Routing Overview
One of two critical systems

Routing (BGP) and naming (DNS) are, by far, the two most critical subsystems of the Internet infrastructure. In the case of BGP, participation in and access to the routing system itself is generally, or rather should be, limited to a subset of trustworthy nodes and admins.
Do all IP hosts route? Yes.

Most hosts make one of three routing decisions:
1) send packet to another via a relay
2) send packet to itself
3) send packet to a directly attached neighbor
Simplified routing decision tree

1. Originating Host
   - Send to local protocol module
   - To 127/8 or equivalent locally configured interface address
     - Yes
       - Do ARP if necessary, send out local interface
     - No
       - To locally attached network?
         - Yes
           - Do ARP if necessary, send out local interface
         - No
           - Do ARP if necessary, send to designated router
Your end host != router

- Need to know your address, network and gateway
  - not so much a “routing system” process
  - this is your host's bootstrap challenge
- We don't tend to think of end hosts as routers
- How do they differ then?
  - network / interface attachments
  - distributed routing algorithms
  - forwarding packets on another's behalf
Real routers work more like this

Receive Datagram

Remove any L2, validate checksum and TTL, apply BCP38/BCP84/filters.

Fail

Log, return error, drop.

Pass

Find next hop.

Local

Do ARP if necessary, apply any L2 headers, decrement TTL, recompute checksum, apply QoS/CoS policies, send directly to end host.

Forward

Do ARP if necessary, apply any L2 headers, decrement TTL, recompute checksum, apply QoS/CoS policies, send to relay router.
Best match forwarding

- Forward packet via the “most specific” route
- Most specific to least specific (IPv4 example):
  - host (/32) route, /31, /30, /29, ... default (/0)
- If no route, drop and return ICMP error to source
Routers as signposts
How do routers build a signpost?

• Maybe manually configured, but that doesn't scale
• Routers gossip amongst themselves
• Well defined “gossip” protocols are used
  • e.g. RIP, EIGRP, OSPF, IS-IS, BGP
  • a bootstrap configuration is generally required
• Reachability information associated with all routes
  • e.g. distance, cost, preference, policy
Key IPv4 field for routing: TTL

• More apt name today would be hop count
  • in fact, that is just what it is called in IPv6 now
• This field prevents packets looping forever
• Other uses are secondary to this
  • traceroute
  • source OS fingerprint and distance detection
  • BGP peering hack (aka GTSM, RFC 3682)
Key IP field for routing: Destination Address

- Consists of both a...
  - host/interface identifier (usually unique) and
  - a network identifier (also usually unique)
- Combined, the daddr helps hosts and routers
  - get the packet to the correct network
  - and to the specific host on the correct network
BGP Overview

• The routing protocol for connecting *domains*
• Besides the *network prefix* the *path* is the key component of a BGP route
• Autonomous system numbers (ASNs) define path
  • generally an ASN == domain
    • NOTE: this is not a reference to DNS!
• Even if you don't use it for actual Internet routing, it might be handy for other things (e.g Team Cymru bogon route server, IP addr to ASN mapping)
IS-IS/OSPF Overview

- Widely used \textit{intradomain} routing protocols
- \textit{Link state} database of entire routed network built by all routers
- Each router can make an optimal forwarding decision, because it has a complete view of all the routers and their attached networks
- Relatively simple idea, but is a bit more complex to implement – i.e. database synchronization issues
A real Internet BGP route entry

route-views.oregon-ix.net>sh ip bgp 68.22.187.0/24
BGP routing table entry for 68.22.187.0/24, version 543323
Paths: (34 available, best #7, table Default-IP-Routing-Table)
   Not advertised to any peer
     8075 2828 23028
       207.46.32.34 from 207.46.32.34 (207.46.32.34)
         Origin IGP, localpref 100, valid, external
     3333 3356 2828 23028
       193.0.0.56 from 193.0.0.56 (193.0.0.56)
         Origin IGP, localpref 100, valid, external
     4513 13789 3561 23028 23028 23028 23028
       209.10.12.125 from 209.10.12.125 (209.10.12.125)
         Origin IGP, metric 4103, localpref 100, valid, external
An example routing table

route-views.oregon-ix.net>show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF
IA - OSPF inter area, N1 - OSPF NSSA external type 1
N2 - OSPF NSSA external type 2, E1 - OSPF external type 1
E2 - OSPF external type 2, E - EGP i - IS-IS
su - IS-IS summary, L1 - IS-IS level-1
L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route

