CCNP ROUTE

ROUTE Module 1.5

Agenda

- **Cisco Documentation**
- **Network Principles**
- **Routing Basics**
  - Facts and fiction about routing protocols and addressing
  - NBMA
- **IPv6**
- **Configuration**
  - /31 masks on point-to-point links
  - IP Unnumbered
  - Static routes
  - On Demand Routing (ODR)
  - RIPv2, RIPng

Absolute Mandatory Commands Minimum

- To alleviate and ease your work with Cisco boxes in labs:

```
# write erase
# delete flash:vlan.dat
# reload
(conf)# line console 0
(conf-line)# logging synchronous
(conf)# line vty 0 15
(conf-line)# logging synchronous
(conf-line)# no login
(conf-line)# privilege exec level 15
(conf)# no ip domain-lookup
(conf)# ip host NAME IP
(conf)# terminal monitor
```

CCNP Study Materials

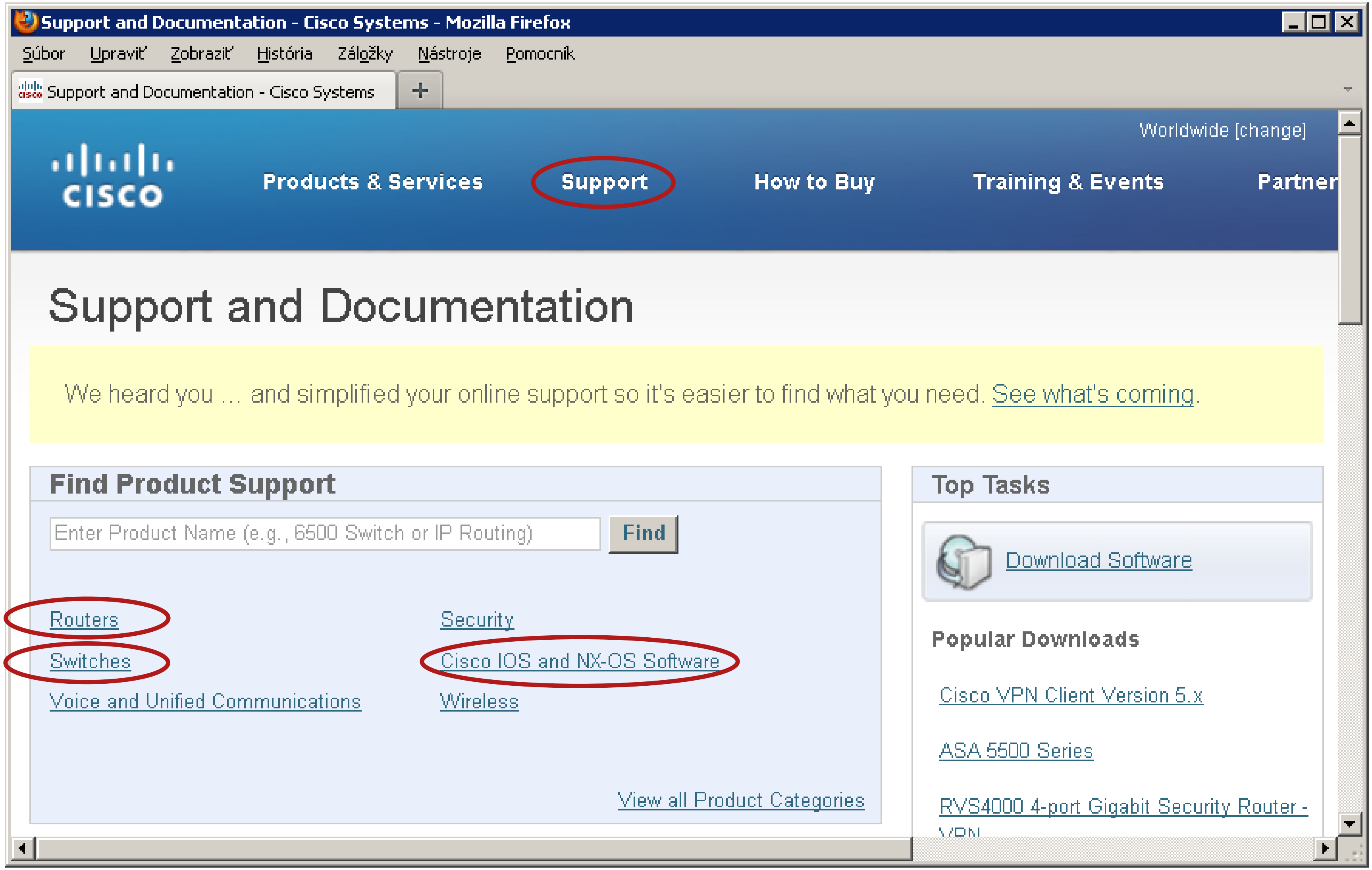
CCNP Study Materials ①

- *No web curriculums at all!!!*
- Not enough details in course
  - *hence cisco.com is your best friend*
- Orientation on web pages are crucial for all IT networkers
  - *…and they are trying to sabotage it all the time* ☺
  - Huge knowledgebase

Cisco Web Documentation

- Products documentation available
  - by HW platforms
  - by IOS versions
- Experience learn us that IOS commands…
  - for routers are best to find directly in relevant IOS documentation
  - for switches are best to find directly in relevant switch product documentation
