

IK1350 Protocols in Computer Networks/ Protokoll i datornätverk Spring 2008, Period 3 Module 8: Dynamic Routing

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with *TCP/IP Protocol Suite*, by Behrouz A. Forouzan, 3rd Edition, McGraw-Hill, 2006.

For this lecture: Chapter 14



KTH Information and
Communication Technology

© 2008 G.Q.Maguire Jr. .

All rights reserved. No part of this course may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission of the author.

Last modified: 2008.02.06:07:34

Outline

- Dynamic Routing Protocols

Routing

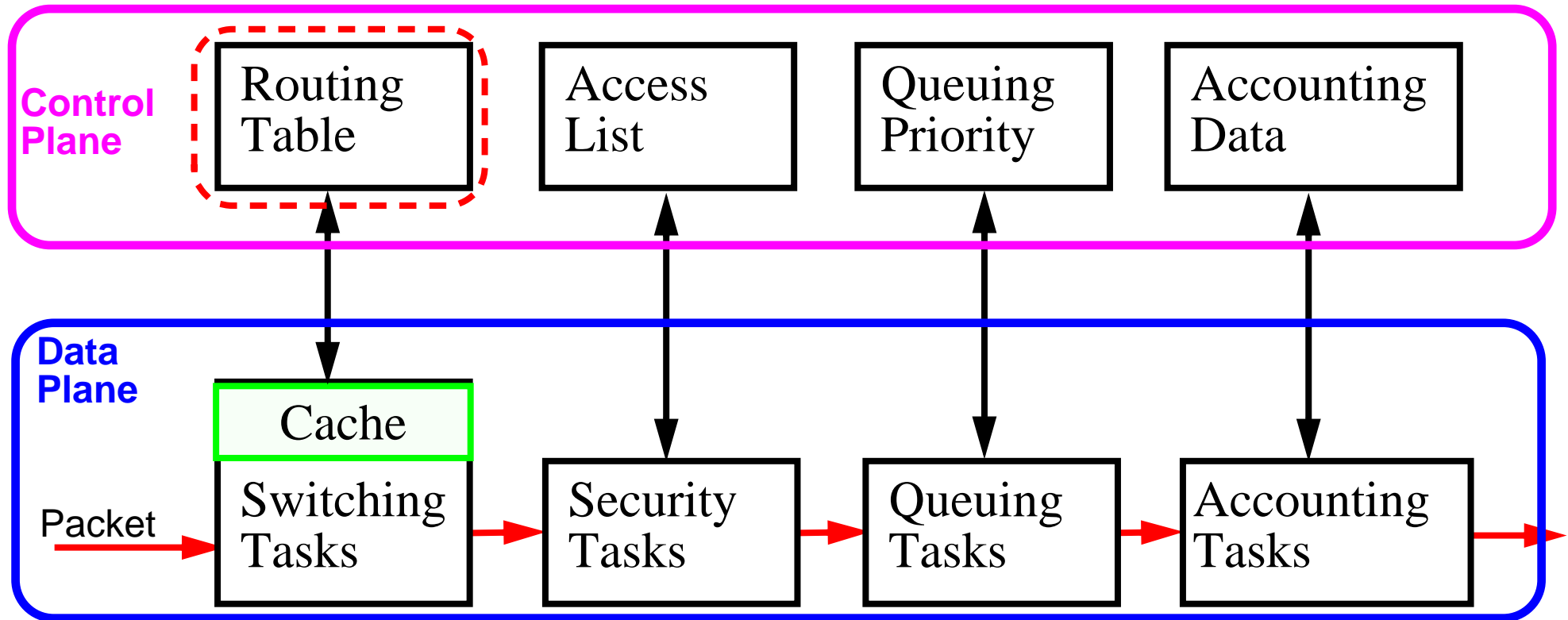


Figure 72: Basic steps in Routing

The routing table tells us which output port to use based on the destination (and possibly the source) IP address. The **data plane** has to run at packet rates (i.e., in real-time). However, a router also performs a lot of other processing

Routing Principles

- Routing **Mechanism**: Use the *most specific* route
 - IP provides the mechanism to route **packets**
- Routing **Policy**: What routes should be put in the routing **table**? <<<
today's topic!
 - Use a routing daemon to provide the *routing policy*

For further information see:

Bassam Halabi, *Internet Routing Architectures*, Cisco Press, 1997, ISBN 1-56205-652-2. -- especially useful for IGRP.

Basic Router Software Architecture

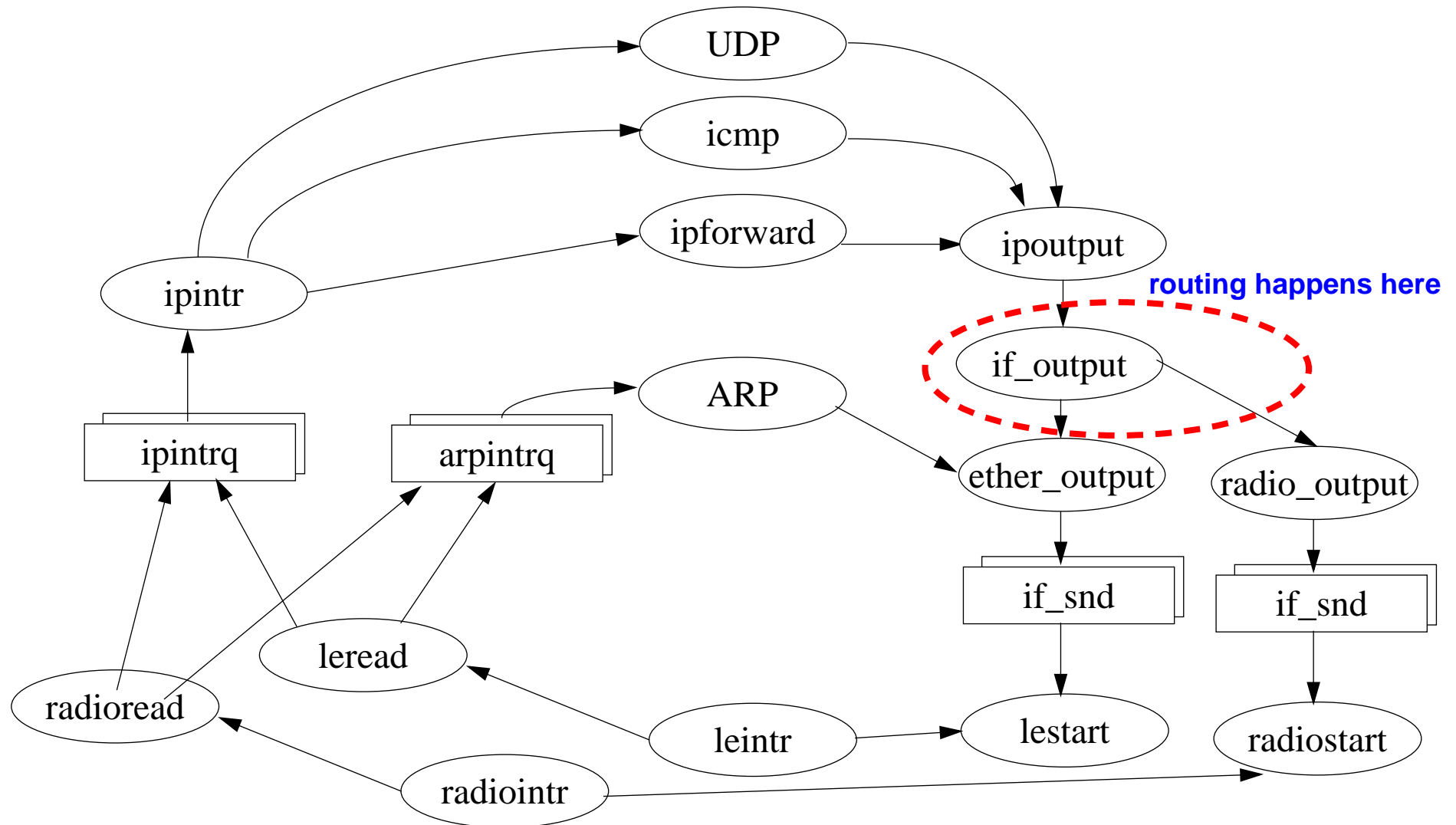
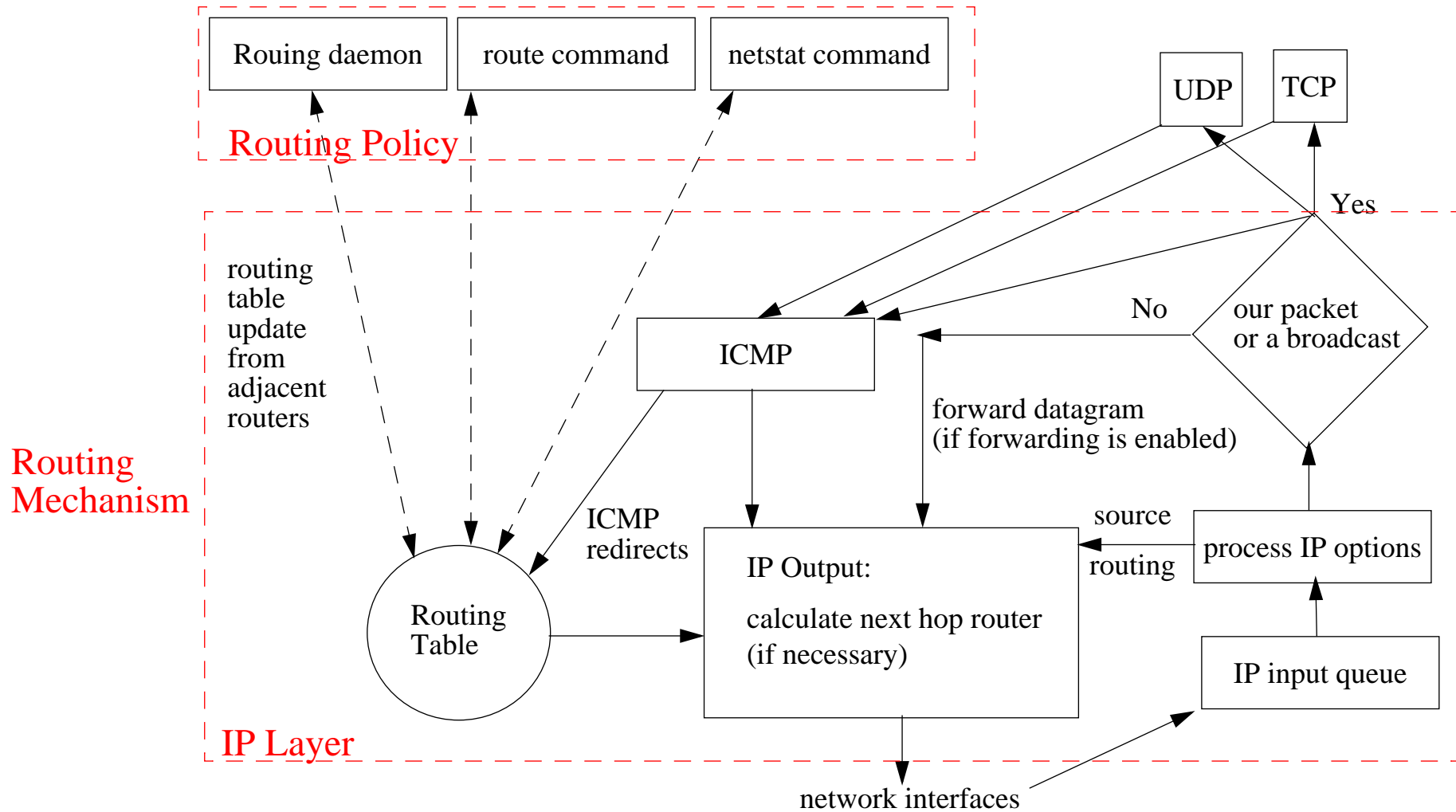


