L40 - RIP

RIP, RIPv2

Routing Information Protocol

Internet´s First IGP

Agenda

- **RIP Principles**
- **RIP Problems**
- **RIP Workarounds**
  - maximum hop count
  - split horizon
  - poison reverse
  - triggered update
  - hold down
- **RIP version 2**

RIP - Routing Information Protocol

- **RIP is an Interior Gateway Protocol (IGP)**
  - due to inherent administrative traffic, RIP suits best for smaller networks (autonomous systems, routing areas)
  - routing decisions are based upon hop count measure
- **RIP was initially released as part of BSD 4.2 UNIX**
  - hence RIP got wide-spread availability
- **UNIX workstations supporting RIP**
  - can determine all routers in their LAN themselves
  - do not need a default-router entry
- **RIP is specified in RFC 1058**
  - RFC category „historic“

Basic RIP Table Entries

NET-ID	VECTOR	DISTANCE	INT	TIME
96.0.0.0	197.23.5.47	7	P1	time_2
126.0.0.0	148.12.77.3	3	P2	time_1
148.12.0.0	-	0	P2	-
197.23.5.0	-	0	P1	-
0.0.0.0	197.23.5.47	1	P1	Default Route

NET-ID ..... IP address network part of the target network
VECTOR ..... IP-address of the nearest router on the way to this network (= next hop)
DISTANCE .... number of hops (=metric)
INT ..... Label of the outgoing physical router-port
TIME ..... timestamp of the last routing update for this entry; important for the aging mechanism

RIP Principles 1

- **RIP is a Distance Vector Protocol**
- **after booting the non-volatile configuration-memory tells a RIP router to which networks it is directly connected**
- **this information is loaded into the routing table**
- **basically the routing table contains**
  - the net-ID of the directly connected networks
  - and the associated distance (in hops) to them
    directly connected networks have hop-count = 0

RIP Principles 2

- **then, this routing table is distributed periodically (every 30 seconds) to all connected networks = routing update**
  - using a broadcast MAC-frame containing
  - an IP-broadcast datagram containing
  - an UDP-datagram with port number 520
  - metric entries of the routing table will be risen by the distance of the interface where transmitted-> in case of RIP -> distance is one -> hop count
- **directly reachable routers**
  - receive this message, update their own routing tables,
  - and hence generate their own routing updates reflecting any corresponding modifications

RIP Principles 3

- **after a specific time**
  - all routers know about all network addresses of the whole network
- **if different routing updates (from different routers) contain the same net-ID**
  - then there are redundant paths to this network
  - only the path with the lowest hop-count is stored in the routing-table
  - on receiving equal hop counts, the net-ID of the earlier one will be selected (and all other associated data)
- **hence, between each two networks exists exactly one active path**

RIP Principles 4

- **all routing table‘s net-IDs are periodically refreshed by routing-update messages**
- **if a routing update tells a better metric than that one currently stored in the table**
  - the routing table must be updated with this new information
  - this update does not take care about if the sender of this routing-update is also the router which is currently selected as next hop
  - "good news“ are quickly adapted
  - RIP trusts good news from any source ("trusted news")

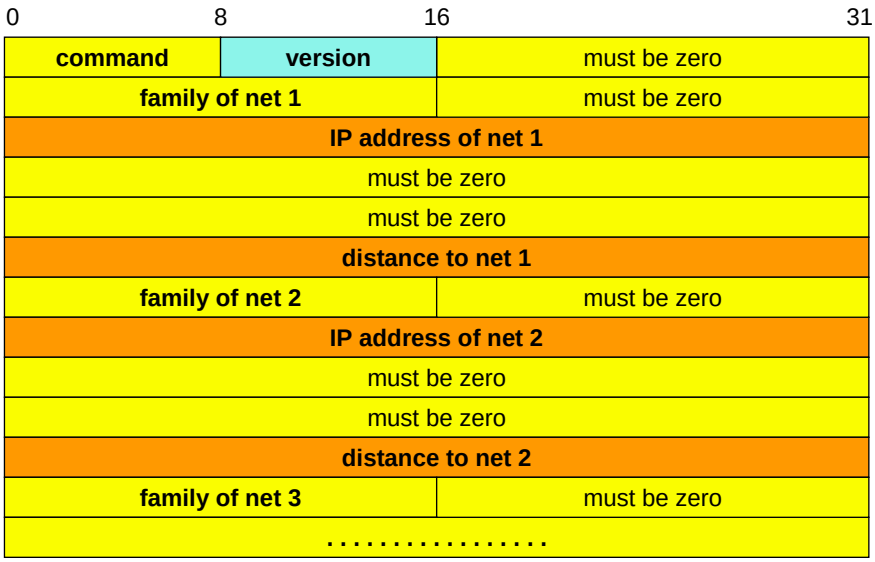
RIP Principles 5

- **if a routing update tells a worse metric than that one currently stored in the table**
  - the routing table must be updated with this new information if the sender of this routing-update is the next-hop router for this network
    - that is: the actual VECTOR in the table is identical with the source address of the routing-update
  - routing-updates from other routers than that one currently registered in the table are ignored
- **summary: routing-updates with worse metric is only relevant if it comes from that router mentioned in the actual table entry**

RIP Principles 6

- **when a routing table entry is not refreshed within 180sec**
  - this entry is considered to be obsolete
  - possible reasons: router-failure, network not reachable
  - without special mechanism
    - we have to wait for 180sec at least in order that all routers have consistent routing tables again
  - improvement by a special network-unreachable message
    - which is distributed to all other routers
    - it takes 180sec in the worst case
  - slow adaptation of "bad news“
  - during these 180sec, forwarding of IP datagram's is done according to the routing table !!

RIP Header plus RIP Data



 RIP, v4.6 11

RIP Header and Data Fields

- **command**
  - Request (1): Router or Host requests for a routing update
  - Response (2): response to a request but also used for periodic routing-updates
- **version**
  - version number of the RIP protocol (= 1 for RIP)
- **family of net x**
  - because RIP is not only build for IP (in the case of IP: 2)
- **IP address of net x**
  - IP-address of the announced network x
- **distance to net x**
  - = hop-count to net x

 RIP, v4.6 12

RIP Principles

to give a clearer idea of the RIP method:

- every router holds a directory, pointing to other networks (VECTOR entries) without knowing the exact location of them
- datagram's follow these "signposts" and finally reach their targets
- information about these signposts is based upon rumours (routing-updates)

