Routing Protocols Classification

Petr Grygárek

Classification criteria

- Internal (IGP) / External (EGP)
  - number of handled routes
  - possibilities of routing politics specification
- Convergence Time
- Distance-vector / Link-state
- Classfull/Classless
- Metric used
- Support for load balancing (equal or nonequal cost)

Classfull/Classless Routing Protocols

- Classful routing protocol does not send subnet masks in updates, presumes all networks are of A/B/C class
- Classless routing protocol can carry subnet mask information in route advertisements
  - VLSM (RFC 1009)
    - Various prefix lengths (subnet masks) for individual subnets of the same network
      - But resulting addresses need to be unique
    - Subnets can be further subnetted for more efficient IP address allocation
  - Control of Route Summarization
    - Smaller routing tables

VLSM Addressing

Distance-vector Routing Algorithms and Protocols

Distance vector algorithms

Why “distance vector” ?

1. Routes are advertised as vectors (vector has length (=metric) and direction (=next-hop))
2. Neighboring routers exchange their „vectors of distances“ to known networks (i.e. routing tables)

DV Algoritms – Common Characteristics

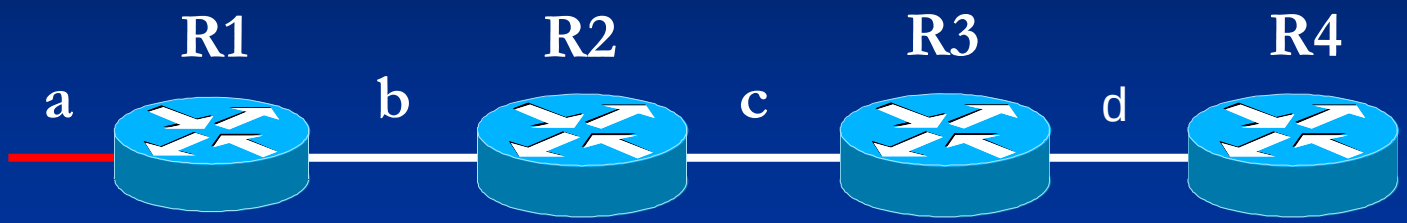
- Principle: Distributed Bellman-Ford (Ford-Fulkerson) algorithm
- Routing tables constructed only based on information from neighboring routers – „routing by rumor“
- Hop-by-hop updates
- Periodic updates
  - (10-90 secs typically, need to balance convergence time vs. load)
- Broadcast (sometime multicast) updates
  - identity of neighbors not known
- Full routing table updates
  - (except applying Split horizon rule)

DV Algorithms - Timers

- Update timer
- Invalid (expiration) timer
  - maintained separately for every route
  - typically 3-7 update timer periods. Reset every time a route is heard about.
  - if expired, route is marked and propagated as unaccessible (but still used by router itself).
- Flush (garbage collection) timer
  - after expiration route marked as invalid (by invalid timer) are removed from routing table
- Holddown timer (will be discussed next)

Timers have to be set consistently across whole routing domain

Convergence Time (without triggered updates)



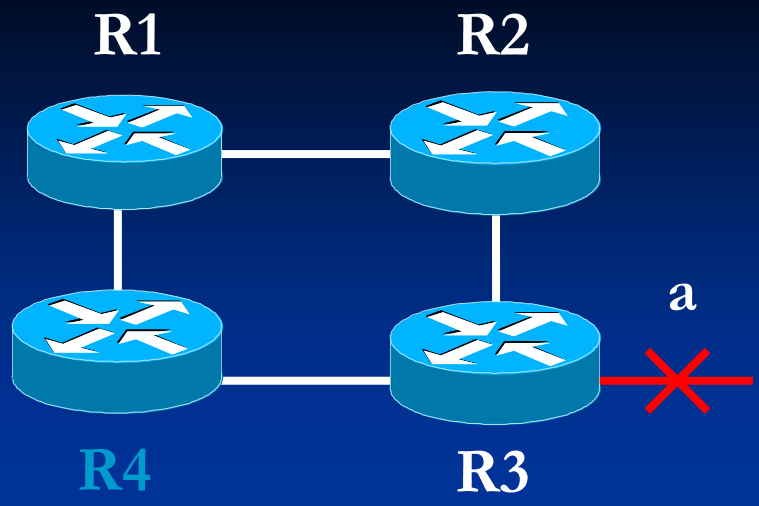
Network **a** just goes up, assume routing update period 30s

