route
  - Needed to support classless addressing
- Support for authentication
- Uses IP multicast destination address
- Route tag option
  - For interaction with external gateway protocols
- Next-hop option
  - Next-hop router associated with advertisement

RIPv1 packet format

Packet format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  command (1)  |  version (1)  |       must be zero (2)        |
+---------------+---------------+-------------------------------+
|                                                               |
~                         RIP Entry (20)                        ~
|                                                               |
+---------------+---------------+---------------+---------------+
```

A RIPv1 entry has the following format:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| address family identifier (2) |      must be zero (2)         |
+-------------------------------+-------------------------------+
|                         IPv4 address (4)                      |
+---------------------------------------------------------------+
|                         must be zero (4)                      |
+---------------------------------------------------------------+
```

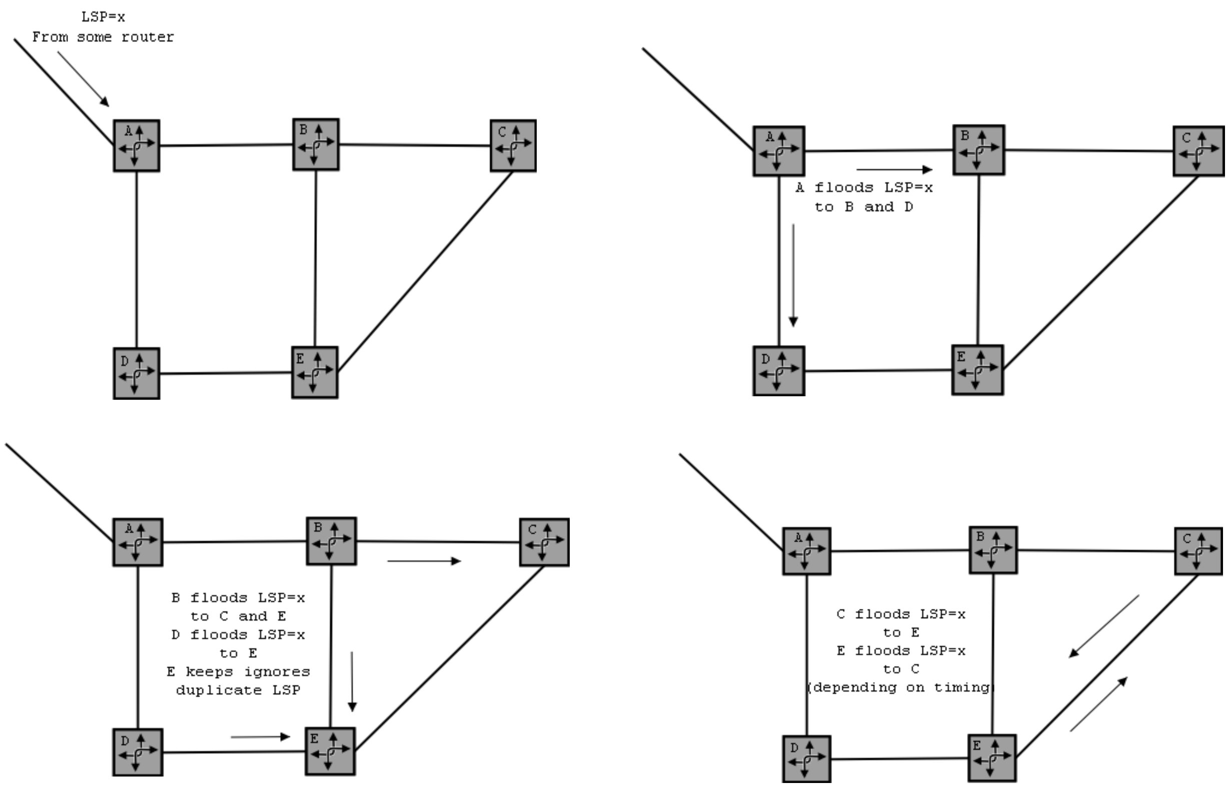
RIPv2 packet format

```
Packet format is the same, RIPv2 entry format is:

0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Address Family Identifier (2) |        Route Tag (2)          |
+-------------------------------+-------------------------------+
|                         IP Address (4)                        |
+---------------------------------------------------------------+
|                         Subnet Mask (4)                       |
+---------------------------------------------------------------+
|                         Next Hop (4)                          |
+---------------------------------------------------------------+
|                         Metric (4)                            |
+---------------------------------------------------------------+

Authentication uses one entry of the format:

0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Command (1)   | Version (1)   |            unused             |
+---------------+---------------+-------------------------------+
```

Link state routing

- All routers have complete network topology information (database) within their *area*
  - Link state packets are flooded to all area routers
- Each router computes its own optimal path to a destination network
- Convergence time is very short
- Protocol complexity is high
- Ensures a loop free environment