Creating Routing Tables

Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct

Routing table B

Net	Hops	Hop-ID
2	0	direct
3	0	direct

Situation after booting

L40 - RIP

Creating Routing Tables

Net	Hops
1	1
2	1

Router A: routing update to net 2

values used by router B to actualise its routing table

Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct

Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	0	direct

Situation after distribution of A's routing update

 RIP, v4.6 15

Creating Routing Tables

Router B: routing update to net 2

Net	Hops
1	2
2	1
3	1

Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	1	2.B

Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	0	direct

Situation after distribution of B's routing update

 RIP, v4.6 16

Routing Updates and Stable States

Router A: routing update to net 2

Net	Hops
1	1
2	1
3	2

Net	Hops
1	2
2	1
3	1

Router B: routing update to net 2

Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	1	2.B

Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	0	direct

Routing tables keep stable state

Agenda

- RIP Principles
- RIP Problems
- RIP Workarounds
  - maximum hop count
  - split horizon
  - poison reverse
  - triggered update
  - hold down
- RIP version 2

RIP-Problems

- **slow convergence**
  - due to 30s routing-update interval and 180s aging timeout
- **routing-loops cannot be safely detected**
  - because routing updates are obligatory ("Trusted Information Principle" of RIP)
- **so, failures and rooting-loops keep large networks inconsistent for a long period of time**
  - datagram's circle around along redundant paths
  - "Count to Infinity" -problem

Count to Infinity 1

Router A: routing update to net 2

Net	Hops
1	1
2	1
3	2

Net	Hops
1	2
2	1
3	1

Router B: routing update to net 2

Routing table A

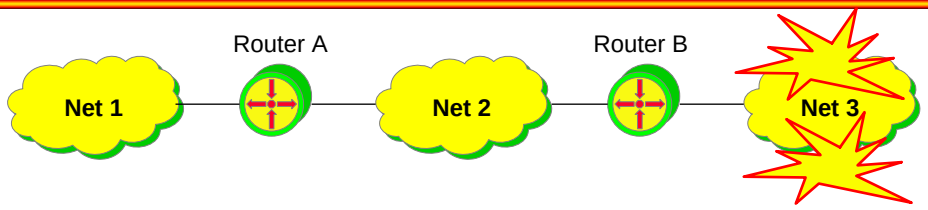
Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	1	2.B

Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	0	direct

Initial situation

Count to Infinity 2



Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	1	2.B

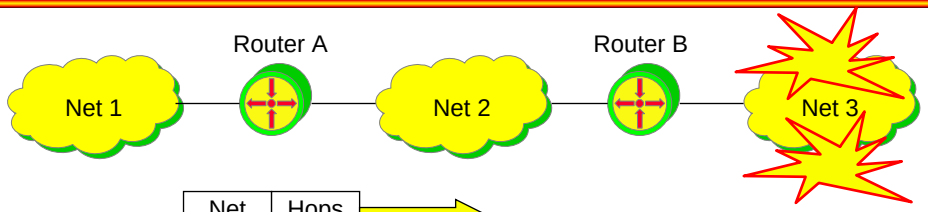
Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	?	?

Failure: net 3 unreachable

 RIP, v4.6 21

Count to Infinity 3



Router A: routing update to net 2

Net	Hops
1	1
2	1
3	2

Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	1	2.B

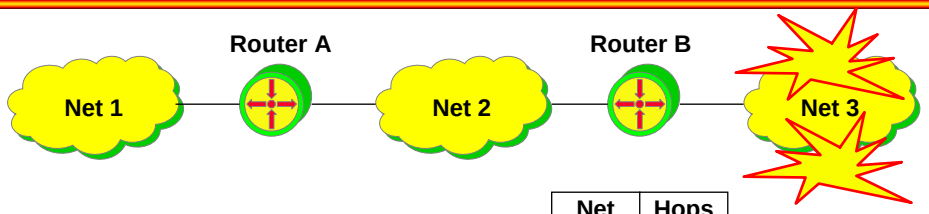
Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	2	2.A

Situation after sending A's routing update

 RIP, v4.6 22

Count to Infinity 4



Net	Hops
1	2
2	1
3	3

Router B: routing update to net 2

Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	3	2.B

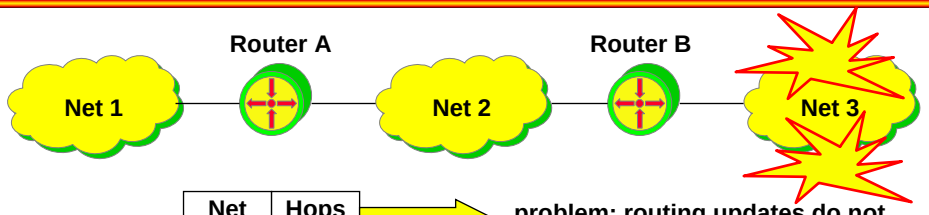
Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	2	2.A

Situation after sending B's routing update

 RIP, v4.6 23

Count to Infinity 5



Router A: routing update to net 2

Net	Hops
1	1
2	1
3	4

problem: routing updates do not contain explicit VECTOR-information !!

(Vector is given only implicitly through source address)

Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	3	2.B

Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	4	2.A

Situation after sending A's next routing update

 RIP, v4.6 24

Solutions against Count to Infinity

- **problem: good news are distributed faster than bad news**
  - the information of a path with lower hop-count is distributed every 30 seconds
  - the information of a network-failure is distributed at least after 180 (or later)
- **remedy:**
  - Maximum Hop Count
  - Split Horizon
  - Poison Reverse
  - Triggered Update
  - Hold Down

Agenda

- **RIP Principles**
- **RIP Problems**
- **RIP Workarounds**
  - maximum hop count
  - split horizon
  - poison reverse
  - triggered update
  - hold down
- **RIP version 2**

Maximum Hop Count

- **maximal distance between each two subnets is limited to 16**
  - hop count between two end-systems cannot exceed 15
- **a DISTANCE-value of 16 in the routing-table means that the corresponding network is not reachable**
  - using hop count = 16 in a routing update allows a router to indicate the failure of a network
  - we have not to age out this entry in all routing tables hence waiting at least for 180 s
  - IP-datagram's with a net-ID pointing to such an entry are discarded by the router and additionally a ICMP message "network unreachable" is generated

Agenda

- **RIP Principles**
- **RIP Problems**
- **RIP Workarounds**
  - maximum hop count
  - split horizon
  - poison reverse
  - triggered update
  - hold down
- **RIP version 2**

Split Horizon 1

- **Maximum Hop Count technique alone (counting to 16) does not avoid routing-loops !**
- **to overcome routing-loops and also slow-convergence, Split-Horizon has been introduced**
  - prevents router from sending information about reachability of networks in that direction from where the information originally came
- **exception of this rule:**
  - if the router knows a better path