- R2 learns about network **a** after 30s (max)
- R3 learns about network **a** after 60s (max)
- R4 learns about network **a** after 90s = 1.5 min (max)

Holddown timer

- Defined as follows: When a previously failed route is received again (with worse metric), ignore that new information for a time equal to the holddown timer.
  - Commonly, routes from the router that was a next-hop before route failed are accepted if propagated with original metric
- Stated another way: if the route's metrics gets worse, do not accept other routes for a while
- Prevents routes to failed networks being re-introduced by routers that have not noticed the failure yet.
- Helps to combat against count-to-infinity problem
- Example topology: triangle with a just failing stub network in one vertex

Holddown timer usage example



Holddown timer applied on R4:

- R3 reports network **a** as unreachable (using triggered update)
  - Holddown timer started
- R1 offers route to **a** via 3 hops
  - do not belive R1 if holddown timer in progress !
  (maybe R1 is still not informed about network **a** unreachability)

Triggered updates (Flash updates)

- If a metric of a route changes (up or down), information is sent immediately without waiting for the next update period
- Greatly reduces Counting-to-Infinity problem (but does not eliminate it completely)
- Possibility to transmit only changed information (not whole routing table)
- 1-5s pause between triggered updates
  - (reduces broadcast storms and flapping)
- Implemented in both RIP versions (RFC 2091), IGRP, …

Split Horizon

- Solves problem of routing loops occuring due to updates passing each other over a single link
  - Count to infinity problem
- Poisson reverse – information not only filtered by Split horizon, but intentionally sent with infinite metric
  - larger updates
  - safety against corrupted information

Counting to infinity problem (even with Split Horizon applied)

- Example topology: rectangle of 4 routers and one additional in the corner which fails.
  - Counting to infinity solves a problem of routing loops occuring due to updates passing each other over alternate paths
  - See http://websrv.cs.fsu.edu/reference/itl/labs/countinf/countinf.htm

Route Poissoning

- Little mess in terminology
- Cisco definition: “When a distance vector routing protocol notices that a particular route is no longer valid, it has two choices. One is simply to quit advertising about that subnet; the other is to advertise that route, but with an infinite metric, signifying that the route is bad. Route poisoning calls for the second of these options, which removes any ambiguity about whether the route is still valid”

DVA – Advanced issues

- Passive Interfaces
- Active and Passive participants

- Problem of update synchronization – periodic network congestion. Possible solution: timing jitter of update periods

- Unicast updates: if neighbor's IP address configured explicitly.
  - Reduction of broadcasts, but requires to maintain list of valid neighbors

- Router may "consume" more than 1 hop
  - (offset lists define consumed hops on interface for incoming/outgoing RIP updates)

- Unnumbered interfaces
- DV algorithms on dial-on-demand circuits

Advantages of DV routing algoritms

- Simple implementation (good interoparability)
- Simple configuration, no complicated planning
- Ease of route filtering (incoming or propagated)

Routing Information Protocol (RIP)