Link state routing illustrated



Link state routing databases

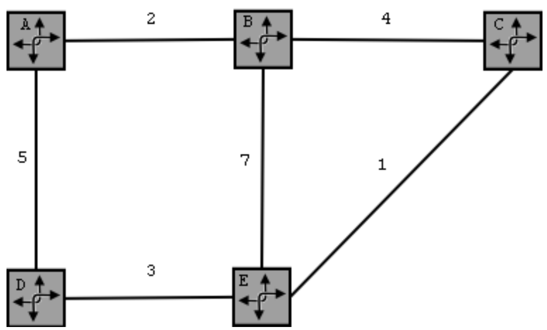
- Link state database
  - Contains latest link state packet from each router
- PATH (permanent) database
  - Contains (router ID/path cost/forwarding direction) triples
- TENT (tentative) database
  - Same structure as PATH, its entries may be candidates to move into PATH
- Forwarding database
  - Contains ID and forwarding direction

Dijkstra's algorithm

- Start with self as root of the tree
  - (my ID, path cost 0, forwarding direction 0) in PATH
- For each node in PATH, examine its LSP and place those neighbors in TENT if not already in PATH or TENT
- If TENT is empty, exit, otherwise find ID with lowest path cost and in TENT and move it to PATH

Dijkstra's algorithm illustrated

1. Start with A, put A in PATH, examine A's LSP, add B and D to TENT
2. B is lowest path cost in TENT, place B in PATH, examine B's LSP, put C,E in TENT
3. D is lowest path cost in TENT, place D in PATH, examine D's LSP, found better E path
4. C is lowest path cost in TENT, place C in PATH, exame C's LSP, found better E path again
5. E is lowest path cost in TENT, place E in PATH, examine E's LSP (no better paths)
6. TENT is empty, terminate



Open shortest path first (OSPF)

- Standardized as RFC 2328 (OSPFv2)
- Complex
- Supports multiple routing metrics (though rarely used)
- Allows 2 tier hierarchy for scalability
- Efficient
- Good convergence properties
- Runs directly over IP
- Recommended IGP by the IETF

OSPF packets

- Hello
  - Link maintenance
- Exchange
  - Initial exchange of routing tables
- Flooding
  - Incremental routing updates

OSPF database records

- Router links
  - Summarizes links from advertising router
- Network links
  - Transit networks (broadcast and non-broadcast)
- Summary links
  - Summary info advertised by area border routers
- External links
  - Imported routers, typically from a EGP

Common OSPF header

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Version #   |     Type      |         Packet length         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Router ID                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Area ID                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Checksum            |             AuType            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Authentication                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Authentication                          |
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
```

Interdomain routing

- Routing domains are independently funded
- Routing domains do not trust each other
- Routing domains may have different policies
- Static routing
- EGP - first interdomain routing protocol
- BGP - current path vector routing protocol

Border gateway protocol (BGP)

- Current version 4 standardized in RFC 1771
- Runs over TCP
- Sequence of AS numbers comprises path
- Select route based on preferences of path(s)
- Can edit path in route advertisements
- Can selectively advertise paths/routes
- E-BGP versus I-BGP

BGP attributes

- Describe routes in BGP updates
- Confusing descriptions
  - e.g. Well known attributes must be supported
  - e.g. Mandatory must be present in the update
- Examples
  - AS path
  - Community
  - Unreachable

Confederations

- Group of ASs that appear as a single AS
- A form of aggregation
- May simplify routing policies
  - e.g. Don't go through confederation X rather than specifying each AS in the confederation
- Sub-optimal routing may result
  - Multiple ASs in a path vector appear as a loop

Message types

- Open
  - First message when neighbors come up
- Update
  - Contains routing information
- Notification
  - Final message just before link is disconnected
- Keepalive
  - Reassures reachability in absence of updates

Route dampening

- Routes that oscillate ripple through Internet
  - Consumes CPU and causes instability
- Unstable (flapping) routes are penalized
  - For some period of time route is surpressed
  - Supression time can increase to a maximum
  - Supression of routes results in lost connectivity
- Bigger/important netblocks dampen slowly

Sample Cisco BGP configuration

```
Router bgp 12345
 bgp log-neighbor-changes
 network 128.160.0.0 mask 255.255.0.0
 neighbor 36.5.1.1 remote-as 54321
 neighbor 36.5.1.1 description E-BGP peer with XYZ corp.
 neighbor 36.5.1.1 password as54321password
 neighbor 36.5.1.1 version 4
 neighbor 36.5.1.1 prefix-list invalid in
 neighbor 36.5.1.1 prefix-list announce out

ip prefix-list invalid seq 10 deny 0.0.0.0/8 le 32
ip prefix-list invalid seq 20 deny 10.0.0.0/8 le 32
ip prefix-list invalid seq 30 deny 127.0.0.0/8 le 32
...

ip prefix-list announce seq 10 permit 128.160.0.0/16
ip prefix-list announce seq 20 deny 0.0.0.0/0 le 32
```

Final thoughts

- Routing protocols work fine 99.99% of the time, but when they don't, failures are generally catastrophic
- Troubleshooting complex routing problems can make your brain hurt
- Generally the only necessary intelligence that is required *in the network* is IP routing
- Internet peering is a fun issue to explore