 RIP, v4.6 29

Split Horizon 2

Router A: routing update to net 2

Net	Hops
1	1

Net	Hops
3	1

Router B: routing update to net 2

Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	1	2.B

Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	0	direct

Routing update to net 2

 RIP, v4.6 30

Split Horizon 3

Router A: routing update to net 1

Net	Hops
2	1
3	2

Net	Hops
1	2
2	1

Router B: routing update to net 3

Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	1	2.B

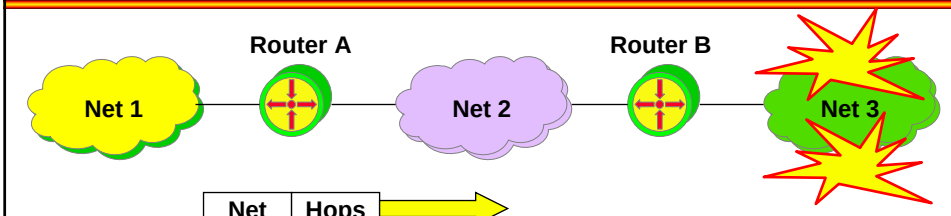
Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	0	direct

Routing update to net1 and net 3

 RIP, v4.6 31

Net 3 unreachable



Router A: routing update to net 2

Net	Hops
1	1

Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	1	2.B

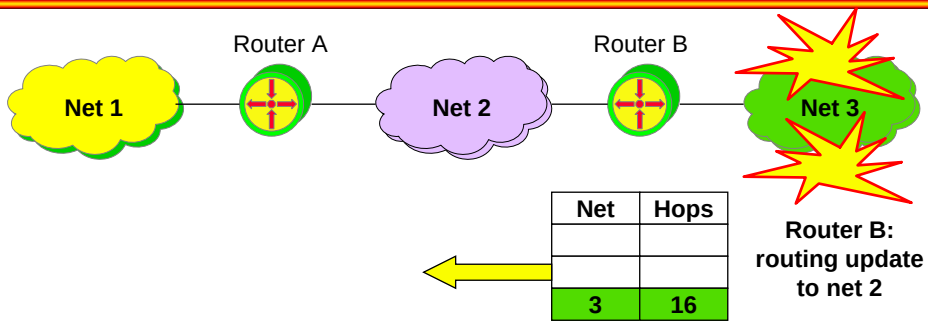
Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	16	-

Situation after sending A's routing update

 RIP, v4.6 32

Net 3 unreachable



Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	16	2.B

Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	16	-

Situation after sending B's routing update

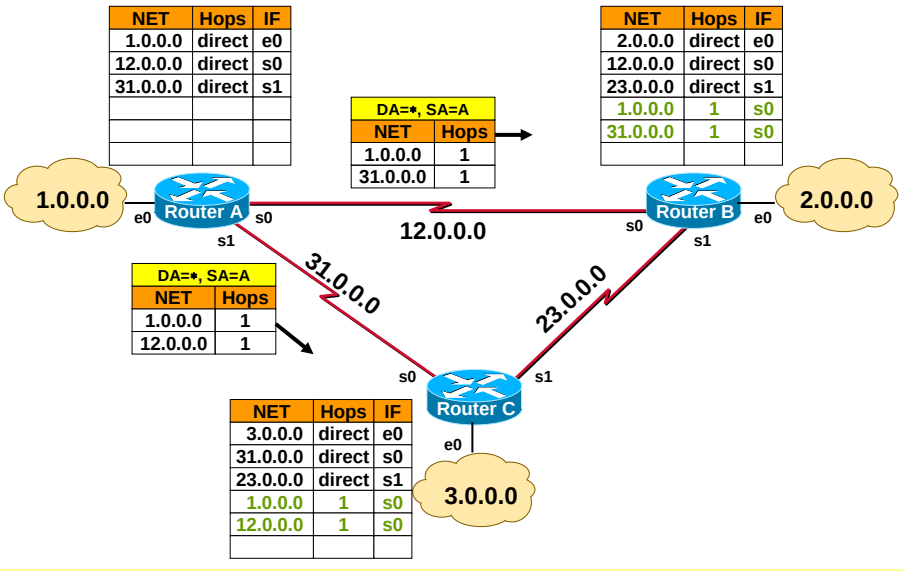
Split Horizon Facts

- **Don't tell neighbour of routes that you learned from this neighbour**
- **That's what humans (almost) always do:**

  Don't tell me what I have told you!
- **Using split horizon a router will not send information about routes he isn't really aware of**
- **the convergence time**
  - is reduced to the time of failure-detection (180 s)
  - note: method of Maximum Hop Count alone would take 16 x 30 s = 480 s in the worst case

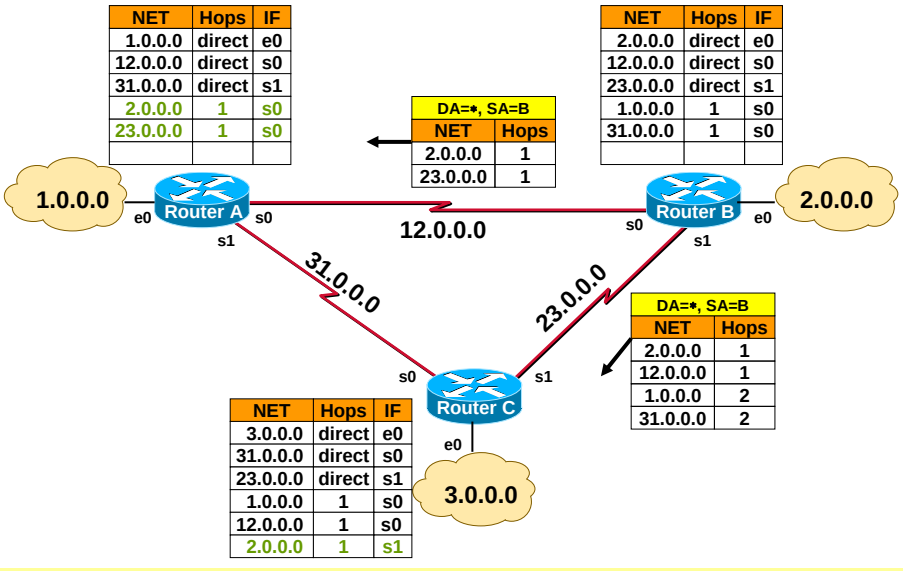
L40 - RIP

RIP At Work (Update Router A)



