RIPE 53
IPv6 Routing Tutorial

Amsterdam, 4 October 2006

Iljitsch van Beijnum

Free Books!

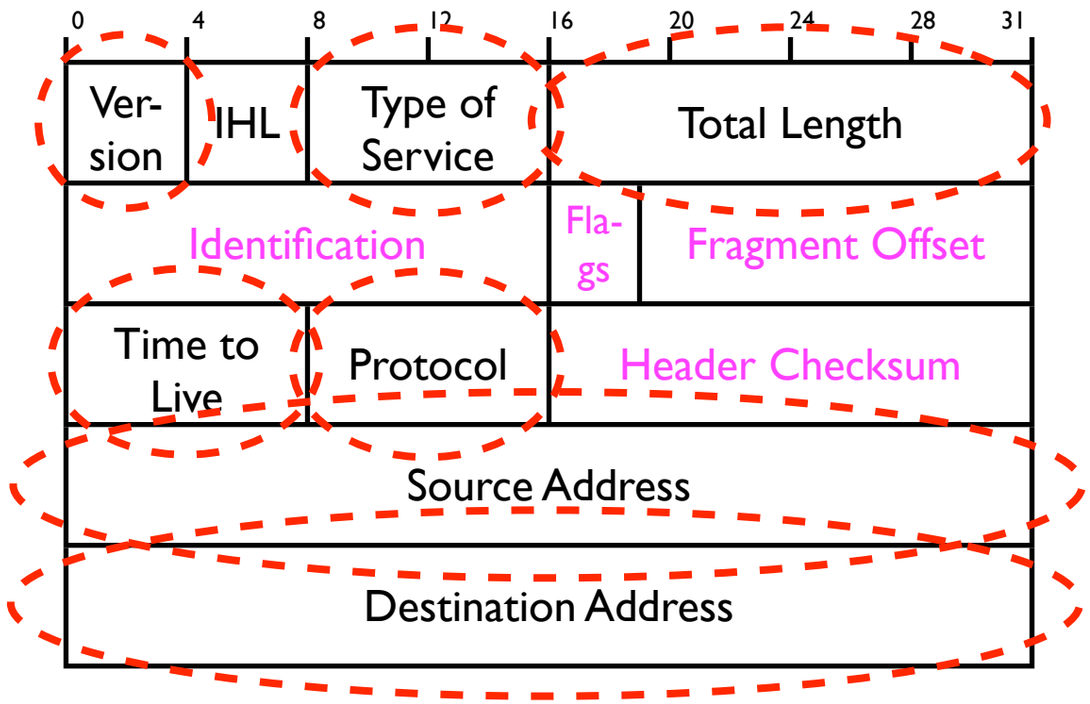
- I'll be giving away two copies of my book "Running IPv6" at 10:30