- *It is usually good to know exact IOS version* (?)

http://cisco.com/go/support



IOS Documentation

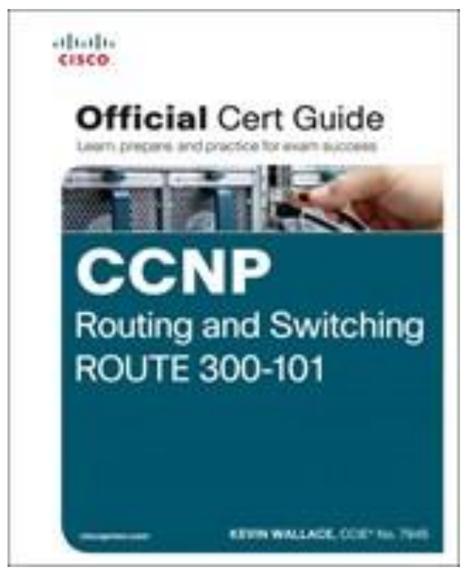
- The most important/interesting are following parts:
  - Configuration Guides consist of thorough description of technologies or protocols and ways how to configure them
  - Command References consist of commands descriptions, syntax and semantics
  - Master Index is alphabet index of commands with references to Command Reference
  - Error and System Messages consist of lists of IOS messages and theirs explanations
- Alternatively it's possible to use Command Lookup Tool to find Command Reference to appropriate command
  - CCO account needed!

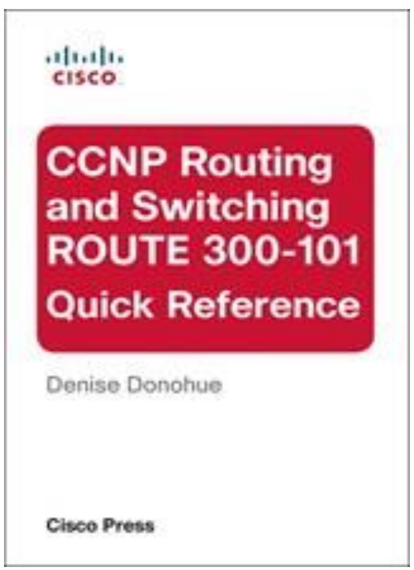
Supporting Documentation

- Case-studies, principle descriptions, configuration examples, technologies reviews
- Many of them have **Document ID** ***NUMBER***
- How to search for them
  - „Configuring ...“
  - „Understanding ...“
  - „Troubleshooting ...“
  - „How to ...“
  - Support → Cisco IOS and NX-OS Software → Technology
- Cross-referencing between documents. Hence, it’s necessary to make bookmarks (Ctrl+D)

Self-study Literature

- **Implementing Cisco IP Routing (ROUTE) Foundation Learning Guide: (CCNP ROUTE 300-101)**
- **CCNP Routing and Switching ROUTE 300-101 Official Cert Guide**