Figure 73: Basic Router Architecture (assuming a radio and an ethernet interface)

Processing



Routing packets in the Internet

Router needs to know where to route packets, to do this they need routing information. Such information can be provided by **manually entered routes** or ICMP Redirect or learning of routes via **a routing protocol**.

Dynamic routing protocols are based on routers talking to each other.

- **Intradomain** - within an AS (aka **Interior Gateway protocols**)
- **Interdomain** - between ASs (aka **Exterior Gateway protocols**)

The most popular dynamic routing protocols are:

- **RIP-1** - Routing Information Protocol (version 1)
- **RIP-2** - Routing Information Protocol (version 2)
- **OSPF** - Open Shortest Path First
- **BGP** - Border Gateway Protocol

Autonomous systems (ASs) - RFC1930 [45]

Each of which is generally administered by a **single entity**.

Each autonomous system selects the routing protocol to be used **within** the AS.

Network	AS number
Swedish University Network (SUNET)	AS1653 and AS2859
SUNET-KI	AS2837
Stockholm University - SU	AS2838
SUNET-KTH	AS2839
KTHNOC KTHNOC-SE	AS3224

For statistics about the number of AS, etc.: <http://www.cidr-report.org/>

For a list of AS number to name mappings: <http://www.cidr-report.org/autnums.html>

To find out who is responsible for a given autonomous system, use a query of the form: <http://www.ripe.net/perl/whois?AS2839>

Routing Metrics

A measure of which route is better than another:

- Number of hops
- Bandwidth
- Delay
- Cost
- Load
- ...

It is possible that the metric uses some **weighted combination** of the above.

Routing Algorithms

- Static vs. Dynamic
- Single path vs. Multi-path
- Flat vs. Hierarchical
- Host-intelligent vs. Router-intelligent
- Intradomain (interior) vs. Interdomain (exterior)
- Link state vs. Distance vector

Issues:

- Initialization (how to get started)
- Sharing
- Updating
- When to share & Who to share with

Intradomain routing protocols

also called “Interior Gateway protocols”

Examples:

- HELLO - an old IGP protocol
- RIP - widely used
- OSPF - increasingly used

Routers that speak **any** dynamic routing protocol **must** speak RIP and OSPF.

Routing Information Protocol (RIP) version 1

RFC 1058 [46] {Note it was written years after the protocol was in wide use!}

RIP is a **distance-vector** protocol - RIP messages contain a vector of **hop counts**.
RIP messages are carried via UDP datagrams which are **broadcast**.

0	7	8	15	16	23	24	31
Command		Version = 1			Reserved		
Family			All 0s				
Network Address							
All 0s							
All 0s							
Distance							

Figure 74: RIP message format (see Forouzan figure 14.9 pg. 394)

- a command: **request** or **reply**
- a version number (in this case 1)
- up to 25 instances (entries) containing:
 - address family (2 = IP addresses)
 - Network Address (allocated 14 bytes, for an IP address we only need 4 bytes - and they are aligned to a 4 byte boundary - hence the leading and trailing zeros)
 - metric [hop count]

RIP v1 operation

As carried out by UNIX daemon “routed” using UDP port 520

Initialization:

```
for all interface which are up
{
    send a request packet out each interface asking for the other
    router's complete routing table
        [command=1, address family=0 {== unspecified}1, metric=16
    }
}
```

Request received:

```
if whole table requested, then send it all 25 at a time
else if a specific set of routes
    then fill in the metric
    else set metric to 16
        [16 == "infinity" == we don't know a route to this address]
```

Response received:

```
if valid (i.e., not 16), then update/add/delete/modify routing table
```

1. Page 24 of RFC 1058 says “If there is exactly one entry in the request, with an address family identifier of 0 (meaning unspecified), and a metric of infinity (i.e., 16 for current implementations), this is a request to send the entire routing table.”[46] - this is different than implied in Forouzan figure 14.10 pg. 395

When are routes sent?

Solicited response: Send a response when a request is received

Unsolicited response:

- If a metric for a route changes, then (trigger) send update, else send all or part of the table every 30 seconds.
- If a route has not been updated for 180 seconds (3 minutes = 6 update cycles), then set metric to 16 and then **after** 60 seconds (1 minute) delete route.

Metrics are in units of hops, thus this protocol leads to selection between routes based on the minimum number of hops.

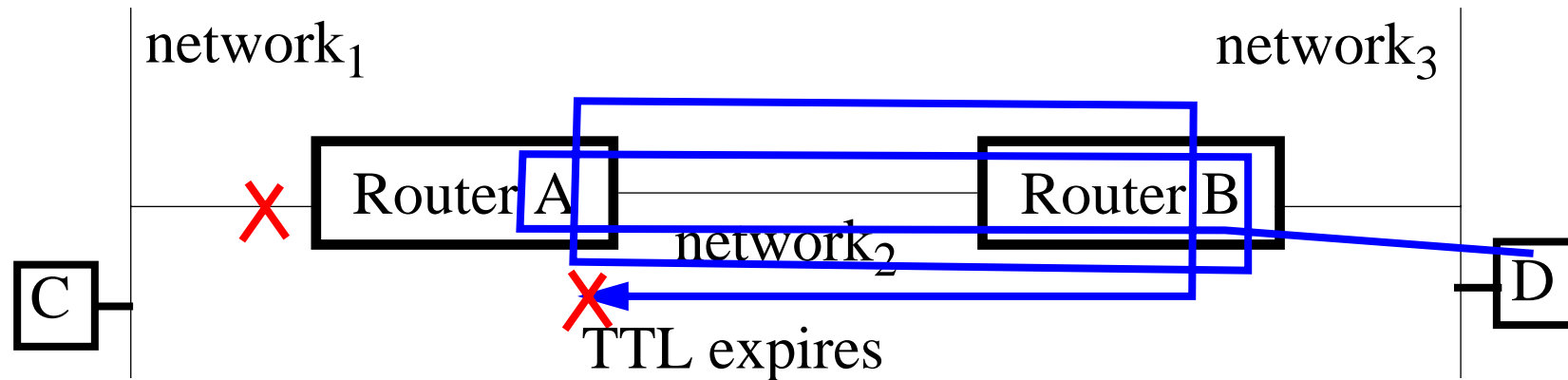
Summary of RIPv1 Timers:

- **Periodic** timer - regular updates random value [25..35] mean 30 s.
- **Expiration** timer - routes not updated within (180 s) expire
- **Garbage collection** timer - 120 s after expiring entries are GC'd

Problems with RIP v1

- RIPv1 does **not** know about subnets (or assumes all interfaces on the network have the same netmask)
- after a router or link failure RIP takes **minutes** to stabilize (since each neighbor only speaks ~every 30 seconds; so the time for the information to propagate several hops is minutes) \Rightarrow while it is unstable it is possible to have routing loops, etc.
- Hops count may **not** be the best indication for which is the best route
- Since the maximum useful metric value is 15, the network diameter must be less than or equal to 15.
- RIP will accept updates from **anyone** - so one misconfigured device can disrupt the entire network.
- RIP uses more bandwidth than other protocols, since it sends the whole routing table in updates.