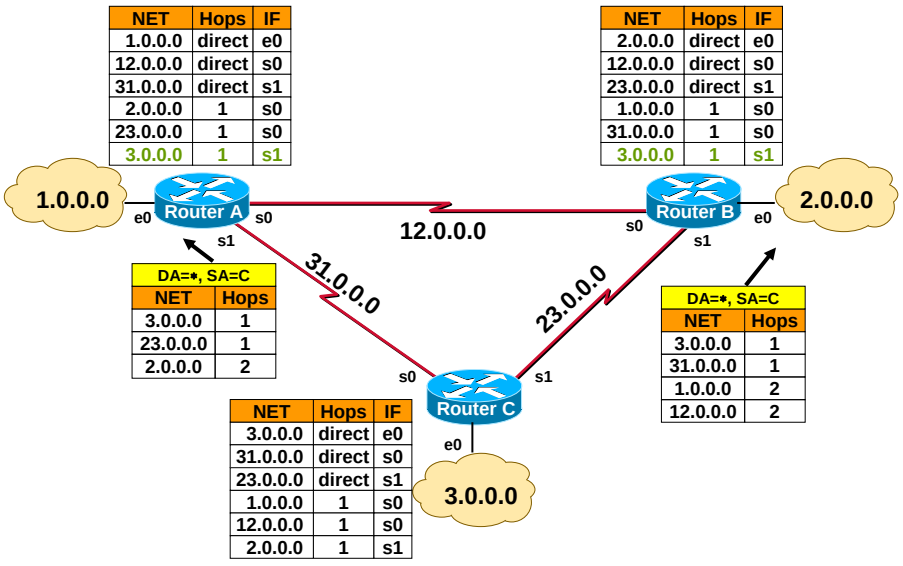
 RIP, v4.6 35

RIP At Work (Update Router B)



 RIP, v4.6 36

RIP At Work (Update Router C)



 RIP, v4.6 37

Agenda

- **RIP Principles**
- **RIP Problems**
- **RIP Workarounds**
  - maximum hop count
  - split horizon
  - poison reverse
  - triggered update
  - hold down
- **RIP version 2**

 RIP, v4.6 38

Poison Reverse 1

- alternative method against routing loops and slow convergence: Poison Reverse
- router sends not-reachability messages (="poison") via routing-updates in the direction from which the information about this network originally came
- so the convergence time is reduced to the time of failure-detection (180 s)
  - method of Maximum Hop Count would take 16 x 30 s = 480 s !

Poison Reverse 2

Router A: routing update to net 2

Net	Hops
1	1
2	16
3	16

Net	Hops
1	16
2	16
3	1

Router B: routing update to net 2

Routing table A

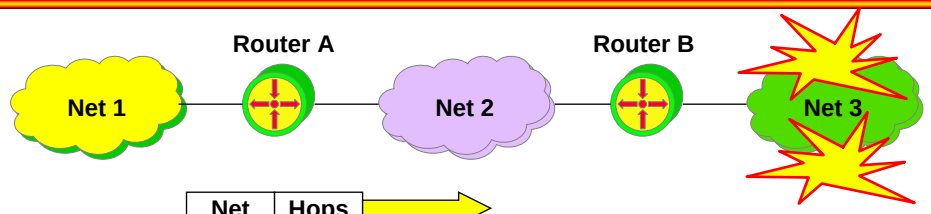
Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	1	2.B

Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	0	direct

Routing update to net1 and net 2

Net 3 unreachable



Router A: routing update to net 2

Net	Hops
1	1
2	16
3	16

Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	1	2.B

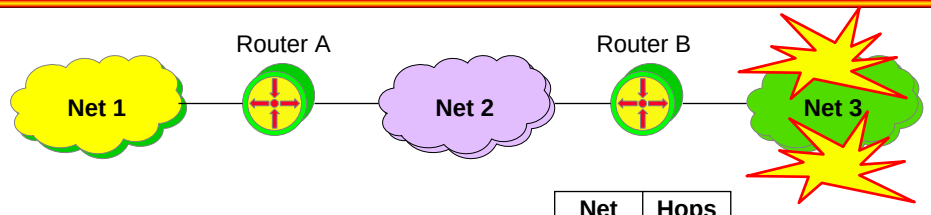
Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	16	-

Situation after sending A's routing update

 RIP, v4.6 41

Net 3 unreachable



Router B: routing update to net 2

Net	Hops
1	16
2	16
3	16

Routing table A

Net	Hops	Hop-ID
1	0	direct
2	0	direct
3	16	2.B

Routing table B

Net	Hops	Hop-ID
1	1	2.A
2	0	direct
3	16	-

Situation after sending B's routing update

 RIP, v4.6 42

Agenda

- **RIP Principles**
- **RIP Problems**
- **RIP Workarounds**
  - maximum hop count
  - split horizon
  - poison reverse
  - triggered update
  - hold down
- **RIP version 2**

Triggered Update

- **Triggered Update has been introduced in order to speed up the convergence time**
- **after notice of a network-failure the router generates immediately a routing-update to indicate this failure**
  - setting hop-count = 16
  - note: the router does not wait for the expiration of the 30 s
- **Triggered Update can also be used when other events occur (e.g. additional links)**
- **Triggered Update without employing additional methods (like Split Horizon) cannot avoid routing-loops**

Agenda

- **RIP Principles**
- **RIP Problems**
- **RIP Workarounds**
  - maximum hop count
  - split horizon
  - poison reverse
  - triggered update
  - hold down
- **RIP version 2**

Hold Down

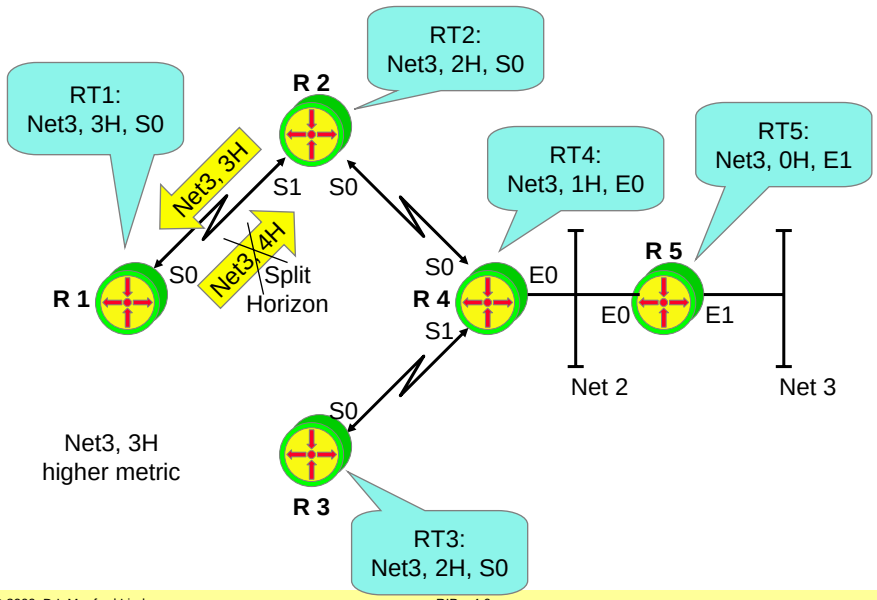
- **Split Horizon is a good means to avoid temporary routing-loops and to improve the convergence time in simple network topologies**
- **complex network topologies require an additional tool to avoid temporary rooting-loops:**