IPv6 Recap

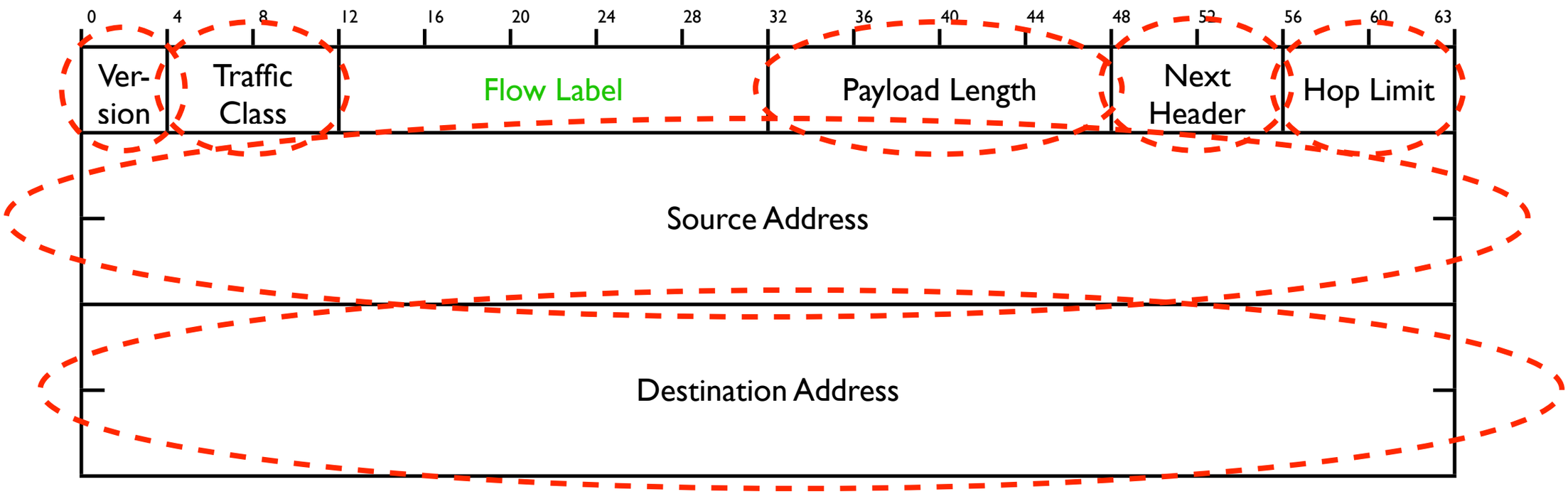
Different in IPv6

- Neighbor Discovery replaces ARP
  - uses multicast, watch the IGMP snooping!
- Minimum MTU 1280 bytes
- No fragmentation in routers → PMTUD
- Link-local addresses
- Longer addresses: 128 bits

IPv4 Header

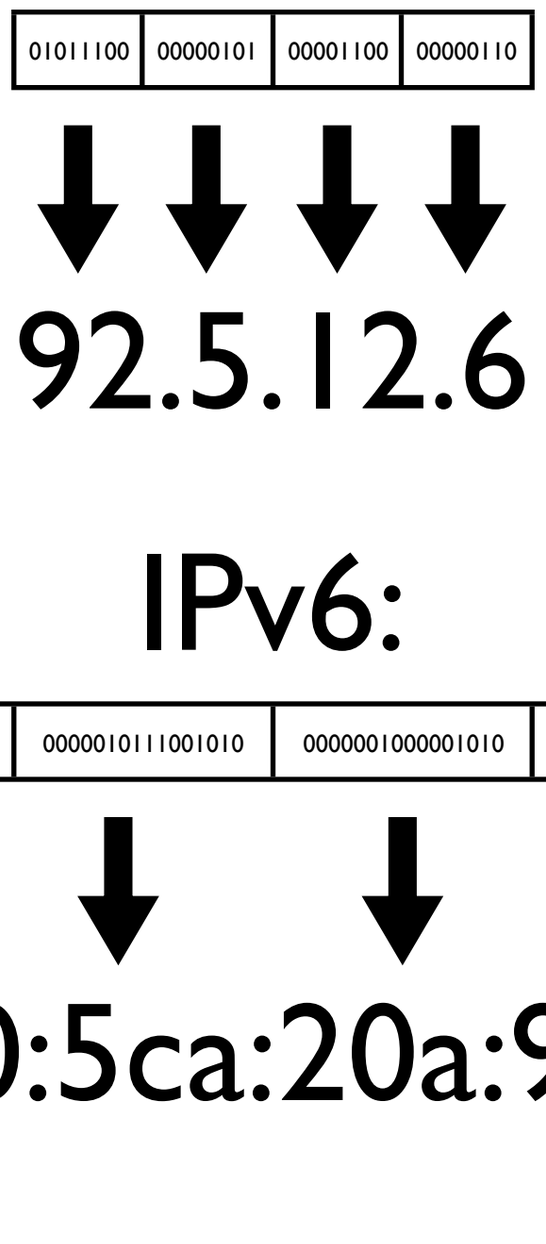


IPv6 Header



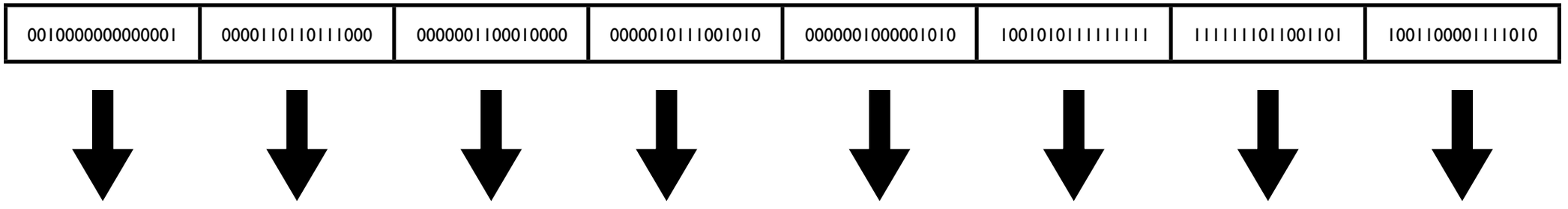
Address Notation

IPv4:



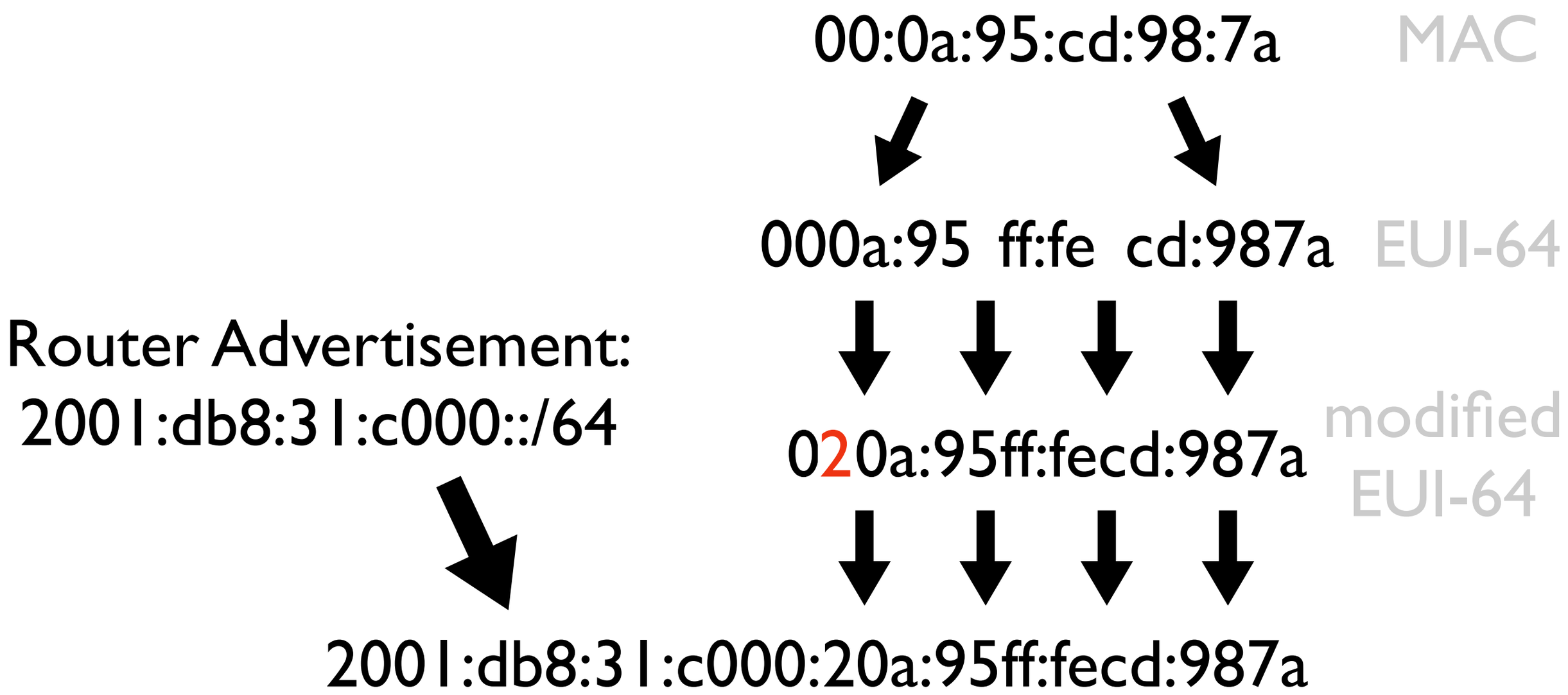
92.5.12.6

IPv6:



2001:db8:310:5ca:20a:95ff:fecd:987a

IPv6 Address Creation



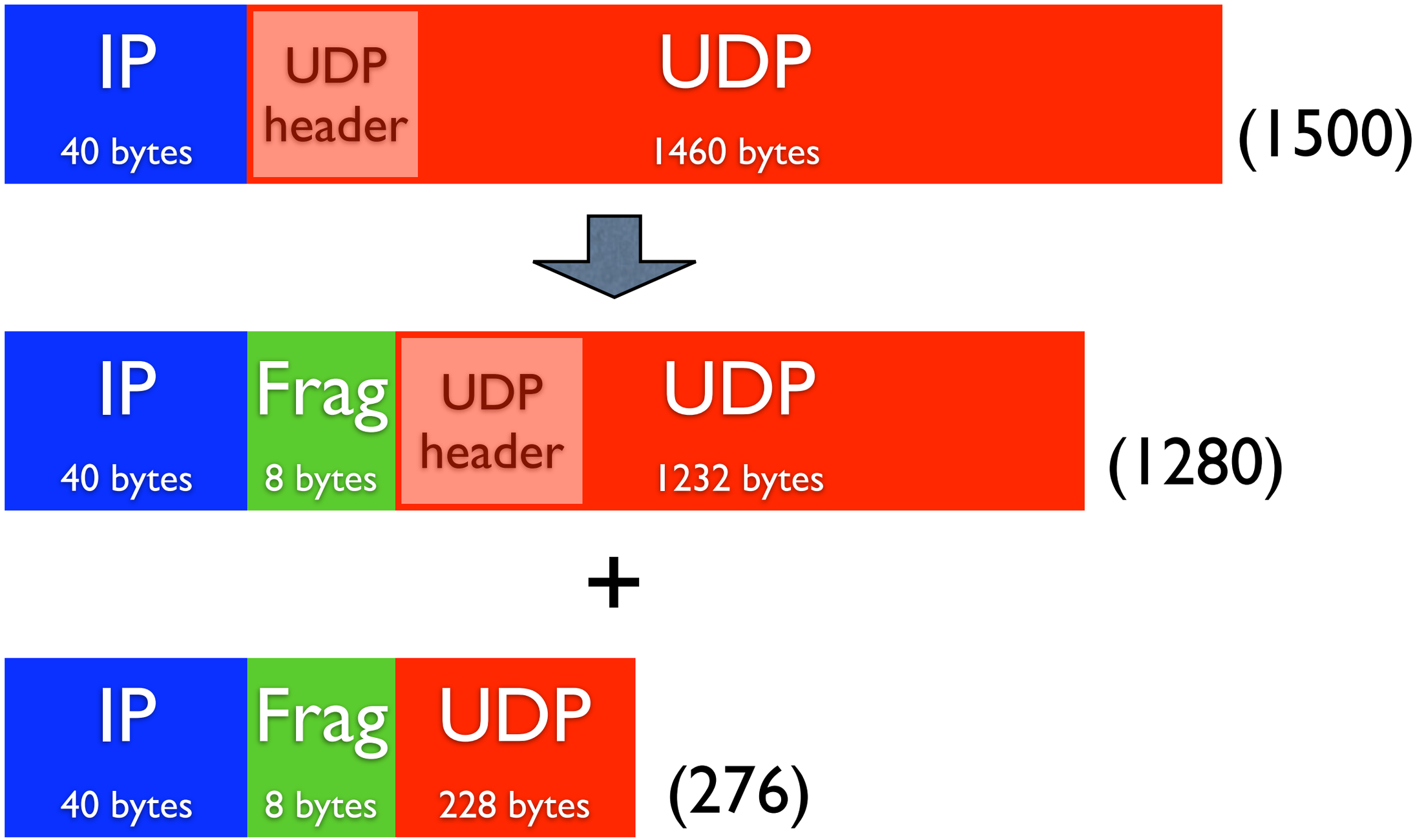
Path MTU Discovery

- Minimum maximum packet size in IPv6: 1280 bytes
- Routers can't fragment in IPv6:
  - > 1280 bytes → router returns "too big"
  - DO NOT filter ICMPv6 packet too big!

Fragmentation

- TCP adjusts to ICMP "too big" messages by reducing packet size
- UDP/ICMPv6 can't
- Solution: host fragments at the source
- Fragment header inserted between IPv6 header and UDP or other payload
  - (firewalls often don't understand this)

Fragmentation (2)



Debugging

- When behind a tunnel or other link with reduced MTU
- Large ping: `ping6 -s 1452 www.kame.net`
  - + 40 bytes IPv6 + 8 bytes ICMPv6 = 1500
- First packet lost for outgoing PMTUD
- Second lost for incoming PMTUD

MTU

- Router advertisement can have MTU option
- All hosts on subnet use advertised MTU

- On Cisco: set with `ipv6 mtu <mtu>`
- BSD/Mac: `ndp -i <interface>`
- Linux: `ip -6 route show`
- XP: `netsh interface ipv6 show interface`

Link-local Addresses

- Every IPv6 interface has a link-local address:

```
en1: flags=8863<UP,BROADCAST,SMART,RUNNING,SIMPLEX,MULTICAST> mtu 1500
     inet6 fe80::20a:95ff:fef5:246e%en1 prefixlen 64 scopeid 0x5
     ether 00:0a:95:f5:24:6e
```

```
Cisco>show ipv6 interface ethernet0
Ethernet0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::204:27FF:FEFE:249F
```

- So fe80::/64 prefix on all IPv6-interfaces...
- fe80::/64 is unroutable by design
- IPv6 routing protocols heavily use link-locals

Packet Filtering

- IPv4 and IPv6 usually treated as different
  - So need ***both*** IPv4 and IPv6 filters
    - v4 filter doesn't catch v6
    - v6 filter doesn't catch v4
- PF on *NIX is the exception

Cisco

```
!
interface Ethernet1
 ipv6 traffic-filter in-ipv6-acl in
 ipv6 traffic-filter out-ipv6-acl out
!
ipv6 access-list in-ipv6-acl
 deny ipv6 2001:DB8::/32 any
 deny icmp any host FF02::1 echo-request
 permit tcp any any established
 deny tcp any any
 permit ipv6 any any
!
ipv6 access-list out-ipv6-acl
 deny tcp any any eq smtp
 permit ipv6 any any
!
```

Cisco (2)