Count to Infinity



- Router A advertises it knows about routes to networks 1 and 2
- Router B advertises it knows about routes to networks 2 and 3
- After one update cycles A and B know about all 3 routes.

If A's interface to Network₁ goes down, then A learns from B - that B knows a route to Network₁; so A now thinks it can reach Network₁ via B. So if D sends a packet for C, it will simply loop back and forth between routers A and B, until the TTL counts down to 0.

Split Horizon

To counter the count to infinity, the split horizon algorithm - never sends information on an interface that it learned from this interface.

RIPv1 implements: Split Horizon with **Poison Reverse Update** - rather than not advertise routes to the source, we advertise them with a metric of 16 (i.e., unreachable) - hence the source simply ignores them.

Unfortunately split horizon only prevents loops between **adjacent** routers (so if there are three or more routers involved the previous problem re-appears)

Triggered updates and Hold-Downs

To decrease convergence time - when the topology changes send out an update immediately.

However, if a node can learn about connectivity from more than one source, then if a delete happens before the add, then the existence of the route is asserted; therefore the hold-down rules says:

When a route is removed, no update of this route is accepted for some period of time - to give everyone a chance to remove the route.

This period of time is the Hold-down time.

The result is to **decrease** the rate of convergence; but the combined effect of triggered updates + hold-downs leads to **faster** convergence than not using triggered updates.

RIP extensions (aka RIP-2)

Defined in RFC 1388 [47] and revised in RFC 2453[48]. Version number is 2.

- for each of up to 25 entries we add the fields:
 - **Route tag** - carries the AS number
 - **Subnetmask** - subnetmask to be used with this address (to support **classless addressing**)
 - **Next-hop IP address**, either the IP address of where packets to this destination should be sent or zero [which means send them to the system which sent the RIP message]
- One entry can be replaced by **Authentication** data

RIP-2 supports multicast to address 224.0.0.9, to reduce load on hosts **not** interested in RIP-2 messages

0	7 8	15 16	23 24	31
Command		Version = 2		Reserved
0xFFFF		Authentication type		
Authentication data (16 bytes)				
if Authentication type = 2, this is a clear text password to be used to authenticate this message				
Family		Route tag		
Network Address				
Subnet mask				
Next-hop address				
Distance				

Figure 75: RIPv2 message format (see Forouzan figures 14.13 pg. 397 and 14.14 pg. 398)

Why would anyone use RIP?

After all these problems you might ask this question.

Answer

- Because RIP is generally the only routing protocol which **all** UNIX machines understand!
- Relatively easy to configure
- It is widely available, since it **must** exist if the device is capable of routing!

Interior Gateway Routing Protocol (IGRP)

Cisco's IGRP [49] - a proprietary protocol with the following goals:

- stable, optimal routing for large networks - with no routing loops
- fast response to changes in net topology
- low overhead in both bandwidth and processor utilization
- ability to **split traffic across several parallel routes** if they are (or nearly are) equal.

It is a distance-vector protocol based on many of the ideas from RIP.

IGRP Metrics

- a vector of metrics each with a 24 bit value
- composite metric is $\left[\left(\frac{K_1}{B} \right) + \left(\frac{K_2}{D} \right) \right] \cdot R$, where K_1 and K_2 are constants, B the unloaded path bandwidth, D a topological delay, and R is reliability
- also we pass the **hop count** and **Maximum Transmission Unit** values

K_1 is the weight assigned to bandwidth (by default 10,000,000)

K_2 is the weight assigned to delay (by default 100,000)

If up to 4 paths are within a defined **variance** of each other, Cisco's IOS (Internetwork Operating System) will split the traffic across them in inverse proportion to their metric.

IGRP Route Poisoning

IGRP poisons routes which increase by a factor of 10% or more after an update [they are thought to be: “too good to be true”].

While this rule may temporarily delete valid routes (which will get reinstated after the next regular update) - it allows use of a zero hold-down time, which leads to faster convergence.

IGRP Default Gateway

Rather than using the fake network 0.0.0.0 to indicate the default network, IGRP allows a real network to be flagged as the default network.

Periodically, IGRP scans the routes offering a path to this flagged network and selects the path with the lowest metric.

Note: Default gateways help to keep the size of local routing tables smaller.

Enhanced IGRP (EGRP) [50]

Uses the distance-vector technology of IGRP, but changes the way routes are advertised and the calculation of entries for the routing table.

EGRP uses:

- a neighbor discover/recovery process of hello packets to learn its neighbors
- a Reliable Transport Protocol to ensure guaranteed, ordered delivery of routing updates
- a Diffusing Update Algorithm (DUAL) - which selects both the best route for insertion into the table **and** a **feasible successor** (for if the primary route fails)
- Variable length subnet masks (VLSM)

EGRP is a Cisco proprietary technology.

Open Shortest Path First (OSPF)

OSPF defined in RFC2328 (see also <http://rtg.ietf.org/wg/ospf/>)

OSPF is a **link-state** protocol. OSPF messages (Link State Advertisements (LSAs)) tell the **status of links** of each of its neighbors and propagates this info to its neighbors. Each router uses this link-state information to build a **complete** routing table. Uses IP directly (protocol field = OSPF (89)) \Rightarrow does **not** use UDP or TCP.

Advantages

- link-state protocols converge faster than distance-vector protocols
- can calculate a route per IP service type (i.e., TOS)
- each interface can have a per TOS cost
- if there are several equally good routes \Rightarrow can do **load balancing**
- supports variable length subnet masks
- enable point to point links to be **unnumbered** (i.e., don't need an IP address)
- uses clear text passwords
- uses multicasting

OSPF uses the Shortest Path First algorithm (also known as [Dijkstra's algorithm](#)).

OSPF networks are generally divided into [areas](#) such that cross-area communication is minimal.

Some routers with multiple interfaces become [border area routers](#) (with one interface in one area and another interface in another area).

The only way to get from one area to another area is via the [backbone](#) - which is area 0. Note: The backbone need **not** be continuous.

- Note that Forouzan refers to “transient” links -- I think that this should be “transit” links. (Since transient implies that the link would be short lived!)

Link state advertisements are sent to all routers in a given area (via [flooding](#)), rather than just neighbors (as in the distance-vector approach) - thus periodic updates are infrequent (every 1 to 2 hours).

A key feature of OSPF is [route aggregation](#) - which minimizes the size of routing tables and the size of the topological database; in addition, it keeps protocol traffic to a minimum.

OSPF building blocks

1. Hello protocol

- Check for & with neighbors and learn designated routers

2. Synchronization of Databases

- Exchange of Link State Database between neighbors
 - Get LSA **headers**
 - Request the transfer of necessary LSAs

3. Flooding protocol

- When links change or when your knowledge is old
 - Send Link State updates to neighbors and flood recursively
 - If not seen before, propagate updates to all adjacent routers, except the router you received it from

OSPF Packets

Common header

0	7	8	15	16	23	24	31
Version		Type		Message length			
Source Router IP Address							
Area Identification							
Checksum				Authentication type			
Authentication (64 bits) ^a							

Figure 76: OSPF Common Header (see Forouzan figure 14.26 pg. 408)

a. Note that the Authentication field is 64 bits, Forouzan figure 14.26 pg. 408 incorrectly shows it as 32 bits.

The 5 types of OSPF packets:

Type	Description
1	Hello
2	Database Description
3	Link State Request
4	Link State Update
5	Link State Acknowledgment

Hello packet

0	7	8	15	16	23	24	31	
Version		Type = 1			Message length			
Source Router IP Address								
Area Identification								
Checksum				Authentication type				
Authentication (64 bits)								
Network Mask								
Hello interval (seconds)				All zeros		E	T	Priority
Dead interval (seconds)								
Designated router IP address								
Backup designated router IP address								
Neighbor IP address								
...								
Neighbor IP address								

Figure 77: OSPF Link state update packet (see Forouzan figure 14.44 pg. 419)

- E = 1 indicates a stub area
- T = 1 indicates router supports multiple metrics
- priority = 0 indicates that this router should not be considered as a designated or backup designated router
- Dead interval is the time before a silent neighbor is assumed to be dead
- list of neighboring routers (of the router which sent the hello packet)

Database Description packet

0	7	8	15	16	23	24	31
Version		Type = 2			Message length		
Source Router IP Address							
Area Identification							
Checksum				Authentication type			
Authentication (64 bits)							
Interface MTU				All zeros	E	B	All zeros
						I	M
						M	S
Database Description sequence number							
LSA header (20 bytes)							
...							
LSA header							

Figure 78: OSPF Database Description packet (see Forouzan figure 14.45 pg. 420)

- Rather than send the entire database - send an outline of it
 - E = 1 indicates the advertising router is an autonomous boundary router (i.e., E ≡ external)
 - B = 1 indicates the advertising router is an autonomous border router
 - I = 1 initialization flag
 - M = 1 ≡ More flag
 - M/S flag: 0=slave, 1=Master
 - Database Description sequence number
 - LSA header(s) - gives information about the link - but **without** details; if details are desired they can be requested

Link State Announcement (LSA) header

Each different LSA has the same general header:

0	7 8	15 16	23 24	31	
Link state age (seconds)		Reserved	E	T	Link State type
Link state ID					
Advertising router					
Link state sequence number					
Link state checksum			Length		

Figure 79: OSPF Link State Advertisement general header format (see Forouzan figure 14.28 pg. 410)