Network Design

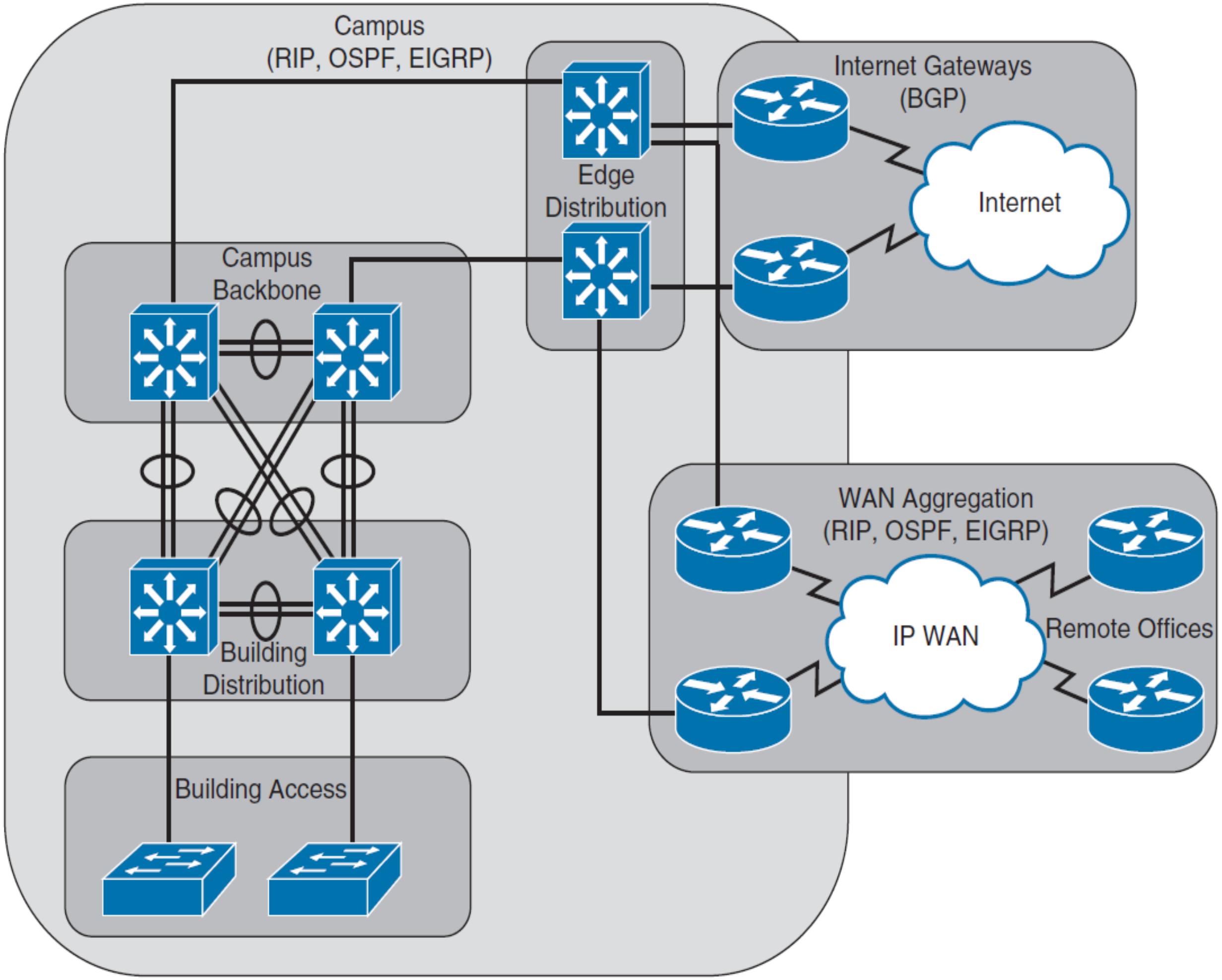
Features of a good design

- Ad-hoc approach *leads you to hell and further!*
- Hierarchically designed network:
  - Has well-known borders of collision, broadcast and error domains
  - Has positive impact on network operation and management
  - Scalable assignment of IP addresses together with their summarization
  - Transparent network flows
  - Divides L2 and L3 functionality

3Layered Network Design

- *Bigger network means more attached devices*
- It’s favorable to divide them according to theirs network function thereby organize them into layers
  - Access connectivity for end devices
  - Aggregation and routing
  - Fast backbone switching or routing
- System of those three layers (access, distribution, core) is old, traditional but still working