- Stateful rules:

```
!
ipv6access-list out-ipv6-acl
 permit tcp any any eq 22 reflect state-acl
   timeout 7500
 permit ipv6 any any reflect state-acl
!
ipv6 access-list in-ipv6-acl
 evaluate state-acl
 deny tcp any any log-input
 deny udp any any log-input
!
```

Routing

IPv6 Routing Protocols

- Completely separate from the same protocol for IPv4:
  - RIPng (next generation): simple, slow
  - OSPFv3: smart, fast

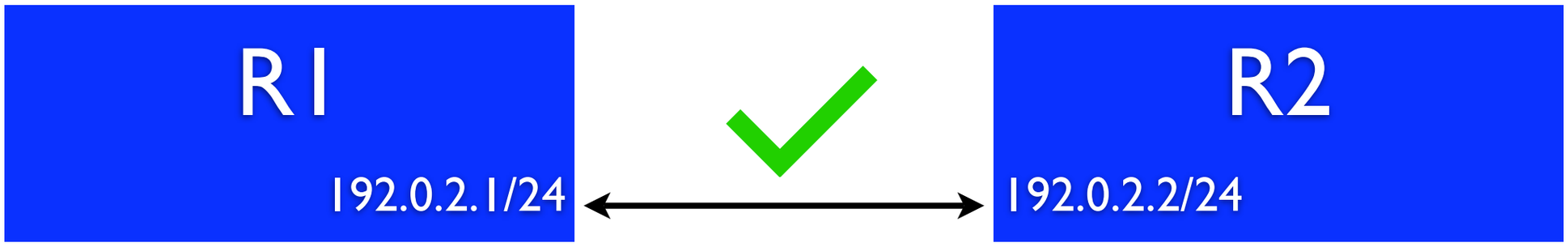
Integrated Protocols

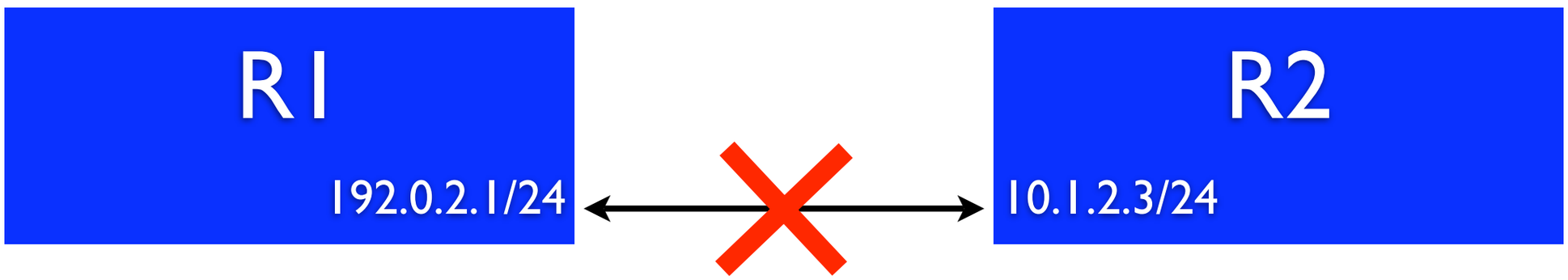
- Same protocol for IPv4, IPv6 (and possibly other protocols):
  - Integrated IS-IS: like OSPFv2/v3, but better in very large networks
  - BGP with multiprotocol extensions: for inter-domain routing

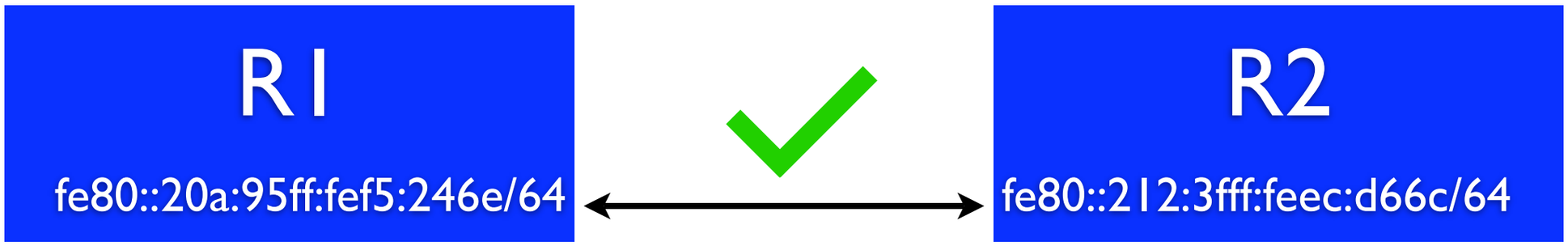
Link-locals

- Important feature in IPv6 routing:
- No need to share subnet prefix!
  - In RIPng/OSPFv3 all routing protocol interactions over link-locals
  - Also: all protocols (including BGP) must exchange link-local next hop addresses for proper ICMPv6 redirect handling