- E = 1 indicates a stub area
- T = 1 indicates router supports multiple metrics

Link state type	Link state ID	Description
1	IP address of the router	Router-LSAs
2	IP address of the designated router	Network-LSAs
3	IP address of the network	Summary-LSAs (IP network)
4	IP address of the border router	Summary to AS border router (BR)
5	IP address of the external network	AS-external-LSAs

- Advertising router - IP address of the router advertising this message
- Link state checksum - a Fletcher's checksum of all but age field

- Length of the whole packet in bytes

Link state update packet

0	7	8	15	16	23	24	31
Version		Type = 4			Message length		
Source Router IP Address							
Area Identification							
Checksum				Authentication type			
Authentication (64 bits)							
Number of link state advertisements							
Link state advertisement (LSA)							
...							
Link state advertisement (LSA)							

Figure 80: OSPF Link state update packet (see Forouzan figure 14.27 pg. 409)

Link state request packet

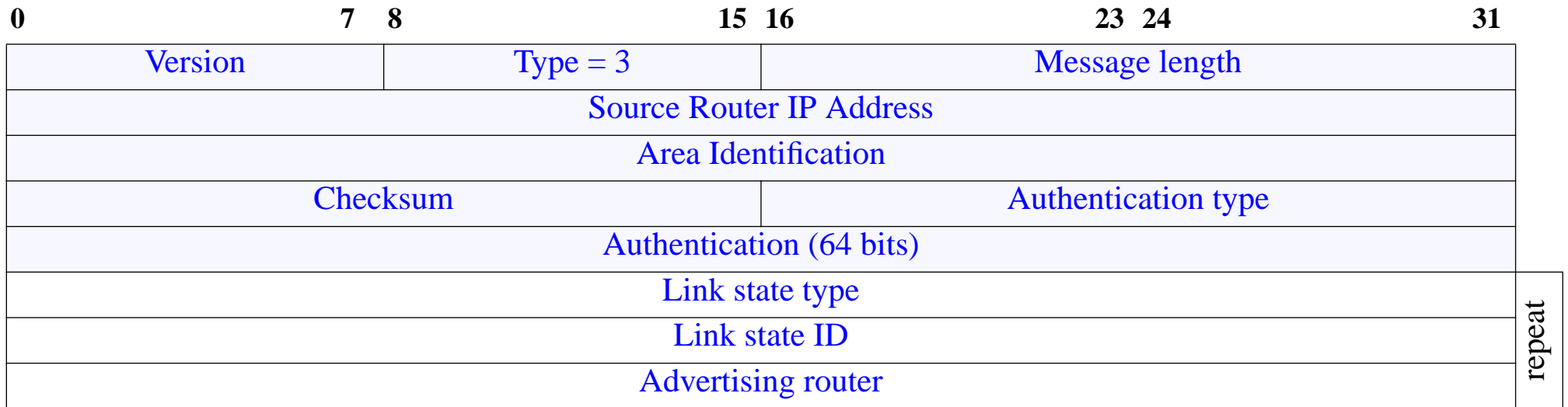


Figure 81: OSPF Link state request packet (see Forouzan figure 14.46 pg. 420)

To ask for information about a specific route (or routes); reply is an update packet.

Link state acknowledgement packet

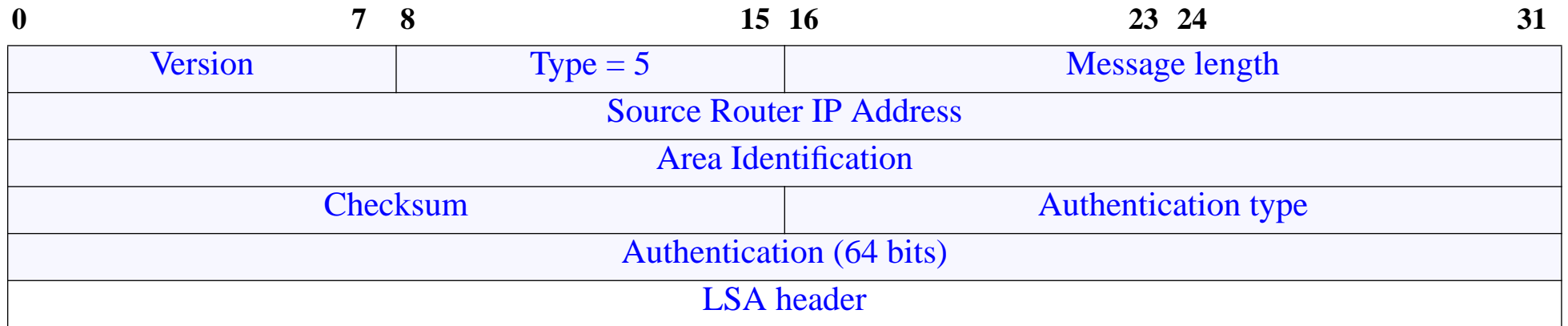


Figure 82: OSPF Link state acknowledgement packet (see Forouzan figure 14.47 pg. 421)

To acknowledge receipt of an update packet.

Interdomain routing protocols

also called “Exterior Gateway Protocols (EGPs)” - used **between** ASs

Examples:

- EGP - an old EGP protocol
- BGP - Border Gateway Protocol

Intradomain routing protocols:

- don't scale up to the large numbers of routers involved in interdomain routing (due to the huge computational resources required)
- distance vector routing becomes unstable beyond a few hops

Exterior Gateway Protocol (EGP)

an exterior gateway protocol with three components:

- neighbor acquisition
- neighbor reach ability, and
- routing information

EGP was designed to provide more automation in configuring routers.

EGP is similar to the distance-vector protocols, but **omits the metrics**, since EGP was designed for the internet where typically routers are connected to a backbone (with its own routing domain) via a single router.

- But since there are no metrics, if there is more than one path, then there can be a loop!

BGP - Border Gateway Protocol

An exterior gateway protocol to exchange routing information between routers in different ASs.

BGP version 3 defined in RFC 1267[53], while version 4 defined in RFC1654[54] and RFC 1771[55].

BGP routers exchange routing information with other BGP routers.

For further information see:

John W. Stewart III, *BGP4: Inter-Domain Routing in the Internet*, Addison-Wesley, 1999, ISBN: 0-201-37951-1 [56]

See also [64] and [65].

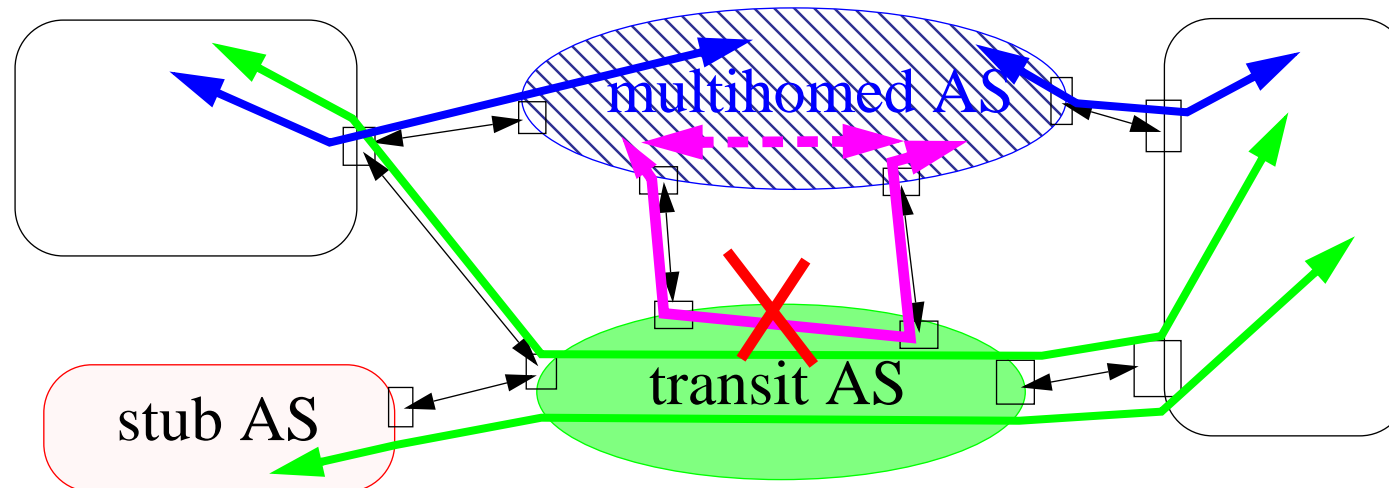
Local vs. Transit traffic

Local traffic: originates or terminates in an AS

Transit traffic: is all other traffic

==> 3 types of ASs

stub AS	connected to only on other AS, carries only local traffic
multihomed AS	connected to multiple AS, but refuses transit traffic
transit AS	connected to multiple ASs, carries both local and transit traffic



BGP operation

BGP routers exchange information based on traffic which transits the AS, derives a graph of AS connectivity; with loop pruning.

Routing policy decisions can be enforced as to what is allowed to transit whom
⇒ **policy-based routing**

- based on economic/security/political/... considerations.
- BGP does **not** implement the policy decisions, but allows the information on which such decisions can be made to propagate as necessary

Uses **TCP** (port 179) to create a session between BGP routers:

- initially two systems exchange their entire BGP routing table,
- then they simply send updates as necessary.

BGP is a **path-vector protocol** - which **enumerates** the route to each destination (i.e., the sequence of AS numbers which a packet would have to pass through from a source to its destination) = a **path vector**

BGP does **not** transmit metrics.