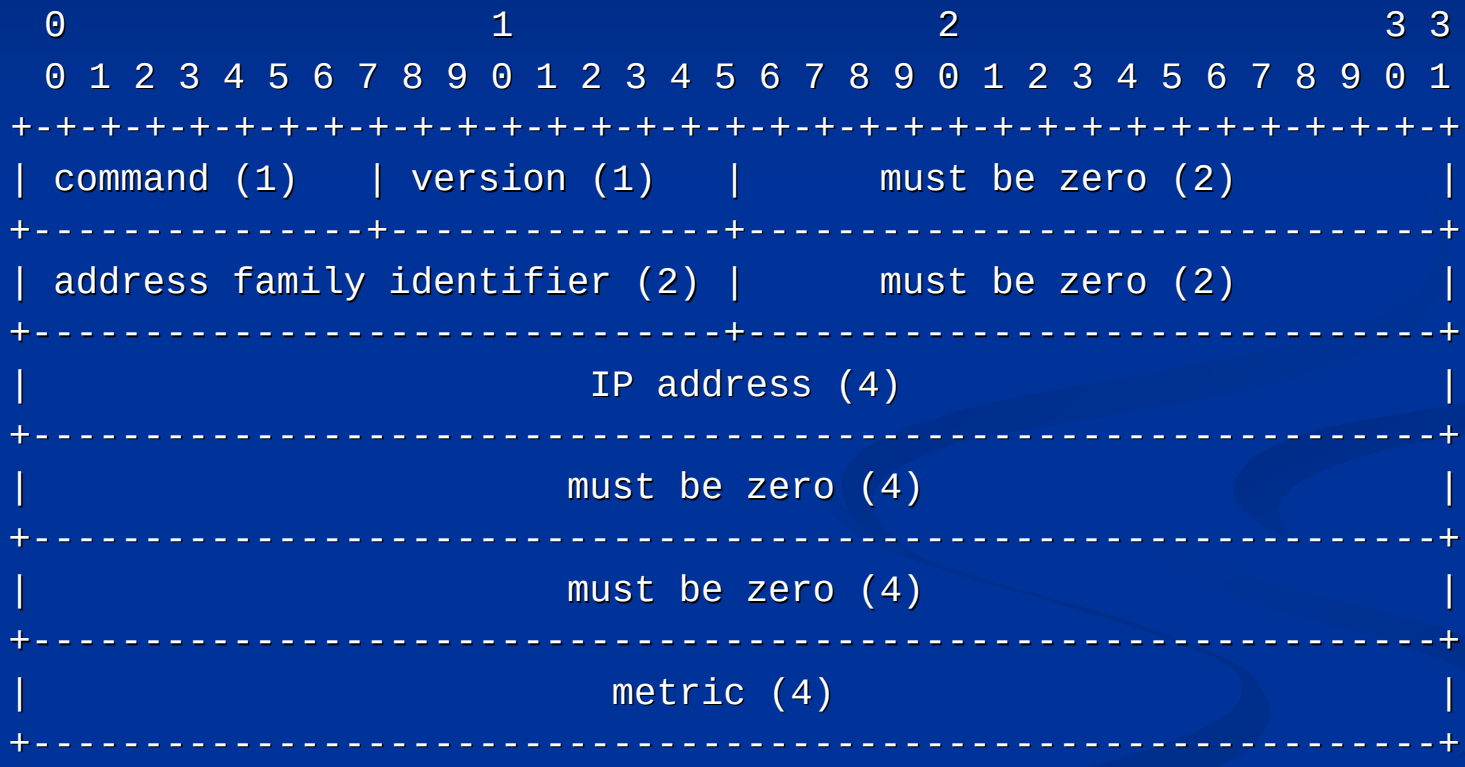
RIP Overview

- A long history - since ARPANET (1969), implemented in various network architectures (IP, IPX, AppleTalk, …)
- Simple metric (hop count)
  - suitable for equal-bandwidth lines and small networks (hop count 16 = infinity)
- Simple implementation and configuration, widespread usage, interoperable implementations
- RIPv1: RFC 1058 (Hedrick, 1988 (!))
- RIPv2: RFC 2453 (+ RFC 1723)
  - RIP versions can be different on different interfaces
- UDP, port 520 (both source and destination)
- Support for equal-cost load balancing (some implementations)