Link-locals (2)







Subnet Prefix Size

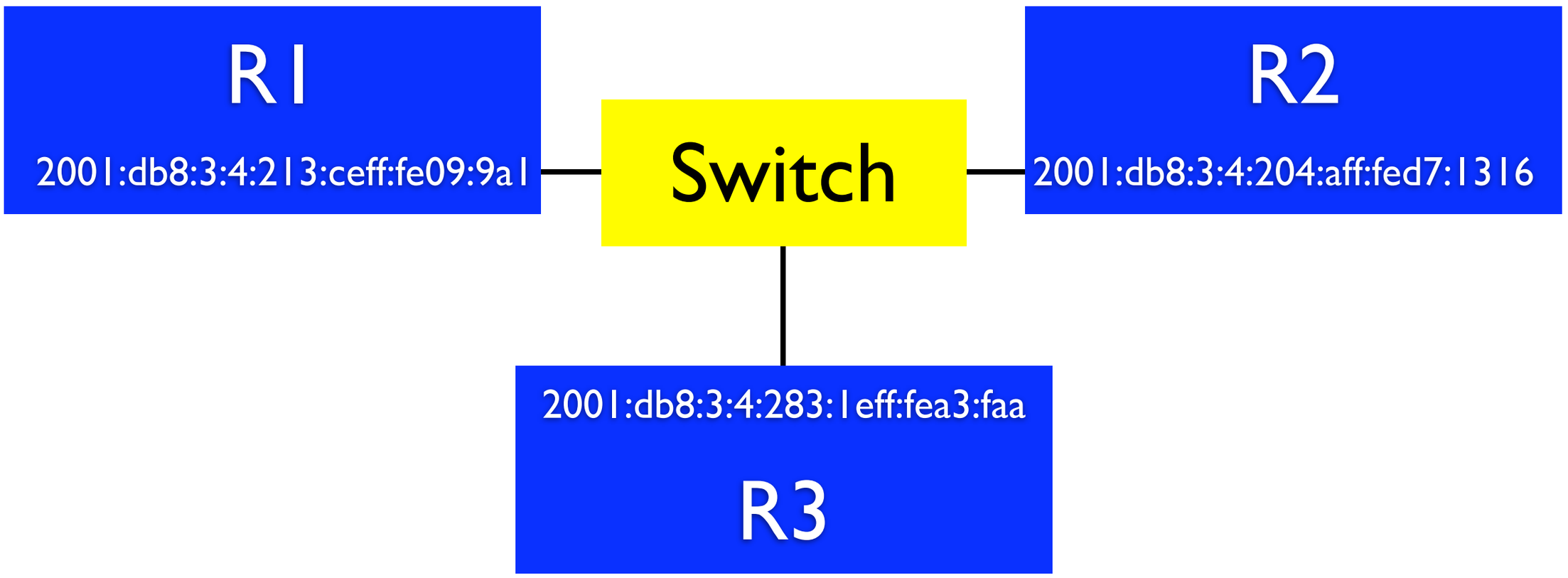
- ipv6 unnumbered: ok
- /127: dangerous, 0 = all router anycast
- /126: works
- /120: ok
- /112: ok
- /64: best choice (RFC 3513 / EUI-64)

EUI-64 Addressing

- Routers as a rule don't configure addresses with stateful autoconfig
- But can often still generate bottom 64 bits of address from EUI-64
- Very useful: don't have to keep track of router addresses

EUI-64 Addressing (2)

```
!
interface Ethernet0
 ipv6 address 2001:db8:3:4:/64 eui-64
!
```



Router ID

- Protocols such as OSPFv3 and BGP need a router ID to work
- Router ID is/looks like an IPv4 address
- No IPv4 address: protocol can't run, need to set router ID manually
- Zebra ospf6d can't take router ID from existing IPv4 address, configure explicitly

Zebra/Quagga

- GPL: free as in beer/speech
- Runs on BSD, Linux and even MacOS
- Zebra development extremely slow, Quagga is a fork
- Includes: RIP, OSPF, RIPng, OSPFv3 and BGP
- Daemon per protocol + zebra daemon to coordinate and manage kernel routing table

Cisco

- Too many IOS versions... Some do IPv6
  - (but not on really old stuff)
    - performance depends on ASIC/linecard
- Small boxes only RIPng
- Some others also OSPFv3 and BGP
- Expensive stuff all of them, including IS-IS

Juniper

- Only has expensive stuff, much simpler
- Talk of expensive IPv6 license???
- IPv6 supported in ASIC, so very fast
- Haven't tested RIPng or IS-IS
- Unlike Cisco/Zebra IPv6 routing protocol config *very* similar to IPv4 config