  Hold Down
- **if a router gets information about a network failure, it ignores further informations about that network from other routers for a specific duration of time**
  - typically 240 seconds

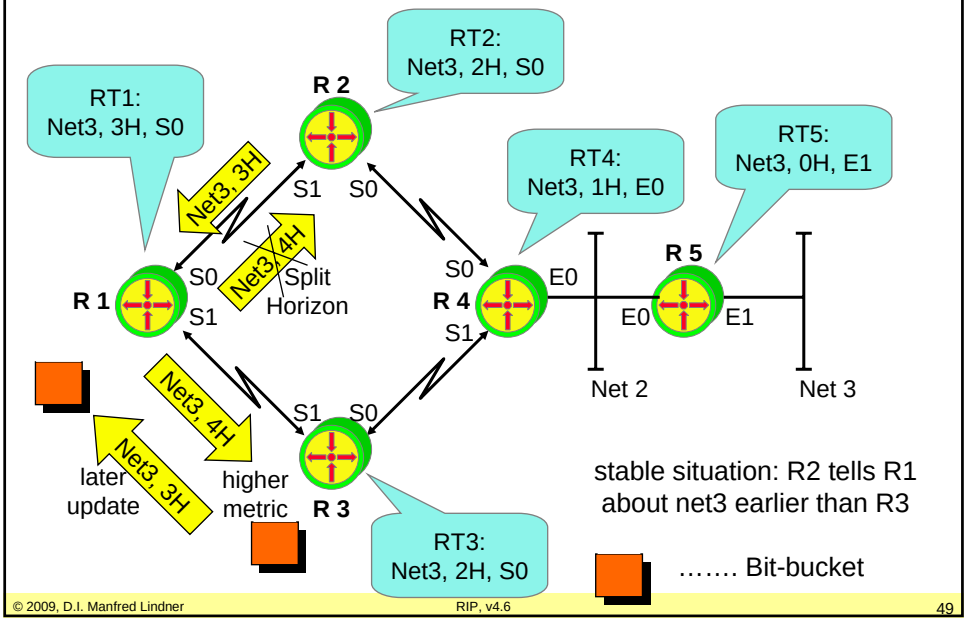
Hold Down

- **basic idea:**
  - network-failure message requires a specific amount of time to spread across the whole network (like a wave)
  - with Hold Down, all routers get the chance to receive the network-failure message
  - steady-state will be awaited to avoid inconsistent routing-tables and rooting-loops
- **disadvantages of Hold Down**
  - longer convergence time...
  - ...can be a drawback at all (in special cases)