However, each path is a list of attributes:

- **well-known attributes** (which every router must understand)
 - well-known **mandatory** attribute - must appear in the description of a route
 - well-known **discretionary** attribute - may appear, but must be recognized, in the description of a route
- **optional attributes**
 - optional **transitive** attribute - must be passed to the next router
 - optional **nontransitive** attribute - the receiving router must discarded it if does not recognize it

For examples of the use of an attribute see [62] and [63]

BGP detects failures (either links or hosts) by sending **keepalive** messages to its neighbors. Generally sent every 30 seconds and as they are only 19 bytes each \Rightarrow only **~5 bits/second** of bandwidth, but with very long lived TCP connections (**semi-permanent connections**)

A major feature of BGP version 4 is its ability to do **aggregation** - to handle CIDR and supernetting. For more information on aggregation see chapter 5 of [57].

Classless Inter-Domain Routing (CIDR)

A standard for both classless addressing and classless interdomain routing scheme (RFCs 1517 [58] .. 1520 [61]).

- Basic concept: to allocate/collapse a block of contiguous IP addresses into a single routing table entry: (network address, count). e.g., 192.5.48.0, 192.5.49.0, 192.5.50.0 = (192.5.48.0, 3)
- **Hierarchical Routing Aggregation** minimizes routing table entries; enables "route aggregation" in which a single high-level route entry can represent many lower-level routes in the global routing tables.
 - Reduces the growth of routing table.
- Allows the addresses assigned to a single organization to span multiple classed prefixes.
- Envisioned a hierarchical Internet.

CIDR addressing scheme and route aggregation has two major user impacts:

- you have to **justify** IP Address Assignments
- get address from your ISP, i.e., **renting** them vs. being **assigned** them

Redistribution of Route Information between protocols

Redistribution: allows a router running more than one routing protocol to distribute information from one protocol to another.

Thus at the border, a router will translate the information obtained from one routing domain and pass it to the other routing domain in the appropriate manner.

- Advertise (aggregated) interior routes to the Internet
- Inject (some) exterior routes into the interior network

Usually the redistributed routes are filtered (as not all the information needs to cross the border).

Resulting BGP Architecture

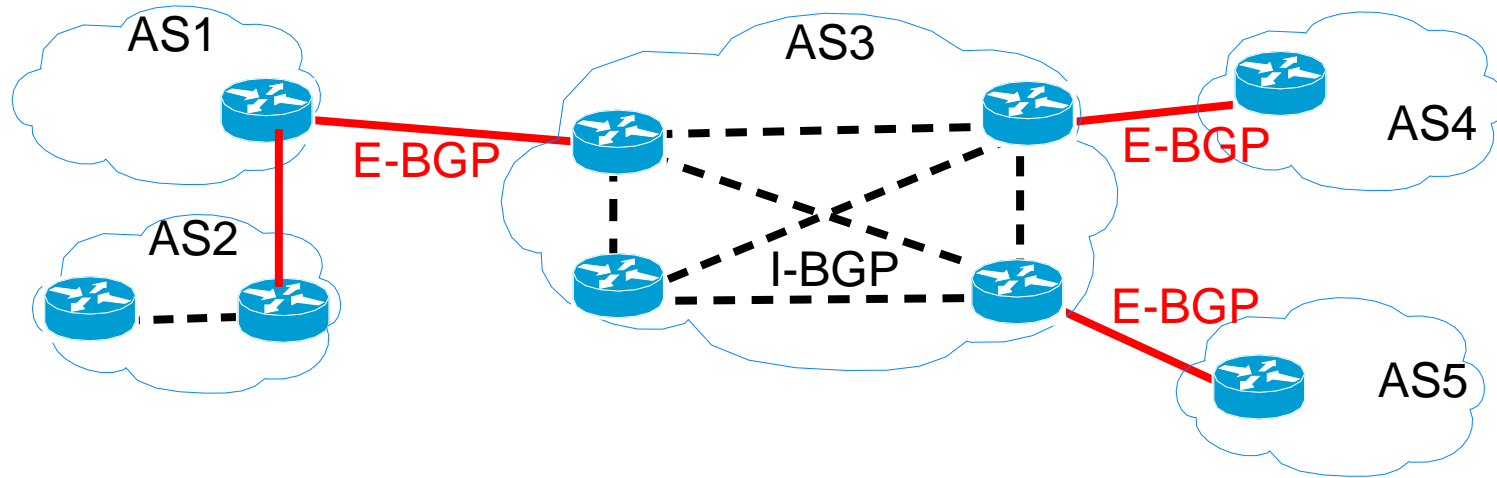


Figure 83: BGP architecture

Two kinds of BGP sessions:

- External (**E-BGP**) coordinates between border routers **between** ASs
- Internal BGP (**I-BGP**) coordinates between BGP peers **within** an AS
 - Note it must be a full mesh, but this does not scale \Rightarrow organization into subASs,

BGP Messages

- Open
- Update
- Keepalive
- Notification

BGP Open Message

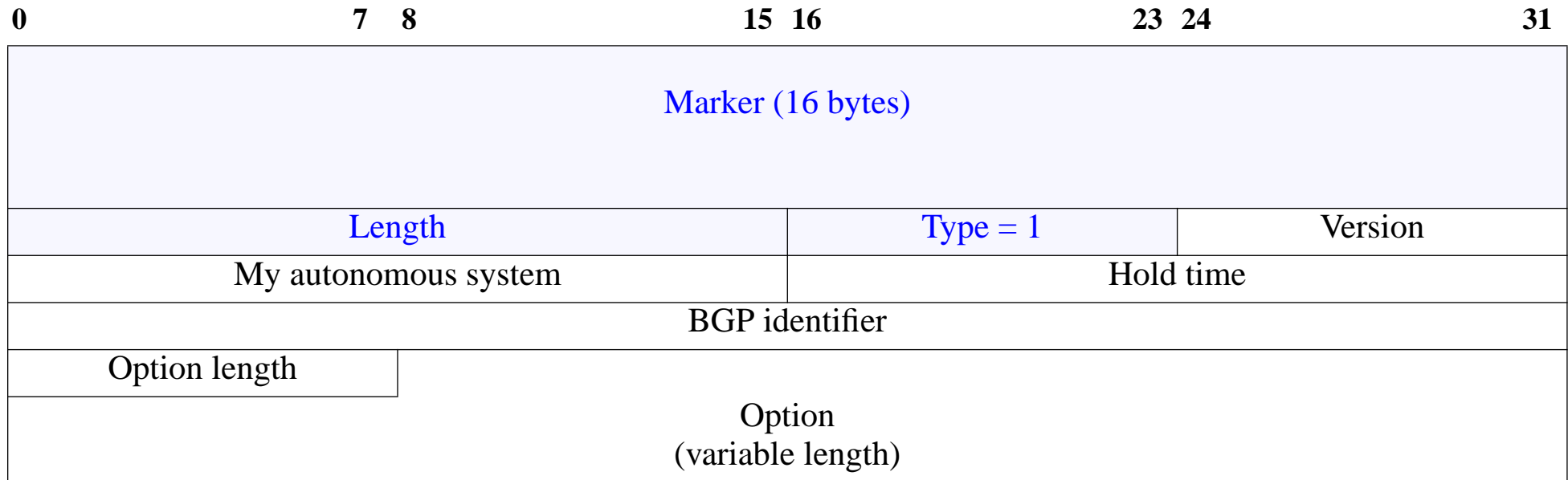


Figure 84: BGP Open messages (see Forouzan figures 14.53 pg. 427 and 14.52 pg. 426)

Marker (for authentication), **Length**, and **Type** are common to all BGP messages.

- Version = 4
- My autonomous system - the AS number
- Hold time - maximum time to wait for a keepalive or update, otherwise the other party is considered to be dead
- BGP identifier - identifies the router sending this message (typically its IP address)
- Option length - zero if none
- Option - options in the form (length of parameter, parameter value)

BGP Update Message

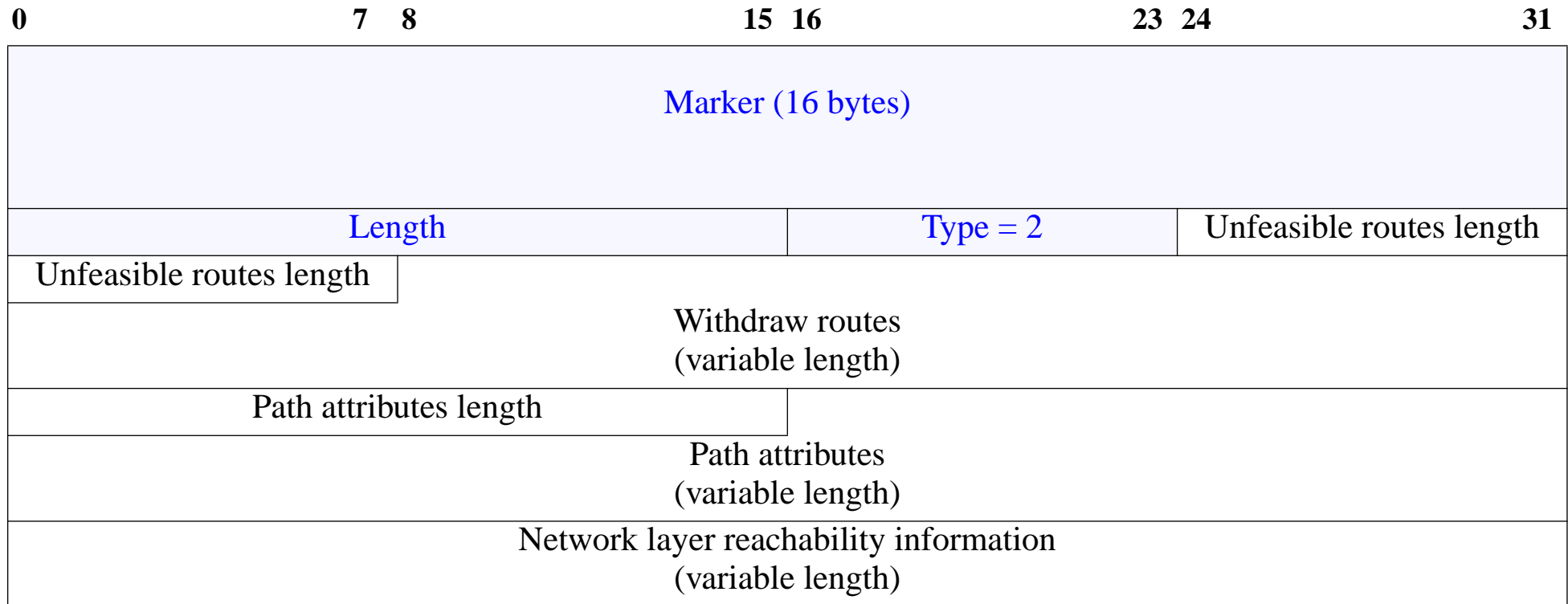


Figure 85: BGP Update message (see Forouzan figures 14.54 pg. 428 and 14.52 pg. 426)