3Layered Hierarchical Model



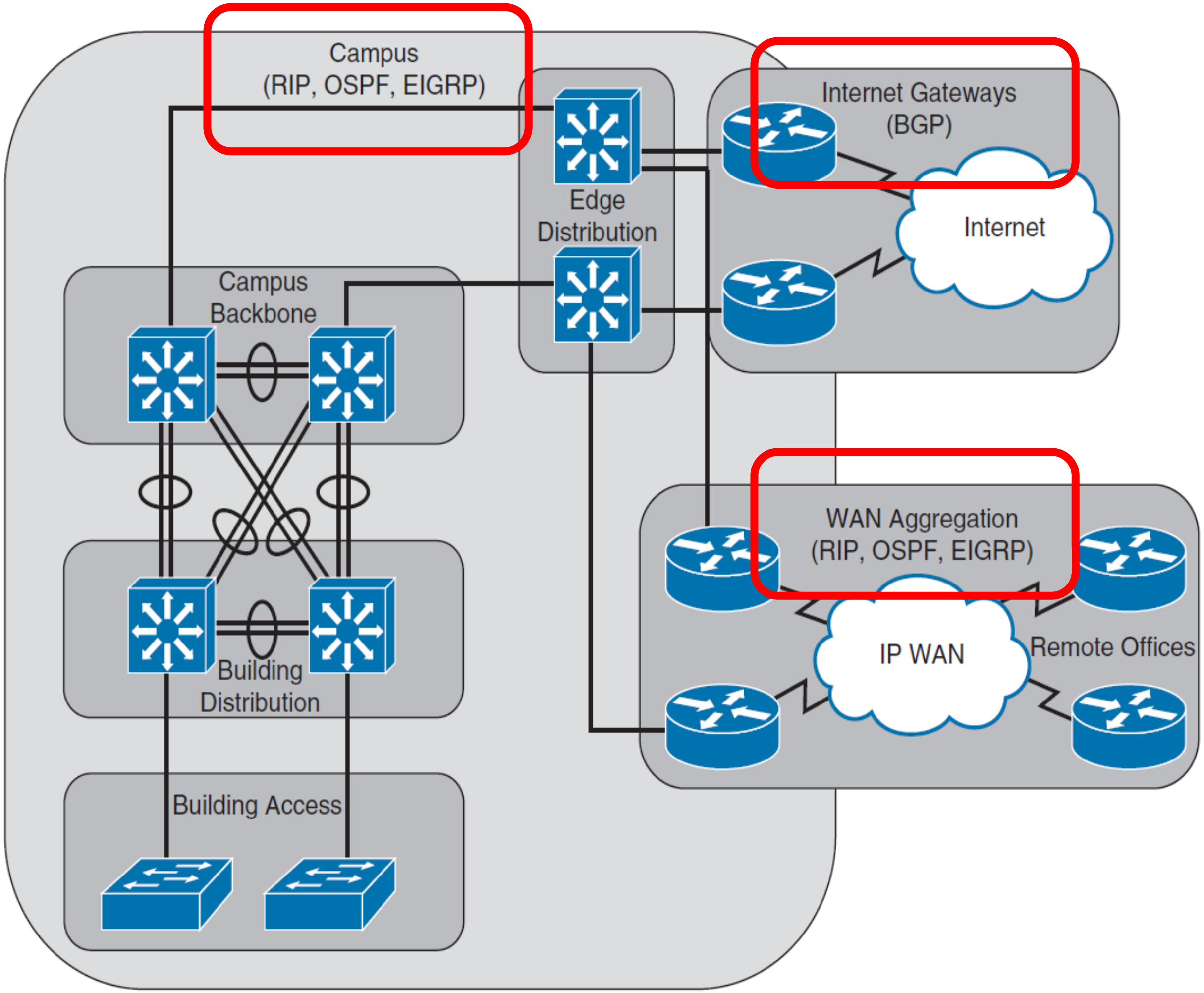
Layers Function ①

- **Access Layer**
  - Usually just switching occurs, sometimes even routing
  - Provides client access to network, VLAN assignment, first line of QoS marking and port-security mechanisms to access the medium
- **Distribution Layer**
  - Aggregates building access switches.
  - Usually routing occurs
  - Provides inter-VLAN communication, address summarization, policy-based routing, enforcing QoS and division of error domains
- **Core Layer (Backbone)**
  - high-speed transfer of data through the network
  - High-end multilayer switches are often used here.

Layers Function ②

- **Edge Distribution**
  - Serves as the ingress and egress point for all traffic
  - Routers or multilayer switches are appropriate devices for this layer.
- **Internet Gateways**
  - Connects the Campus network out to the Internet
  - Single connection or multiple ISP
- **WAN Aggregation**
  - Connects the Campus network out to remote offices
  - MPLS is usually used here

Routing in Enterprise network



Routing protocols selection

- We will learn about the RIPv2, RIPng, OSPFv2, OSPFv3, EIGRP, BGP, MP-BGP and IS-IS routing protocols
- Which is the right one?