Gateway of last resort is 128.223.51.1 to network 0.0.0.0

B  216.221.5.0/24 [20/489] via 208.51.134.254, 18:06:49
B  210.51.225.0/24 [20/0] via 12.0.1.63, 18:07:52
B  210.17.195.0/24 [20/0] via 216.218.252.164, 18:08:11
B  209.136.89.0/24 [20/0] via 216.218.252.164, 18:08:21
B  209.34.243.0/24 [20/0] via 157.130.10.233, 17:59:49
B  205.204.1.0/24 [20/0] via 157.130.10.233, 18:00:57
B  204.255.51.0/24 [20/0] via 157.130.10.233, 17:59:44
B  204.238.34.0/24 [20/0] via 157.130.10.233, 18:00:28
Want router access?

- Telnet to route-views.routeviews.org
- Browse to http://routerproxy.grnoc.iu.edu/
- Go easy, don't ruin it for the rest of us please
  - notwithstanding potential bugs or attacks, by default access it intended to be limited (sorry, no “enable”), but they can still be very helpful for remote analysis and troubleshooting
You do have enable, kind of

• On Unix, Linux, Mac OS X
  • netstat -arn

• On Microsoft Windows
  • route print
There is router security and there is route security

- Few serious network engineers use HTTP
  - “That's probably a good thing!” you say
- Many Cisco networks still use Telnet
  - this is where you security people go “WTF!?!?”
- Many networks have SNMPv1 write enabled
  - then you go “OMFG!?!?”
- Almost nobody watches out for more specifics
  - “Specifics smurifics, whoop-dee $#!&@”
Au contraire

• Router security
  • authentication, filtering, crypto... DONE!
  • uhm, no

• Route security
  • this is the old, “my security, depends on your ability to do security” problem
  • say you have and announce a /16
  • someone announces /24's in that /16.
  • uh-oh
Examine your own router

- Microsoft Windows
  - `ipconfig /all`
  - `route print`
- UNIX (varies depending on flavor)
  - `ifconfig`
  - `netstat -arn (or route -n)`
  - `cat /etc/dhcp*/dhclient.conf (or something like it)`
  - `cat /etc/resolv.conf`
- Mac OS X
  - like UNIX, but also check Sys prefs → Network
Determine your...

- IPv4/IPv6 address(es). More than one?
- Net mask – in dotted decimal (IPv4) and / notation
- ARP cache
- Default route, default router
- Network interface list
- Recursive/caching name servers
- MAC address and OUI assignment
Recall...

Routers as signposts...
Best match forwarding...
Hierarchical routing
Route announcements

Send to me for 192.0.2.0/24

Default to me 0.0.0.0/0

192.0.2.1/32
192.0.2.2/32 over here
Distance Vector (DV) Routing

- Algorithm known as Bellman-Ford
- Routers gossip amongst themselves
  - kind of like the “telephone game”
- As announcement propagates, distance increases
DV bootstrap 1
DV bootstrap 2

a

ab=0  ad=0  ab=0  bc=0  bc=0  ce=0
bc=1  be=1  be=0  ad=1  ad=1  be=1
de=1  ce=1  de=1  de=1  de=1

da

ad=0  de=0  be=0  ce=0
ab=1  be=1  de=0  ab=1
ce=1  de=1  ad=1  bc=1
c
DV converged

\[
\begin{align*}
ab &= 0 \quad ad &= 0 \\
bc &= 1 \quad be &= 1 \\
de &= 1 \quad ce &= 2 \\
\end{align*}
\]

\[
\begin{align*}
ab &= 0 \quad bc &= 0 \\
be &= 0 \quad ad &= 1 \\
ce &= 1 \quad de &= 1 \\
\end{align*}
\]

\[
\begin{align*}
bc &= 0 \quad ce &= 0 \\
ab &= 1 \quad be &= 1 \\
de &= 1 \quad ad &= 2 \\
\end{align*}
\]
DV Summary

• Simple distance calculation determines path
• Periodic route updates sent to neighbors
• Convergence time can be slow
• Updates can be “triggered” upon a metric change
• Loop avoidance optimizations delay convergence
• Examples:
  • RIP (IP/IPX), RIPv2, RIPng, IGRP (cisco)
Link-state (LS) routing