- Unfeasible routes length (2 bytes) - length of next field
- Withdraw routes - list of all routes that must be deleted
- Path attributes length(2 bytes) - length of next field
- Path attributes - specifies the attributes of the path being announced
- Network layer reachability information - prefix length and IP address (to support CIDR)

BGP Keepalive Message

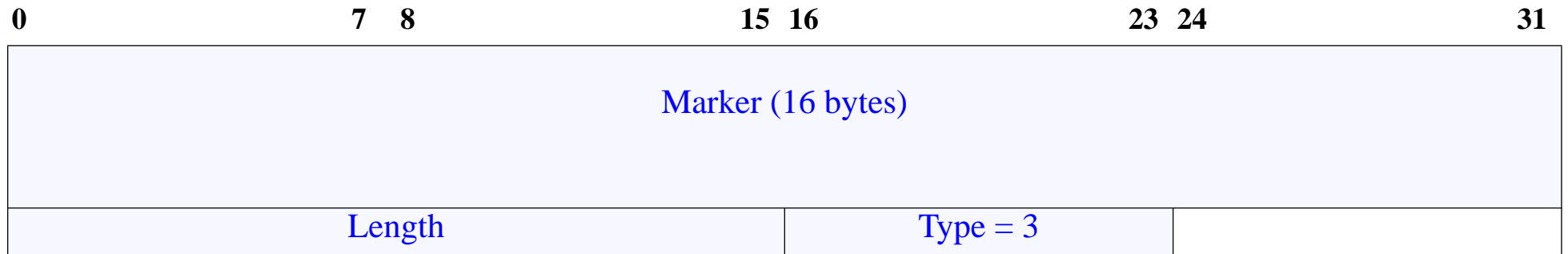


Figure 86: BGP Keepalive message (see Forouzan figures 14.55 pg. 429 and 14.52 pg. 426)

- Sent to reassure your peer that you are still alive

BGP Notification Message

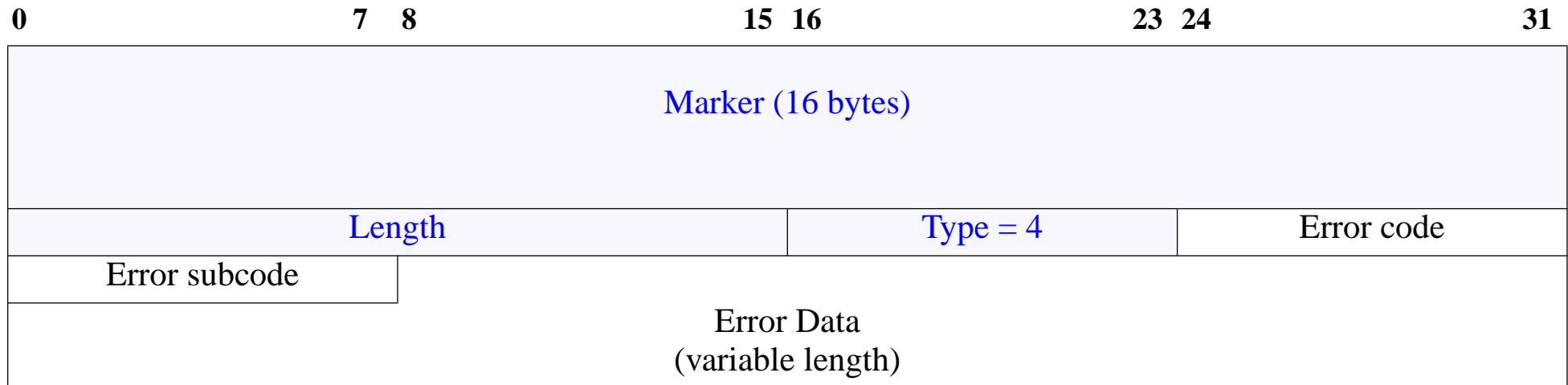


Figure 87: BGP Notification message (see Forouzan figures 14.56 pg. 429 and 14.52 pg. 426)

- Error code (1 bytes) - category of error
 - 1 = Message header error
 - 2 = Open Message error
 - 3 = Update Message error
 - 4 = Hold time expired
 - 5 = Finite state machine error
 - 6 = Cease
- Error subcode (1 bytes) - the particular error in this category
- Error Data - information about this error

Interconnections of networks

Since different networks have different users, policies, cost structure, etc.

But, the value of a network is proportional to the (number users)² [Metcalfe's Law]

Therefore, network operators **want** to connect their networks to other networks.

⇒ Internet eXchange Points (IXs or IXPs)

List of public internet exchange points: <http://www.ep.net/ep-main.html>

For a discussion of why IXPs are important see [66]

- No internet exchange points ⇒ no internetworking!
- Cost advantages in peering
- QoS advantages

Federal Internet eXchange (FIX)

A top-level routing domain - i.e., it does not use default routes.

Each was built around an FDDI ring which interconnected routers from each of the operators.

Each of these routers was in turn connected to the rest of the operator's network via a high speed link (often at speeds up to 45Mbps).



Note that it need not be a physical ring, but was often an FDDI switch (such as the DEC Gigaswitch/FDDI).

Commercial Internet eXchange (CIX)

A nonprofit trade association of Public Data Internetwork Service Providers.

- a neutral forum - for forming consensus on legislation and policies
- fundamental agree for all CIX members to interconnect with on another
- no restriction on traffic between member networks
- no “settlements” or traffic charges

Global Internet eXchange (GIX)

Global Internet eXchange (GIX), Guy Almes, Peter Ford, Peter Lothberg.

proposed in June 1992 - Stockholm D-GIX became Netnod-IX

Some of Sweden's Internet exchange points

- Luleå Internet Exchange ⇒ Polarix <http://www.polarix.se/>
- NorrNod <http://www.norrnod.se/>
- NETNOD Internet eXchange <http://www.netnod.se/>
- RIX -GH Gävleborg Regional Internet Exchange <http://www.rix-gh.se/>
- SOL-IX - Stockholm <http://www.sol-ix.net/>

Other useful contacts:

SNUS (Swedish Network Users Society)

“... its goal, from the users perspective, to force the evolution and development of the networks and interconnections between networks, to arrange seminars, to exchange information between the members, and to write agreements with companies.”

- SOF (Swedish Operators Forum)
- North American Network Operators' Group (NANOG) <http://www.nanog.org/>
- ...

Network Access Points (NAPs)

At the NAP a high-speed network or switch is used to interconnect a number of routers for the purpose of exchanging traffic.

Started with several NSF sponsored NAPs:

- Sprint NAP - in Pennsauken NJ
- PacBell NAP - in San Francisco
- Ameritech Advanced Data Services (AADS) {now SBC} NAP - in Chicago (ATM exchange point) <http://www.aads.net/main.html>
- MAE-East (part of MCI) - in Washington DC <http://www.mae.net/>
- MAE-West (part of MCI) - in San Jose CA

In addition to handling IP packets, NAPs were required to support InterDomain Routing Protocol (IDRP) {the ISO OSI Exterior Gateway Protocol} and route CLNP (ConnectionLess Networking Protocol) packets.

NAPs today

Using GigE, switch fabrics, resilient packet ring (Spatial Reuse Protocol (SRP)) technology, e.g., Cisco's Dynamic Packet Transport (DPT), ... with dedicated fiber connections to/from members.

NAP managers are increasingly concerned about security, reliability, and accounting & statistics.

Various NAP have different policies, methods of dividing costs, fees, co-location of operators equipment at the NAP, etc.

Router Arbiter Project

Router Arbiter (RA) -<http://www.ra.net> (July 1994 through March 1998)

- provide a common database of route information (network topology and policies) [Routing Arbiter Database (RADB) became Routing Assets Database (RADb) <http://www.merit.edu/nrd/services/radb.html>]
- promote stability and manageability of networks

Instead of a full mesh connection between providers, all the providers peer with a central **router server**. A Router server (RS):

- maintains a database of all information operators need to set their routing policy (written in RIPE 181, see RFC 1786 ⇒ Routing Policy Specification Language (RPSL))
<http://www.merit.edu/internet/documents/internet-drafts/draft-blunk-rpslng-08.txt>)
- does not forward packets or perform any switching function
- a distributed rover runs at each RS and collects information which the central network management system later queries.