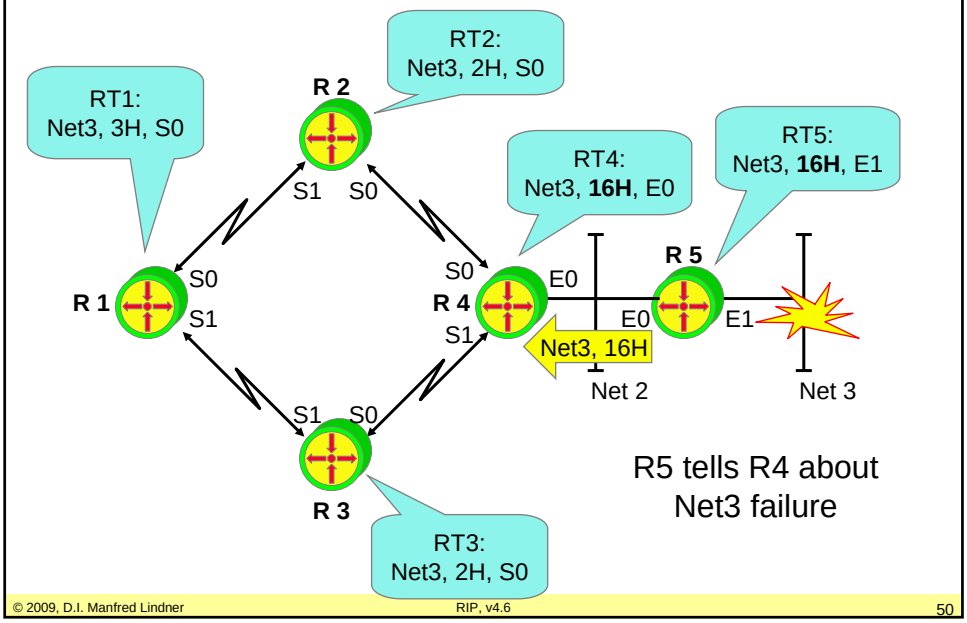
Split Horizon 1



Split Horizon 2

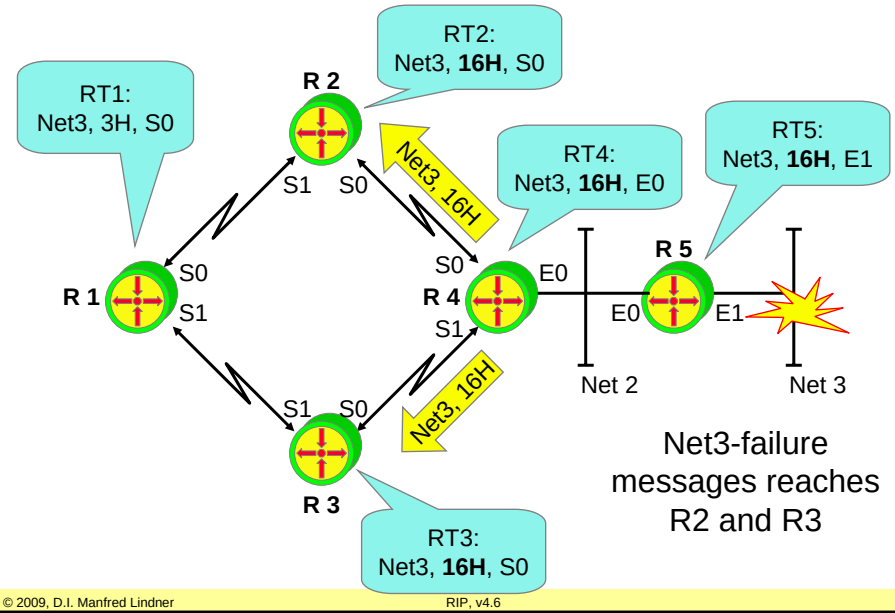


Split Horizon 3

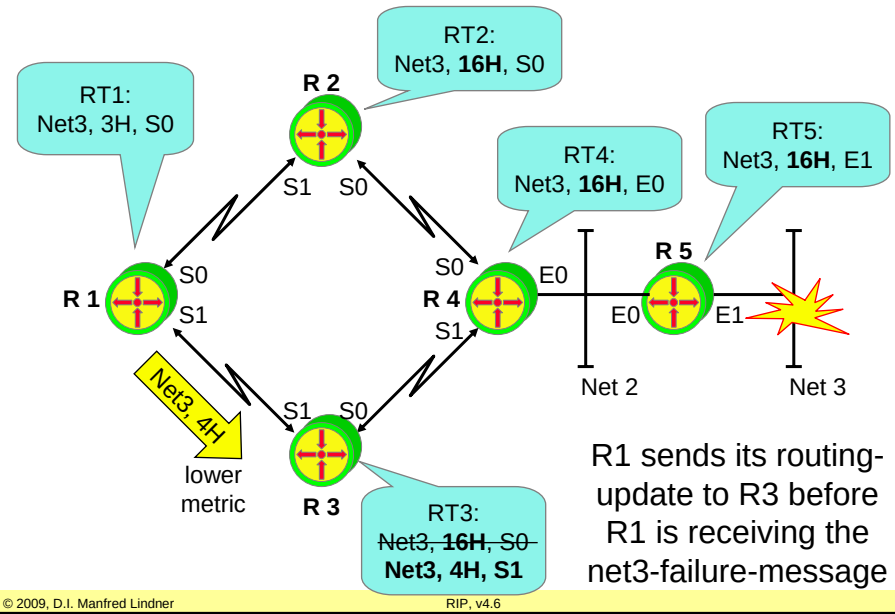


L40 - RIP

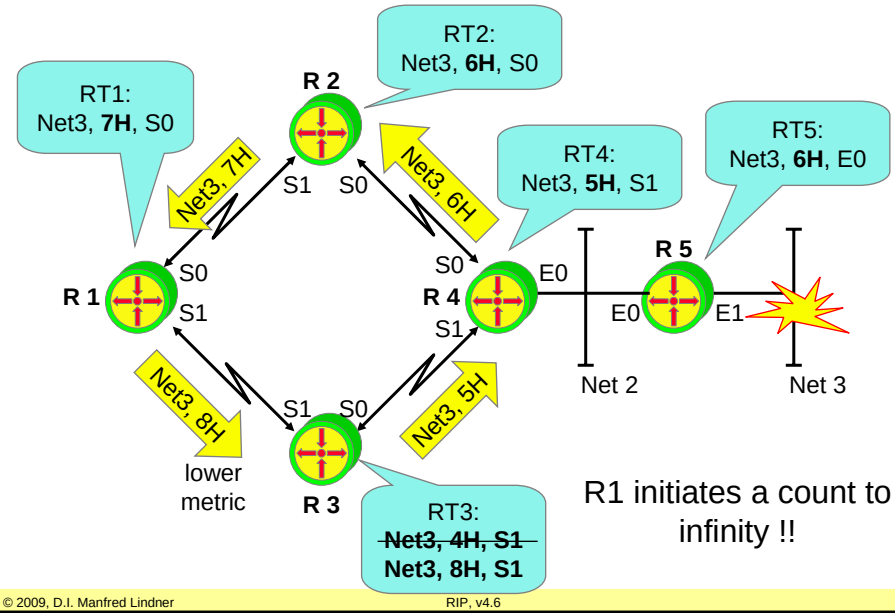
Split Horizon 4



Split Horizon 5

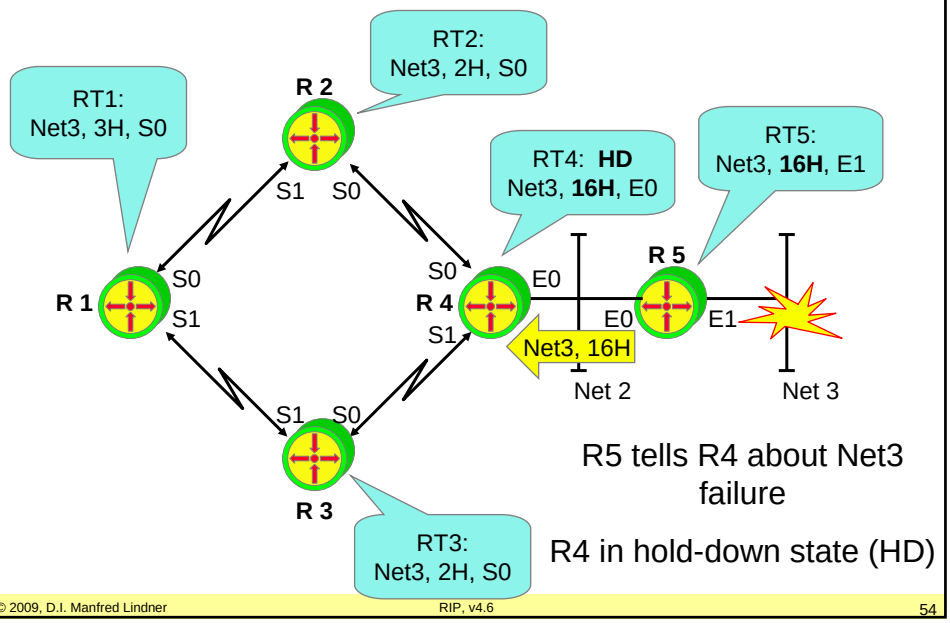


Split Horizon 6



 RIP, v4.6 53

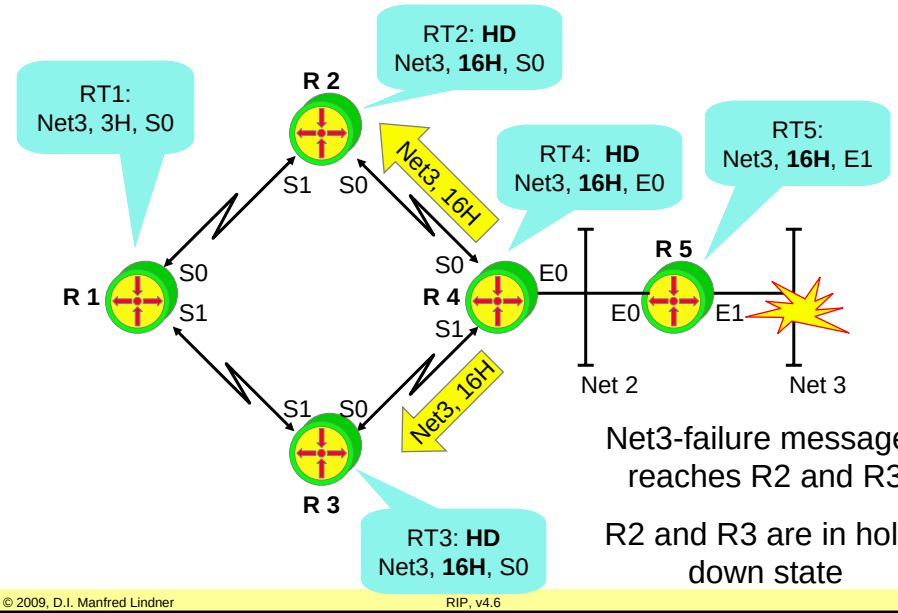
Hold Down 1



 RIP, v4.6 54

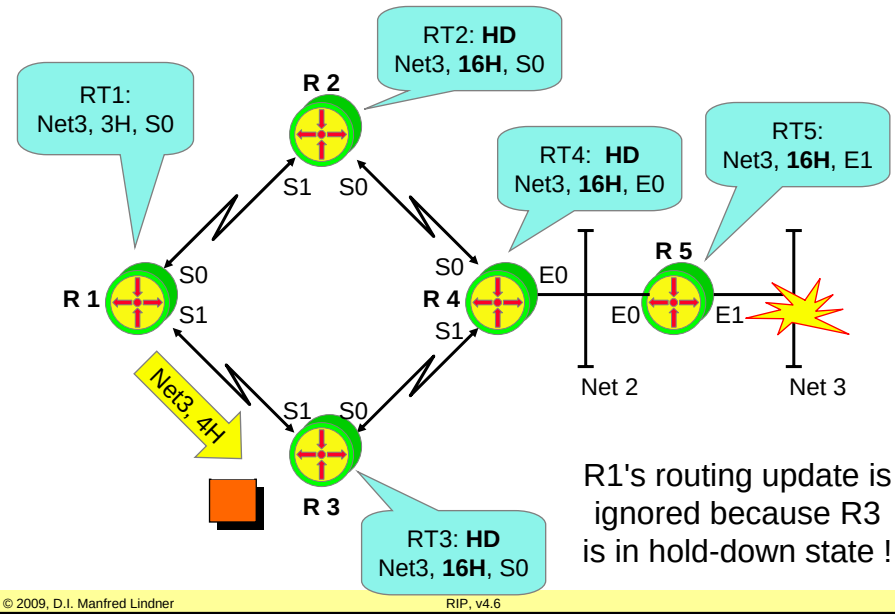
L40 - RIP

Hold Down 2



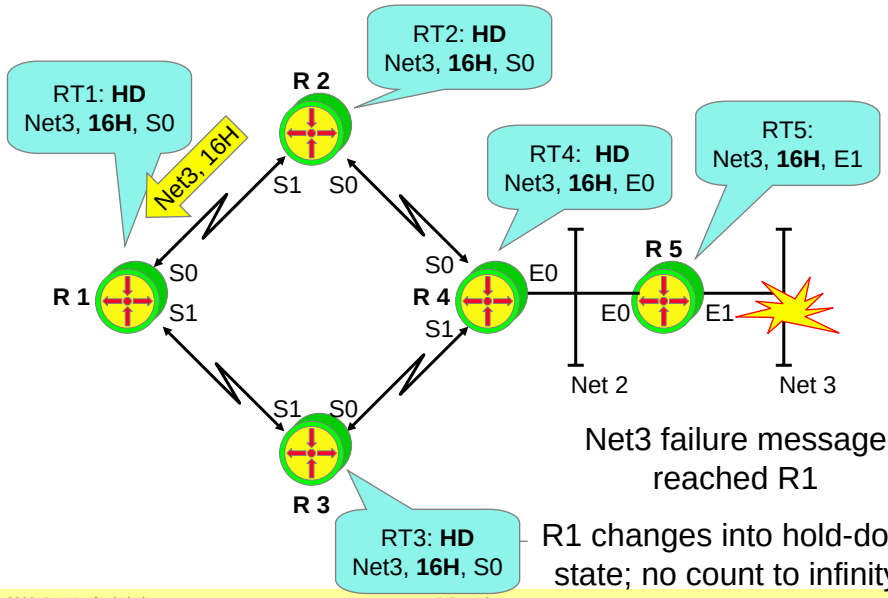
 RIP, v4.6 55

Hold Down 3



 RIP, v4.6 56

Hold Down 4



Net3 failure message reached R1

R1 changes into hold-down state; no count to infinity !

Timer Synchronization / RIP Timers

- **timer synchronization**
  - in case of many routers on a single network processing load might affect update timer
  - routers might get synchronized and collisions occur more often
  - therefore either use
    - External timer or add a small random time to the update timer (30 seconds + RIP_JITTER = 25...35 seconds)
- **RIP timers and their values**
  - UPDATE (30 seconds)