Characteristics to consider:

- Scalability
- Vendor interoperability
- IT staff's familiarity with protocol
- Speed of convergence
- Capability to perform summarization
- Interior or exterior routing
- Type of routing protocol

Routing protocols categories

- Distance-Vector – RIP, EIGRP
  - Sends a full copy of its routing table to its directly attached neighbors
  - Convergence time can be an issue
  - Necessary techniques for eliminating routing loops
    - Split horizon and poison reverse
- Link-State – OSPF, IS-IS
  - Routers build a topological map of a network
  - Full routing information are exchanged only when two routers initially form their adjacency
- Path-Vector – BGP
  - information about the exact path packets take to reach a specific destination network

Routing Basics

IP Protocol

- IPv4 RFC 791
- IPv6 RFC 2460

- IP guarantees
  - Logical addressing of networks and host belonging to them
  - A way how to deliver packets between end-users
  - Best-effort delivery

IPv4

- In IPv4 every network interface has its own address
  - Errata exists – **`ip address`** *`A.B.C.D`* **`secondary`**
- Address is 4B long written in dot-decimal notation
  - *Don’t be shy and try* **`ping 2481303803`** ☺

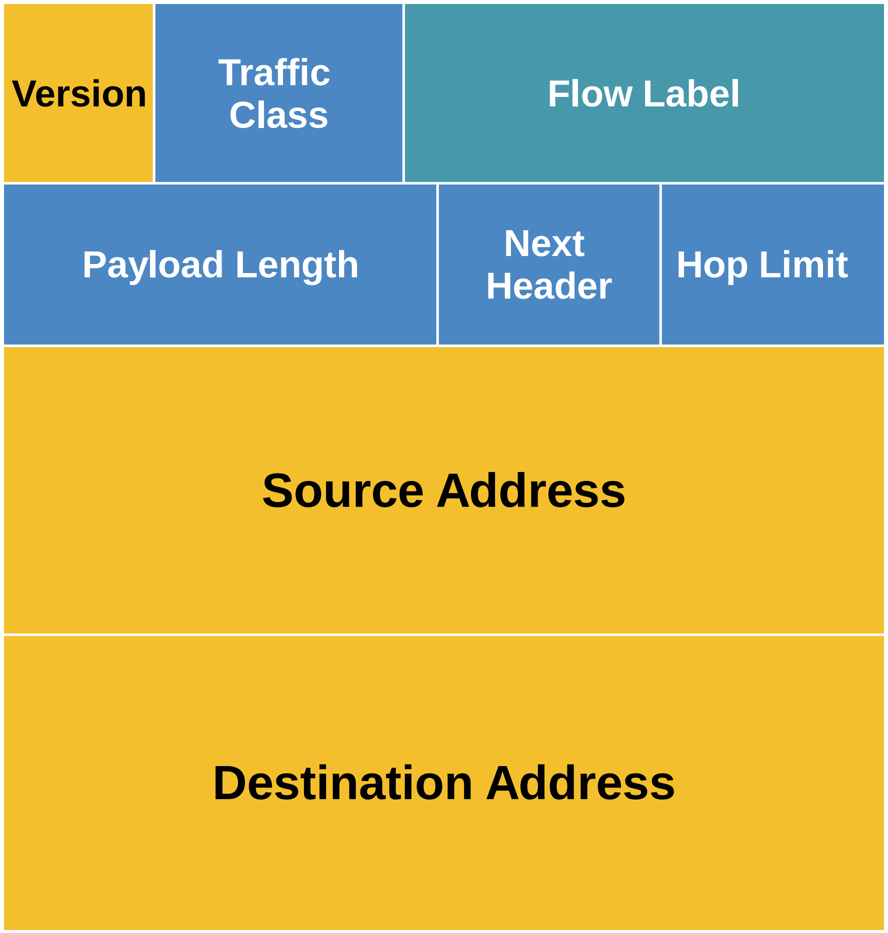
<table>
<tr><td>Version</td><td>IHL</td><td>Type of Service</td><td colspan="2">Total Length</td></tr>
<tr><td colspan="3">Identification</td><td>Flags</td><td>Fragment Offset</td></tr>
<tr><td colspan="2">Time to Live</td><td>Protocol</td><td colspan="2">Header Checksum</td></tr>
<tr><td colspan="5">Source Address</td></tr>
<tr><td colspan="5">Destination Address</td></tr>
<tr><td colspan="4">Options</td><td>Padding</td></tr>
</table>