- Algorithm known as Dijkstra
- Routers exchange their connectivity in a LSP
  - LSP = link state packet
  - as opposed to exchanging the routing table
- A LSP includes:
  - router id
  - sequence number
  - links and costs for each link
  - time-to-live (TTL)
LS bootstrap

NOTE: sequence of events an example only

1) a floods LSP_01 to b and d
2) b floods LSP_01 to c and e
2) d floods LSP_01 to e, e ignores duplicate LSP
3) c or e flood LSP_01 to the other, whoever is faster
LS summary

• Each router builds their own map from LSPs
• Good convergence time
• Good loop avoidance
• Can be more complex and resource intensive
  • not really an issue these days in practice
• Generally preferable over distance vector
• Examples:
  • OSPF, IS-IS
Path vector (PV) routing

- More similar to DV than LS
  - Like DV, routes are exchanged, not connectivity
- Each hop appends itself to the path of a route
- A “hop” is an Autonomous System (AS)
  - An AS is roughly an Autonomous ISP/network
- Path policy & preferences influence route selection
  - I.e. paths can be altered, routes can be rejected
- Examples: BGP
Influencing BGP route selection

- Most specific prefix (best match) still matters most
- Highest local preference value associated w/ route
- Shortest AS path length (a weak form of distance)
- Origin type (e.g. IGP versus EGP)
- Lowest MED, a peer announced parameter
- Prefer external over internal route
- ...first received route, lowest router id, etc.
- Use of community strings (policy) changes things
IPv4 BGP Topology Example

AS 65000
192.0.0.0/8
10.0.0.0/8
172.16.0.0/8
...

MED = - 10

AS 65100
169.254.0.0/16
1.0.0.0/8
221.0.0.0/8
...

PREPEND++

AS 65020
172.16.32.0/20

PREF =+ 100

AS 65010
192.0.2.0/24
10.50.0.0/16
172.16.32.0/19

IX
public peering
Protocol encapsulation

- RIP uses UDP port 520 via IP broadcast/multicast
- IS-IS runs directly over layer 2 multicast
- OSPF is IP protocol 89, via IP multicast
- BGP uses TCP port 179, unicast of course
Example Implementations

- Cisco IOS
- Juniper JunOS
- Zebra and derivatives
  - Quagga, Vyatta
- BIRD Internet Routing Daemon
- OpenBGPD
- MikroTik RouterOS
BGP Remote Triggered Blackhole

• Goal: Have remote router /dev/null certain traffic
• Trick: use next-hop address that points to /dev/null
• Trick: Using policy, set next-hop for matching traffic
• Team Cymru bogon route server does this
• Many ISPs offer this as a DDoS relief service
• IPaddr getting packeted? Have upstream null it
• Also see IETF RFC 5635
unicast Reverse Path Check (uRPF)

• Goal: mitigate source address spoofing
• Trick: Validate source address to ingress interface
  • is there a route back via that interface?
• Loose versus strict mode
• Easier than ACLs (filters)? Maybe
  • doesn't work for everyone
• What do you do if you have a default route?
NetFlow

- Key router technology for analysis and monitoring
- NetFlow is not like pcap
- A unidirectional summary of traffic for a “flow”
- Flow is a unique tuple of addrs, proto, ports
- Router “exports” flows at timer, RST, FIN, etc
- Data may be limited due to sampling
- Scales very well and is very popular
NetFlow version 5 (most popular)

• Doesn't support IPv6 (need NetFlow v9 for that)
• saddr/daddr and netmasks
• Next hop address
• ingress/egress interface id
• Total bytes/packets in a flow
• Flow start/end time
• Protocols, ports, type/code, TCP flags, ToS bits
• src/dst ASN (peer or origin)
NetFlow illustrated

- http://flows.is-net.depaul.edu
- http://flows.ittns.northwestern.edu
Flow specification (flow-spec)

- Using BGP, exchange “flow-spec” to act on
- Largely used as a distributed firewall filter
- Can be more precise than a BGP RTBH
- Besides filtering, you can rate limit, log, pass
- Not widely implemented
Exchanges and Peering

- Networks need to connect to each other
- Question: Who pays who?
- Question: Where do they physically connect at?
- Peering is an entire “ecosystem” unto itself
- Paid versus settlement-free peering
- Transit versus peering and exchanges
- Peering requirements and network types
Overflow slide

- Other router management tools and processes
- Example routing configurations
- “traffic engineering”
- Class-of-service / quality of service
- Avoiding network capacity collapse
- Anycast - services on shared unicast addresses
- Mapping public Internet exchanges