Internet Routing Registry (IRR)

- a neutral Routing Registry
- set of routing DBs which includes
 - RIPE Routing Registry (European ISPs) - <http://www.ripe.net/db/irr.html>
 - MCI Routing Registry (MCI customers)
 - CA*net Routing Registry (CA*net customers)
 - ANS Routing Registry (ANS customers)
 - JPRR Routing Registry (Japanese ISPs)
 - Routing Arbiter Database (RADB) (all others)
- entries are maintained by each service provider

Internet Performance and Analysis Project (IPMA) IRR Java Interface

<http://salamander.merit.edu/ipma/java/IRR.html>

Euro6IX

The European IPv6 Internet Exchanges (Euro6IX) project [67] examined how to build an IPv6 exchange. It contains good examples of how to combine both switching and routing mechanisms to build a high performance internet exchange.

They also describe additional services which such an exchange might provide. These range from DNS to content distribution.

Cisco's NetFlow Switching

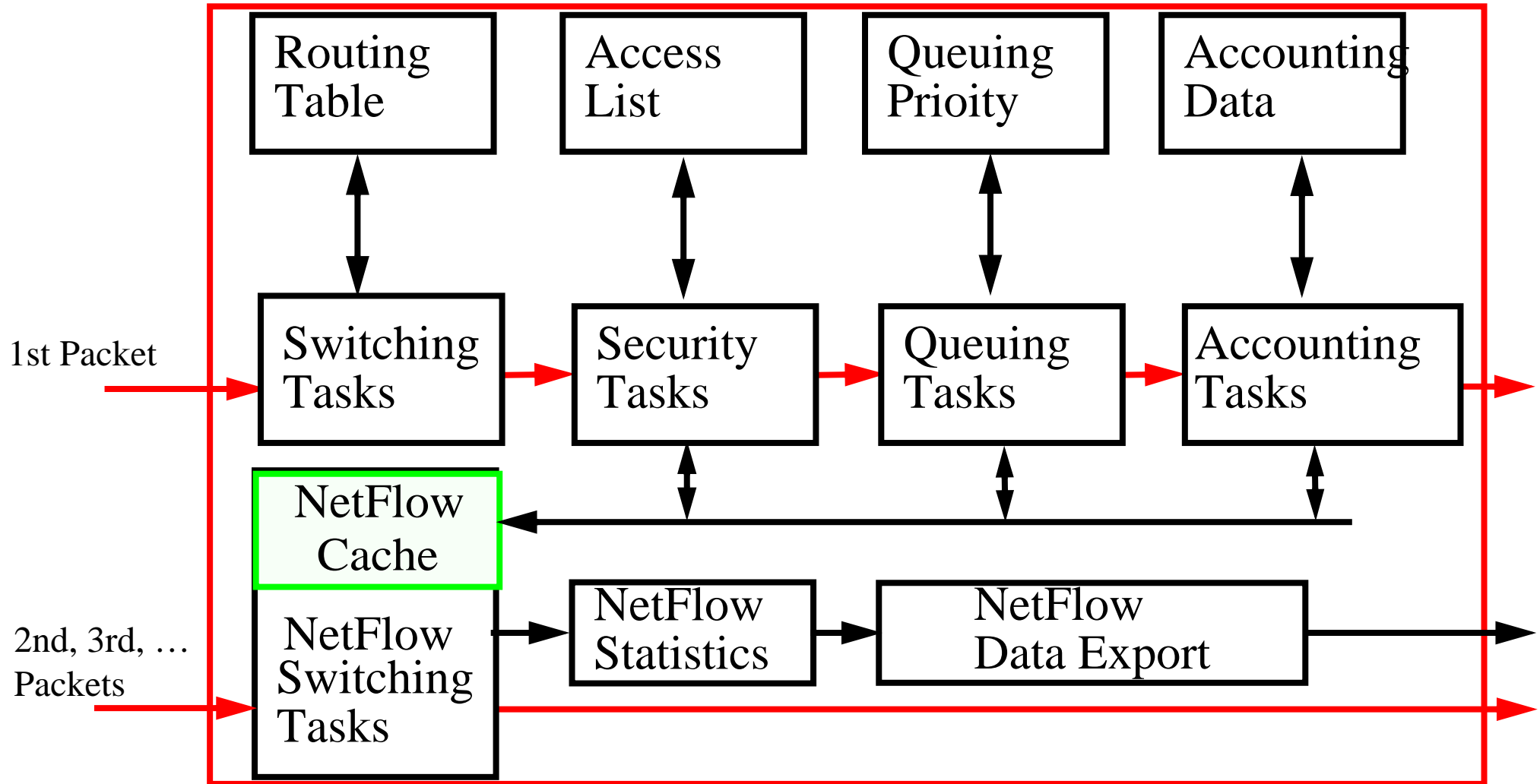


Figure 88: Cisco's NetFlow Switching

Flows

A flow is defined as a “uni-directional stream of packets between a given source network-layer address and port number and a specific destination network-layer address and port number”.

Since many application use well known transport-layer port numbers, it is possible to identify **flows per user per application basis**.

There is a well defined Netflow Switching Developer’s Interface which allows you to get the statistics concerning the NetFlow cache and the per flow data (the later gives you essentially billing records).

A general introduction to NetFlow Switching is available at

http://www.cisco.com/univercd/cc/td/doc/product/software/ios120/12cgcr/switch_c/xcp3/xcovntfl.htm

Cisco's Tag Switching

Combine routing with the performance of switching, based on the concept of “label swapping”, in which units of data (e.g., a packet or a cell) carry a short, fixed length label that tells switching nodes how to process the data.

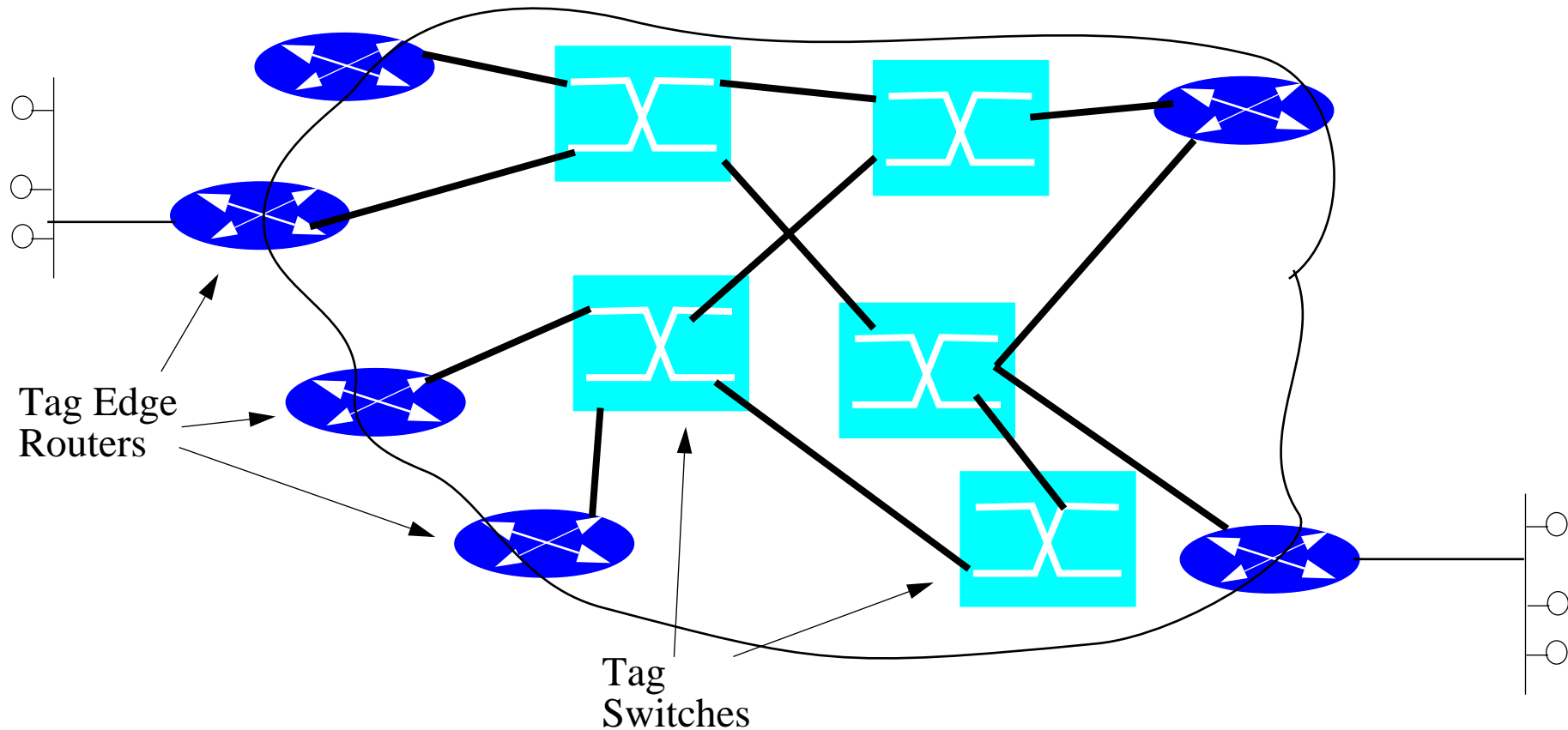


Figure 89: Tag Switching

A Tag Edge router labels a packet based on its destination, then the Tag Switches make their switching decision based on this tag, without having to look at the contents of the packet.