    - period to send routing update
  - INVALID (180 seconds)
    - aging time before declaring a route invalid ("16") in the routing table
  - HOLDDOWN (180 seconds)
    - how long a router will wait before accepting an update with better metric for an invalid route
  - FLUSH (240 seconds)
    - time before a non-refreshed routing table entry is removed

Further RIP-Problems

- **load balancing over redundant paths with same hop counts is not possible**
- **routing-updates always contain the whole routing-table !!**
  - problematical in large networks (many net-ID entries)
- **periodic routing updates produce a permanent network load**
  - especially WAN-links are most affected
- **RIP-updates are sent in 512 byte blocks, 24 entries each -> larger networks require several routing-updates, which take some seconds on slower links**

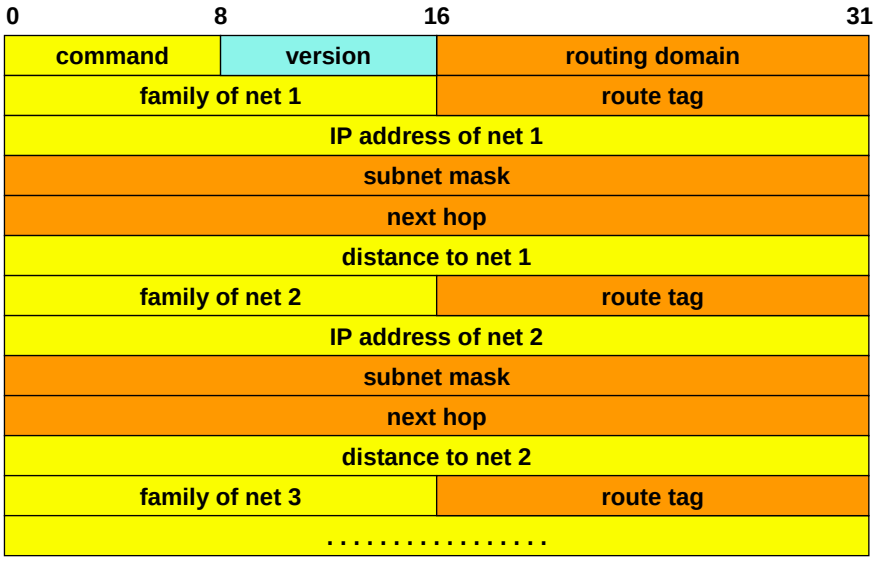
Agenda

- **RIP Principles**
- **RIP Problems**
- **RIP Workarounds**
  - maximum hop count
  - split horizon
  - poison reverse
  - triggered update
  - hold down
- **RIP version 2**