RIP version 1

- Classful routing protocol
- Does not include subnet mask information.
- Automatic summarization at major network boundaries.
- Updates sent as broadcasts by default.
- Next-hop is the sender IP address (source of IP/UDP packet)
- Max 25 entries (update length max. 512B)
- Entry has 5x4B:
  Address Family Identifier, IP address, metric, unused
  (inefficient)

RIPv1 Packet Format



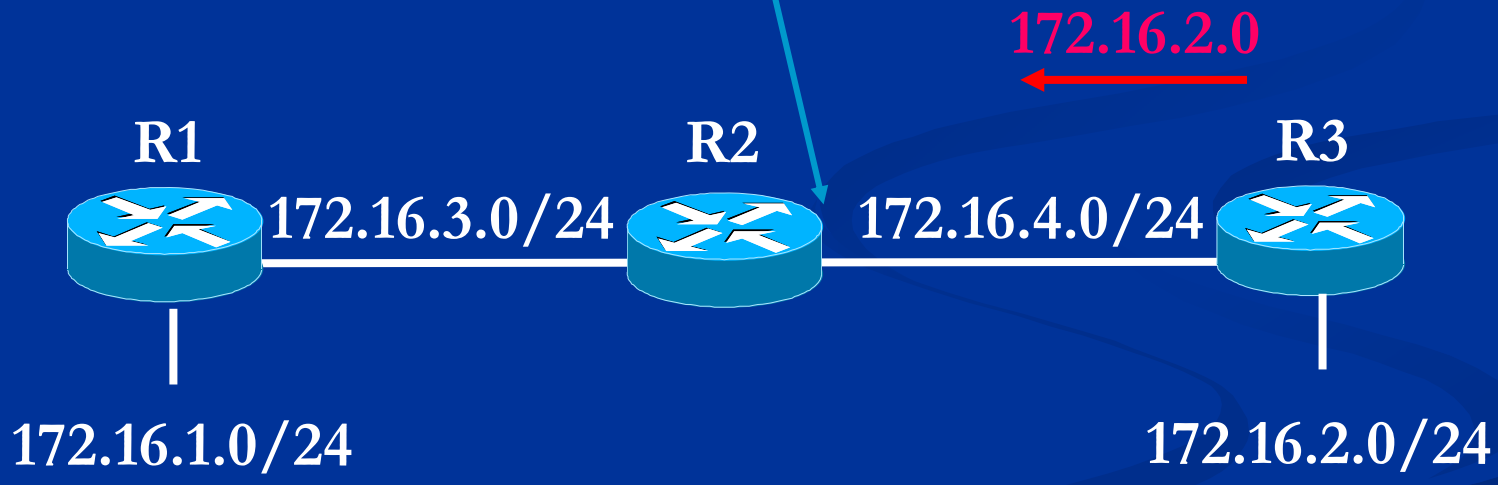
RIP v1 and subnet masks

- RIP is classfull – no masks in routing updates
- Assumes consistent (constant) subnet mask for all subnets of one major (i.e. class-level) network
- If some router's interface is connected to the subnet of a major network and receives a routing update, router will use receiving interface's mask for the advertised subnet. Othewise the class default mask is used

Handling of constant subnet mask in classful routing protocols

172.16.2.0, but with what mask ?