The Tag Edge routers and Tag Switch exchange tag data using Tag Distribution Protocol (TDP).

Basics of Tag switching:

1. Tag edge routers and tag switches use standard routing protocols to identify routes through the internetwork.
2. Using the tables generated by the routing protocols the tag edge routers and switches assign and distribute tag information via the tag distribution protocol (TDP). When the Tag routers receive this TDP information they build a forwarding database.
3. When a tag edge router receives a packet it analyzes the network layer header, performs applicable network layer services, selects a route for the packet from its routing tables, applies a tag, and forwards the packet to the next hop tag switch.
4. The tag switch receives the tagged packet and switches the packet based **solely** on the tag.
5. The packet reaches the tag edge router at the egress point of the network, the tag is stripped off and the packet delivered as usual.

Tag Locations

- in the Layer 2 header (e.g., in the VCI field for ATM cells)
- in the Layer 3 header (e.g., in the flow label field in IPv6) or
- in between the Layer 2 and Layer 3 headers

Creating tags

Since tag switching decouples the tag distribution mechanisms from the data flows
- the tags can be created:

- when the first traffic is seen to a destination or
- in advance - so that even the first packet can immediately be labelled.

See also: Multiprotocol Label Switching (MPLS) and Generalized Multi-Protocol Label Switching (GMPLS)

Routers do more than routing

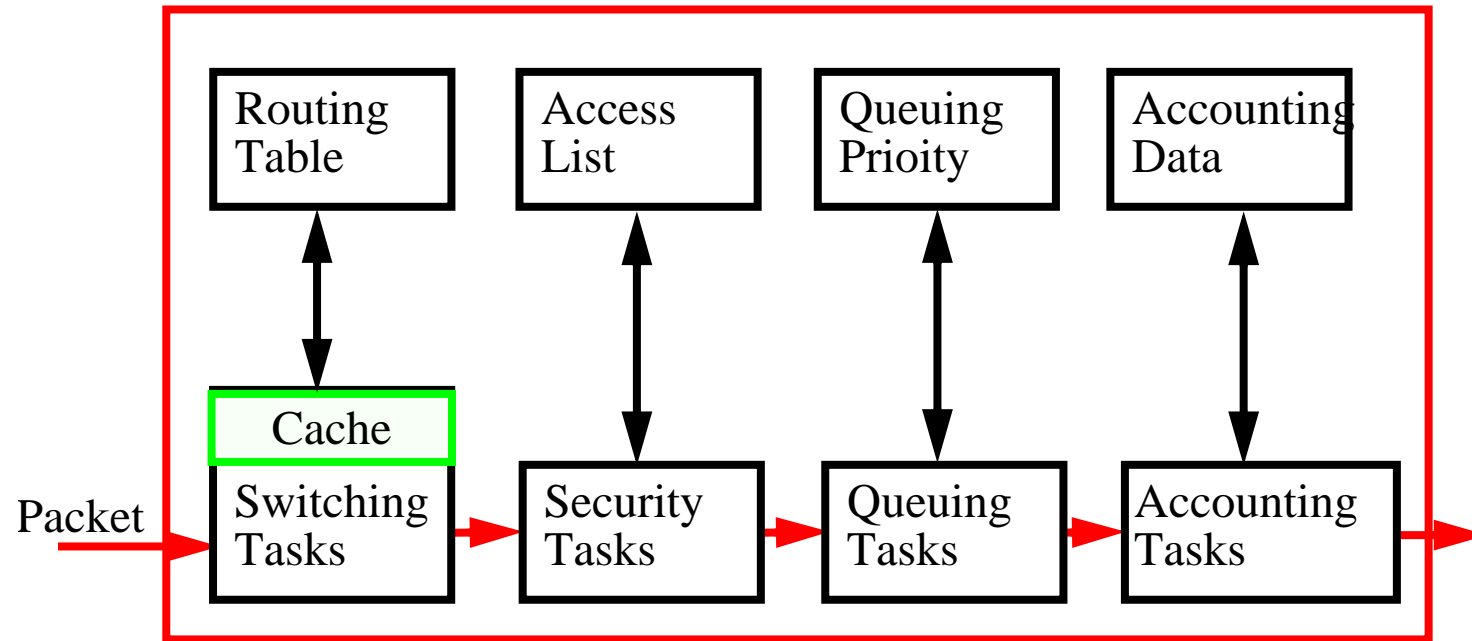


Figure 90: Basic steps in Routing

Earlier we have looked at the routing step, but today much of the excitement is in the details of the other functions. For example, in order to support various QoS features you might want to use more sophisticated queue management: Weighted Round Robin, Fair Queuing, Weighted Fair Queuing, Random Early Detection (RED), Weighted RED,

Summary

This lecture we have discussed:

- IP routing
 - Dynamic routing protocols
 - RIP, OSPF, BGP, CIDR
 - Cisco's NetFlow Switching and Tag Switching
- NAPs and other interconnect points

References

- [45] J. Hawkinson and T. Bates, “Guidelines for creation, selection, and registration of an Autonomous System (AS)”, IETF RFC 1930, March 1996 <http://www.ietf.org/rfc/rfc1930.txt>
- [46] C. Hedrick, “Routing Information Protocol”, IETF RFC 1058, June 1988
<http://www.ietf.org/rfc/rfc1058>
- [47] G. Malkin, “RIP Version 2: Carrying Additional Information”, IETF RFC 1388, January 1993 <http://www.ietf.org/rfc/rfc1388>
- [48] G. Malkin, “RIP Version 2”, IETF RFC 2453, November 1998
<http://www.ietf.org/rfc/rfc2453>
- [49] Charles L. Hedrick, “An Introduction to IGRP”, Cisco, Document ID: 26825, Technology White Paper, August 1991
<http://www.cisco.com/warp/public/103/5.html>
- [50] “Enhanced Interior Gateway Routing Protocol”, Cisco, Technology White

Paper, Document ID: 16406, April 19, 2005

<http://www.cisco.com/warp/public/103/eigrp-toc.html>

[51] J. Moy, “OSPF Version 2”, IETF RFC 2328, April 1998

<http://www.ietf.org/rfc/rfc2328.txt>

[52] “OSPF Design Guide”, Cisco, Technology White Paper,
Document ID: 7039, August 26, 2004

<http://www.cisco.com/warp/public/104/1.html>

[53] K. Lougheed and Y. Rekhter, “A Border Gateway Protocol 3 (BGP-3)”,
IETF RFC 1267, October 1991

<http://www.ietf.org/rfc/rfc1267.txt>

[54] Y. Rekhter and T. Li (editors), “A Border Gateway Protocol 4 (BGP-4)”,
IETF RFC 1654, July 1994

<http://www.ietf.org/rfc/rfc1654.txt>

[55] Y. Rekhter and T. Li (editors), “A Border Gateway Protocol 4 (BGP-4)”,
IETF RFC 1771, March 1995

<http://www.ietf.org/rfc/rfc1771.txt>

[56] John W. Stewart III, *BGP4: Inter-Domain Routing in the Internet*,
Addison-Wesley, 1999, ISBN: 0-201-37951-1

- [57] Bassam Halabi, *Internet Routing Architectures*, Cisco Press, ISBN 1-56205-652-2
- [58] R. Hinden (Editor), “Applicability Statement for the Implementation of Classless Inter-Domain Routing (CIDR)”, IETF RFC 1517, September 1993
<http://www.ietf.org/rfc/rfc1517.txt>
- [59] Y. Rekhter and T. Li (Editors), “An Architecture for IP Address Allocation with CIDR”, IETF RFC 1518. September 1993 <http://www.ietf.org/rfc/rfc1518.txt>
- [60] V. Fuller, T. Li, J. Yu, and K. Varadhan, “Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy”, IETF RFC 1519, September 1993 <http://www.ietf.org/rfc/rfc1519.txt>
- [61] Y. Rekhter and C. Topolcic, “Exchanging Routing Information Across Provider Boundaries in the CIDR Environment”, IETF RFC 1520, September 1993 <http://www.ietf.org/rfc/rfc1520.txt>
- [62] R. Chandra, P. Traina, and T. Li, “BGP Communities Attribute”, IETF RFC 1997, August 1996 <http://www.ietf.org/rfc/rfc1997.txt>

[63] E. Chen and T. Bates, “An Application of the BGP Community Attribute in Multi-home Routing”, IETF RFC 1998, August 1996

<http://www.ietf.org/rfc/rfc1998.txt>

[64] Iljitsch van Beijnum, web site <http://www.bgpexpert.com/>, last modified April 28, 2005 12:00:00 PM

[65] Iljitsch van Beijnum, *BGP*, O’Reilly, 1st Edition September 2002, ISBN 0-596-00254-8

[66] “Internet Exchange Points: Their Importance to Development of the Internet and Strategies for their Deployment: The African Example”, Global Internet Policy Initiative (GIPI), 6 June 2002 (revised 3 May 2004),

<http://www.internetpolicy.net/practices/ixp.pdf>

[67] Cesar Olvera Morales, Jordi Palet Martinez, Alvaro Vives, Alain Baudot, Carlos Parada, Raffaele D’Albenzio, Mario Morelli, David Fernandez, and Tomás de Miguel, Specification of the Internal Network Architecture of each IX Point, Deliverable D2.1, Euro6IX: European IPv6 Internet Exchanges

Backbone Project, IST-2001-32161, version 4.4, 30 July 2002

http://www.euro6ix.org/Reports/public/euro6ix_public_d2_1_v4_4.pdf

[68]