RIPng on Zebra

- zebra config:

```
interface gif0
 ipv6 address 3ffe:2500:310:4::1/64
 ipv6 nd suppress-ra
```

- ripngd config:

```
router ripng
 default-information originate
 network gif0
 network em0
```

show interface

```
zebra-t# show interface eth0
Interface eth0
  Description: First Ethernet interface
  index 3 metric 1 mtu 1500 <UP,BROADCAST,RUNNING,MULTICAST>
  HWaddr: 00:01:02:29:23:b6
  inet 172.16.1.5/24 broadcast 255.255.255.255
  inet6 fe80::201:2ff:fe29:23b6/64
  inet6 2001:db8:31:2::1/64
    input packets 9624, bytes 1142979, dropped 0, multicast packets 0
    input errors 0, length 0, overrun 0, CRC 0, frame 0, fifo 0, missed 0
    output packets 5549, bytes 1042517, dropped 0
    output errors 0, aborted 0, carrier 0, fifo 0, heartbeat 0, window 0
    collisions
```

show ipv6 ripng

```
ripngd# show ipv6 ripng
Codes: R - RIPng

   Network                    Next Hop                  If Met Tag Time
R  ::/0                       ::                         0   1   0
R  2001:db8:31:1::/64         fe80::260:70ff:fe35:aa5e   3   2   0 02:59
S  2001:db8:31:2::/64         ::                         3   1   0
R  3ffe:9500:3c:600::/56      fe80::204:27ff:fefe:249f   3   2   0 02:54
```

RIPng on Cisco

```
!
ipv6 unicast-routing
!
interface Ethernet0
 ipv6 address 3FFE:2500:310:3::/64 eui-64
 ipv6 rip athome enable
 ipv6 rip athome default-information originate
!
ipv6 router rip athome
 redistribute connected
 redistribute static
!
```

IPv6 on Cisco

- Not on by default, use `ipv6 unicast-routing`
- Can do EUI-64 addressing. Or not.
- When address present on interface, router advertisements are sent
  - suppress with: `ipv6 nd suppress-ra`

show ipv6 rip database

```
#show ipv6 rip database
RIP process "athome", local RIB
 2001:DB8:31:2::/64, metric 2, installed
     Ethernet0/FE80::204:27FF:FEFE:249F, expires in 155 secs
 3FFE:9500:3C:600::/56, metric 2, installed
     Ethernet0/FE80::201:2FF:FE29:23B6, expires in 173 secs
```

OSPFv3 on Zebra

```
!
interface em0
 ipv6 address 2001:db8:31:6::1/64
 ipv6 ospf6 cost 10
!
router ospf6
 router-id 192.0.2.18
 redistribute static
 interface em0 area 0.0.0.0
!
```

OSPFv3 on Cisco

```
!
interface FastEthernet2/0
 ipv6 address 2001:DB8:31:6::/64 eui-64
 ipv6 ospf 230 area 0.0.0.0
 ipv6 ospf cost 10
!
ipv6 router ospf 230
 log-adjacency-changes
 default-information originate
 redistribute connected
 redistribute static
!
```

show ipv6 route ospf

```
#show ipv6 route ospf
IPv6 Routing Table - 644 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS -
ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2
- OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2

O   2001:7F8:1::/64 [110/2]
     via FE80::290:6902:EE02:E43E, FastEthernet2/0
O   2001:DB8:31:2::/64 [110/2]
     via FE80::212:1E02:EE05:58DB, FastEthernet2/0
OE2  3FFE:9500:3C:600::/56 [110/0]
     via FE80::212:1E02:EE05:58DB, FastEthernet2/0
```

show ipv6 ospf neighbor

```
#show ipv6 ospf neighbor
Neighbor ID  Pri  State           Dead Time  Interface ID  Interface
192.0.2.91   128  FULL/BDR        00:00:38   3             Ethernet2/0
192.0.2.17   128  FULL/DROTHER    00:00:35   2             Ethernet2/0
192.0.2.19     1  FULL/DROTHER    00:00:30   8             Ethernet2/0
```

OSPFv3 on Juniper

```
interfaces {
    ge-0/0/0 {
        family inet6 {
            address 2001:db8:31:6::/64 {
                eui-64;
            }
        }
    }
}
protocols {
    ospf3 {
        area 0.0.0.0 {
            interface ge-0/2/0;
        }
    }
}
```

BGP

- Address policies very different from IPv4
- One BGP session can carry both IPv4 and IPv6 routes
- But for eBGP better to have IPv4 session for IPv4 info and IPv6 session for IPv6 info
- For iBGP easier to have just one session
- Cisco "show ip bgp" equivalent: "show bgp"