RIP Version 2

- **RFC 2453 specifies a new, extended RIP version:**
  - RIPv2 is RFC category “Standard”
  - RIPv1 is RFC category “Historic”
- **RIPv2 is an alternative choice to OSPF**
- **RIPv2 utilizes the unused fields of the RIPv1 message-format**
- **several new features are supported:**
  - Routing Domains
  - transmission of subnet-masks
  - transmission of next hop redirect information
  - route advertisements via EGP - protocols
  - authentication

RIPv2 Header plus RIPv2 Data



Message Fields of RIPv2 1

- **version = 2 (RIPv2)**
- **IP address of net x, distance to net x, command fields**
  - have the same meaning as for RIPv1
- **subnet mask**
  - contains the subnet-mask to the "IP address"-field
  - now discontiguos subnetting and variable length subnet masks (VLSM) are supported
- **RIPv2 is a classless routing protocol**

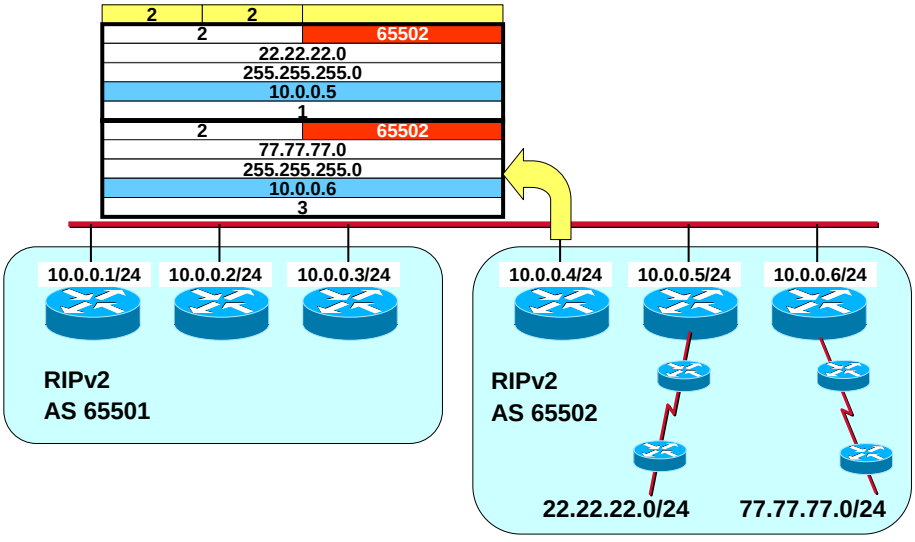
Message Fields of RIPv2 2

- **routing domain**
  - indicates the routing-process for which the routing-update is destined
  - now routers can support several domains within the same subnet
- **route tag**
  - contains the autonomous system number for EGP and BGP
  - on receiving a routing-update with a routing tag unequal zero, the associated path must be distributed to other routers; so interior routers notice the existence of exterior networks (tagging exterior routes)

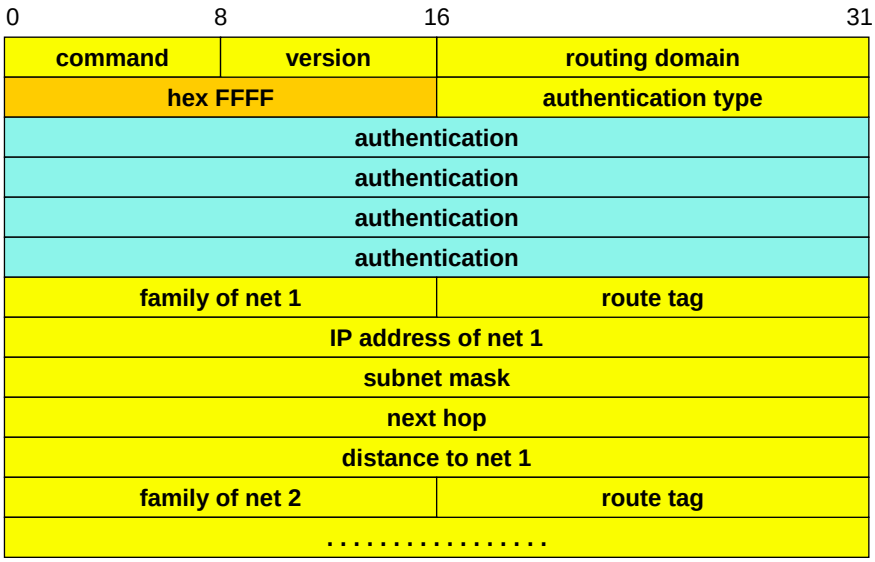
Message Fields of RIPv2 3

- **next hop**
  - datagram's for the network specified in the "IP address" - field have to be redirected to that router whose IP address is specified in the "next hop" field
  - this next-hop router must be located in the same subnet as the sender of the routing-update
  - a next hop value of 0.0.0.0 indicates, that the sender-router acts as next hop itself for the given network
  - so in cases when there are several routers in a subnet, just one router needs to send a routing-update
  - using RIPv2 this router announces which networks can be reached over other routers

Next Hop and Route Tag



RIP2 Header plus Authentication



Authentication

- **family of net = hex FFFF**
  - this value registered in the first family of net entry announces 16 authentication octets to follow
  - currently only a single type (type = 2) defined with a simple (clear text) password protection
  - Cisco supports also type 3 based on Message Digest 5 (MD5)
  - routing updates without valid authentication are ignored by the receiving router (only trusted router are accepted)
- **family of net unequal hex FFFF**
  - has the same meaning as for RIP1

Treatment of Routing Updates

- **RIPv2 uses a class D multicast-address (224.0.0.9)**
  - no use of broadcast messages (like RIPv1 does)
  - only a router who is member of this group will receive and must process this routing update
- **remember:**
  - RIPv1 used broadcast addresses
  - Seen by each IP host
  - Slows down stations

Compatibility RIPv1 <-> RIPv2

- **RIPv1 Compatibility Mode**
  - version number >1 and family of net = hexFFFF indicates RIPv2
  - RIPv1 routers will ignore header extensions
  - RIPv2 router uses broadcast addresses
- **RIPv1 Mode**
  - RIPv2 sends RIPv1 messages
- **RIPv2 Mode**
  - Send genuine RIPv2 messages
  - RIPv2 messages are sent as multicasts