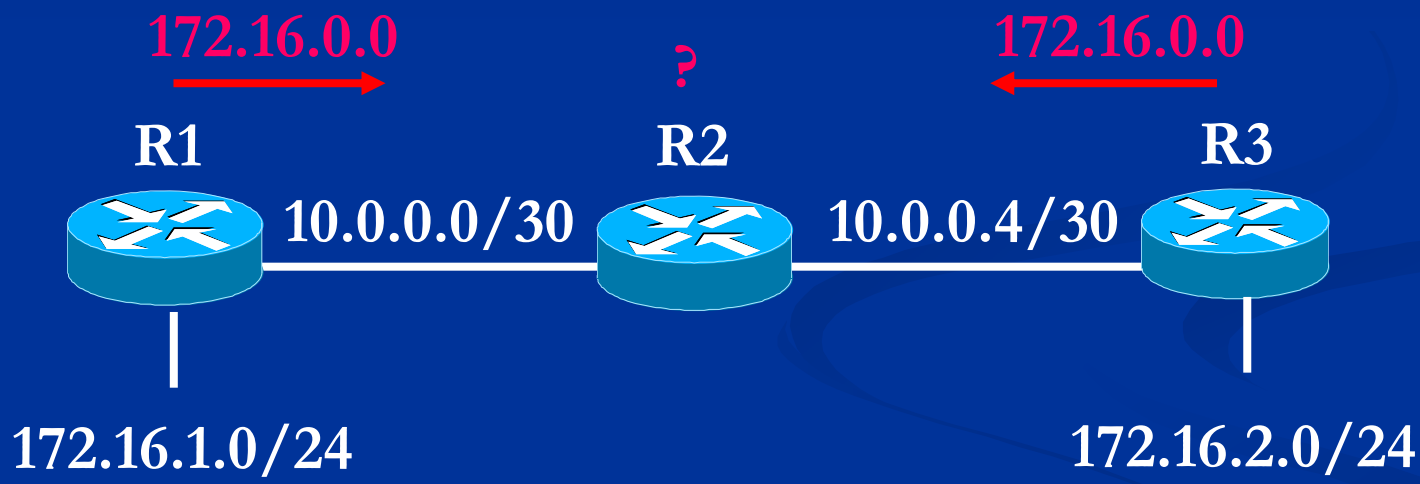
I will count with mask placed on the receiving interface !



Aggregation and subnet continuity requirement

- Subnets are advertised only on interface belonging to the same major network as those subnets
  - Subnet summarization (subnet hiding) at the boundary of classful networks
- The previous implies a need of contiguous subnets