BGP on Zebra

```
!
router bgp 65000
 bgp router-id 192.0.2.1
 neighbor 2001:db8:8:34::1 remote-as 64702
 !
 address-family ipv6
 network 3ffe:2500:310::/48
 neighbor 2001:db8:8:34::1 activate
 neighbor 2001:db8:8:34::1 prefix-list 6bone in
 exit-address-family
!
ipv6 prefix-list 6bone seq 5 permit 3ffe::/16 le 48
!
```

BGP on Cisco

```
!
routerbgp 65500
 bgp log-neighbor-changes
 neighbor 3ffe:9500:3C:74::10 remote-as 64900
 no neighbor 3ffe:9500:3C:74::10 activate
!
address-family ipv6
 neighbor 3ffe:9500:3C:74::10 activate
 neighbor 3ffe:9500:3C:74::10 prefix-list out-v6 out
 network 2001:DB8:31::/48
 no synchronization
 exit-address-family
!
ipv6 prefix-list out-v6 seq 5 permit 2001:DB8:31::/48
ipv6 route 2001:DB8:31::/48 Null0
!
```

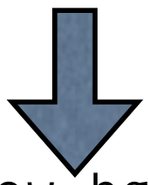
Two Next Hops

- BGP for IPv6 has two next hop addresses:
  - regular global one like in IPv6

    (shows up in "show bgp" commands)
  - link local one

    (shows up in "show ipv6 route")
- Necessary for generating proper redirects

BGP on Cisco



```
#show bgp
BGP table version is 3, local router ID is 10.0.0.10
   Network          Next Hop            Metric LocPrf Weight Path
*> 2001:DB8:31::/48 3ffe:9500:3C:74::10
                                             0             0 9000 i
```

```
#show ipv6 route
IPv6 Routing Table - 17 entries
B   2001:DB8:31::/48 [20/0]
     via FE80::20A:95FF:FECD:987A, Ethernet0
```

iBGP on Cisco (1)

```
!
routerbgp 65500
 neighbor rrclients peer-group
 neighbor rrclients remote-as 65500
 neighbor 172.16.1.5 peer-group rrclients
 !
 address-family ipv4
 neighbor rrclients activate
 neighbor rrclients route-reflector-client
 neighbor 172.16.1.5 peer-group rrclients
 no synchronization
 network 192.0.2.0
 exit-address-family
 !
...
```

iBGP on Cisco (2)

```
...
 !
 address-family ipv6
 neighbor rrclients activate
 neighbor rrclients route-reflector-client
 neighbor 172.16.1.5 peer-group rrclients
 neighbor 172.16.1.5 activate
 network 2001:DB8:31::/48
 no synchronization
 exit-address-family
!
```

- IPv6 (and IPv4) routing information exchanged over IPv4 iBGP session

BGP on Juniper (1)

```
protocols {
    bgp {
        group ibgp {
            type internal;
            local-address 192.0.2.7;
            family inet {
                unicast;
            }
            family inet6 {
                unicast;
            }
            peer-as 65500;
            neighbor 192.0.2.18;
        }
...
```

BGP on Juniper (2)

```
...
            group bgp-v6 {
            type external;
            import bgp-v6-in;
            family inet6 {
                unicast;
            }
            export bgp-v6-out;
            neighbor 2001:7f8:1::a506:3000:1 {
                authentication-key "$9$5Fdsikekasi/97dj";
                  ## SECRET-DATA
                peer-as 64900;
            }
        }
    }
```

Filtering

- ISPs get /32 or shorter prefixes, but...
  - http://lacnic.net/en/registro/index.html
  - https://www.ripe.net/ripe/docs/ripe-ncc-managed-address-space.html
  - http://www.arin.net/reference/micro_allocations.html
  - http://www.apnic.net/db/min-alloc.html
- Or: http://www.space.net/~gert/RIPE/ipv6-filters.html

IPv6 Global Table

- You'll see:
  - many /32s (ISP PA blocks)
  - some < /32 (very large ISP PA blocks)
  - a few /35 (old ISP PA blocks)
  - 2002::/16 (6to4 automatic tunneling)
  - some /48s (criticial infra, exchanges, PI...)

How to Filter?

- No filters: depend on maximum-prefix
- Allow anything < /64: rejects very little
- Allow anything <= /48
  - still 65536 /48s per /32...
- Allow anything <= /32 + exception blocks
- Allow only actual allocations (still doable)

Multihoming

- Until recently: no IPv6 blocks for end-users
- IETF working on shim6, doesn't need BGP
- But now some PI possible
- Using /48 out of ISP /32, similar to IPv4?
  - filtering on by others gets in the way... BGP community the solution? See:

`draft-van-beijnum-v6ops-pa-mhome-community-01.txt`

Thanks for listening!

http://www.bgpexpert.com/

iljitsch@muada.com