Discontiguous subnets problem



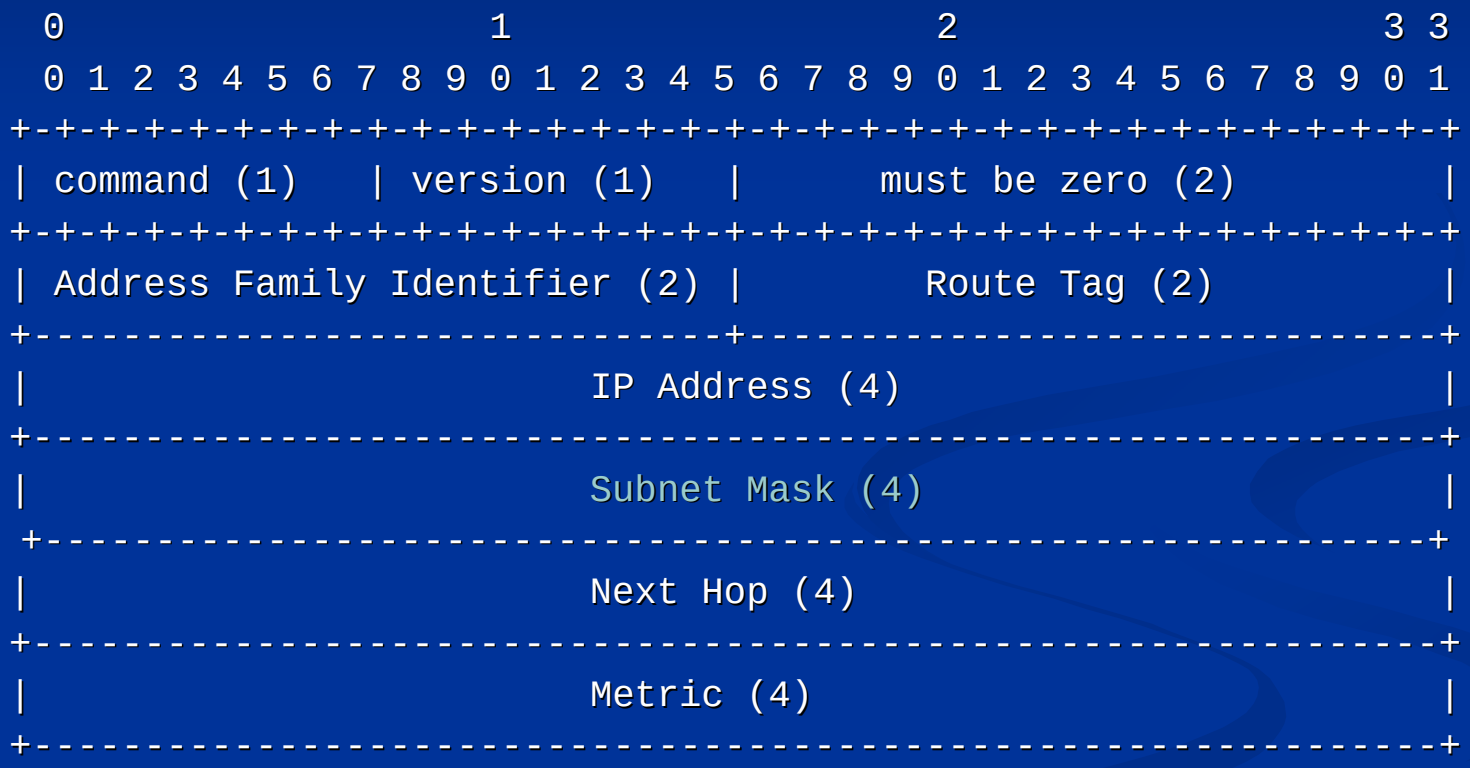
RIPv1 Commands

- Update Request (on router bootup)
  - Full table request/specific route request
- Update (solicitated or unsolicitated)

RIP version 2

- Classless routing protocol-includes subnet masks in updates
- Allows disabling of automatic summarization at major network boundaries (do it for discontinuous subnets !).
- Allows updates as multicasts (224.0.0.9)
- Support for route tags (marking of external routes)
- Support for authentication
- Explicitly defined next-hop for each propagated route
  - 0.0.0.0 if matches the update sender's IP address
  - Useful for route redistribution between routers on the shared network

RIP v2 Packet Format



RIP v2 Authentication

- Cleartext password (RFC standard)
- MD5 hash (Cisco proprietary)
  - (RFC 2082 – RIP-2 MD5 Authentication)
- Authentication info appended right behind RIP header as route entry with AFI=FFFFh and Route Tag acting as Authentication Type. Then 16 octets of Authentication Data (password/hash) follows.

RIP Timers

- Update timer 30 s
- Invalid timer: 6x30s=180 s
  (starts when no info about route for 180s)
- Flush Timer: 240s: After Invalid timer expires, route metric is set to 16; after 240s it is flushed out of routing table completely
- Hold down timer (if Holddown feature implemented)

Default route in RIP

- propagated normally as 0.0.0.0 network
- If a router receives multiple defaults, it chooses the best based on smaller metric (or load balances)

Interior Gateway Routing Protocol (IGRP)

- Cisco proprietary
- Hop limit 255
- Unequal-cost load balancing
- Composite metric
- Slower timing

Labs

- Discontiguous subnets – RIP1
  chain of 3 routers, side routers have ethernets addressed with subnets of class C networks, connecting lines are class A.
- Migration to RIP2
  - router rip
    version 2
    no auto-summary
- Propagation of default route (from one of side routers)
  - router rip
    default-information originate
    (redistribute static // on some IOS versions)
- RIPv2 authentication
  - Key chain jméno
    Key 1
    Key-string heslo
    interface s1
    ip rip authentication key-chain jméno
    ip rip authentication mode md5
- (Connect into triangle), look at multiple routes in routing table. Set hop-offset on some interface
  - router rip
    offset-list <route-prefix-ACL#> in | out <hopcount-offset> <interface>
- route filtering
  - router rip
    distribute-list <ACL#> in | out <interface>