IPv4 header – example

```
◢ Internet Protocol Version 4, Src: 13.76.219.191, Dst: 10.0.2.15
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes
  ▷ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 40
    Identification: 0xe074 (57460)
  ◢ Flags: 0x00
      0... .... = Reserved bit: Not set
      .0.. .... = Don't fragment: Not set
      ..0. .... = More fragments: Not set
    Fragment offset: 0
    Time to live: 64
    Protocol: TCP (6)
  ▷ Header checksum: 0xa541 [validation disabled]
    Source: 13.76.219.191
    Destination: 10.0.2.15
```

IPv6

- In IPv6 every network interface has **several** addresses
- Address is 16B long written in hex notation



IPv6 header – example

```
Internet Protocol Version 6, Src: fe80::e98c:aac:5b07:6d60, Dst: ff02::c
    0110 .... = Version: 6
    .... 0000 0000 .... .... .... .... .... = Traffic class: 0x00 (DSCP: CS0, ECN: Not-ECT)
    .... .... .... 0000 0000 0000 0000 0000 = Flowlabel: 0x00000000
    Payload length: 154
    Next header: UDP (17)
    Hop limit: 1
    Source: fe80::e98c:aac:5b07:6d60
    Destination: ff02::c
```

IP addressing, routing, subnetting

Global addressing

Address notation in IPv6 networks

- Originally very flexible

```
2001:db8:0:0:1:0:0:1      2001:0db8:0:0:1:0:0:1
2001:db8::1:0:0:1         2001:db8::0:1:0:0:1
2001:0db8::1:0:0:1        2001:db8:0:0:1::1
2001:dB8:0000:0:1::1      2001:DB8:0:0:1::1
```

- RFC 5952 is the new norm!
- Only valid representation is `2001:db8::1:0:0:1`

Network address

- Every address has two parts:
  - **Network ID** a.k.a. prefix, network part, NetID
  - **HostID** a.k.a. host part
- Routing in any routing protocol concerns only NetID
  - *Once we deliver packet to borders of right network, the rest of work is on L2 delivery mechanism*
  - 1 IP network = 1 broadcast domain
  - All hosts on same segment consider themselves as adjacent – they're able to communicate with each other
    - This is not true for IPv6!

Network ID

- *It has variable length!*
- Many ways in history how to derive it:
  - 1st approach: the first octet is NetID, the rest is Host ID
  - 2nd approach: IP address classes (A, B, C, D, E)
  - 3rd approach: subnet mask (CIDR, VLSM) (*?*)
- When length of NetID is variable, there is well-known term **network address** which is always 4B/16B long
  - = Net ID complemented with 0 up until 4B/16B length
  - **Broadcast** = Net ID complemented with 1 up until 4B length
    - Only valid for IPv4 network!

Subnet Mask ①

- Meaning of **subnet mask**:
  - 1: n^th^ bit is included into NetID
  - 0: n^th^ bit is included into HostID

158	193	138	40
10011110	11000001	10001010	00101000
11111111	11111111	11111111	00000000

- IP address AND subnet mask = NetID

Subnet Mask ②

- Border between NetID and HostID doesn't have to be align to bytes (case of VLSM and CIDR)
- *Hence NetID doesn't have to end on 0*

158.193.138.40 & 255.255.255.224 = 158.193.138.32

10011110	11000001	10001010	00101000

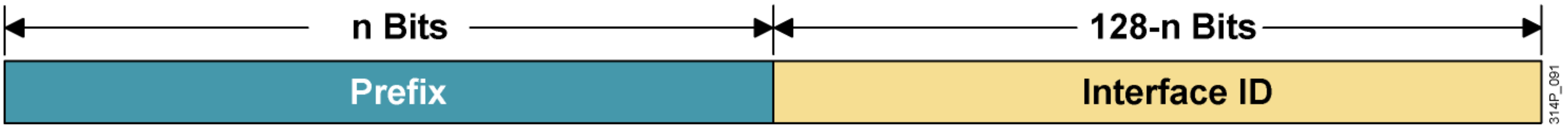
AND

11111111	11111111	11111111	11100000

=

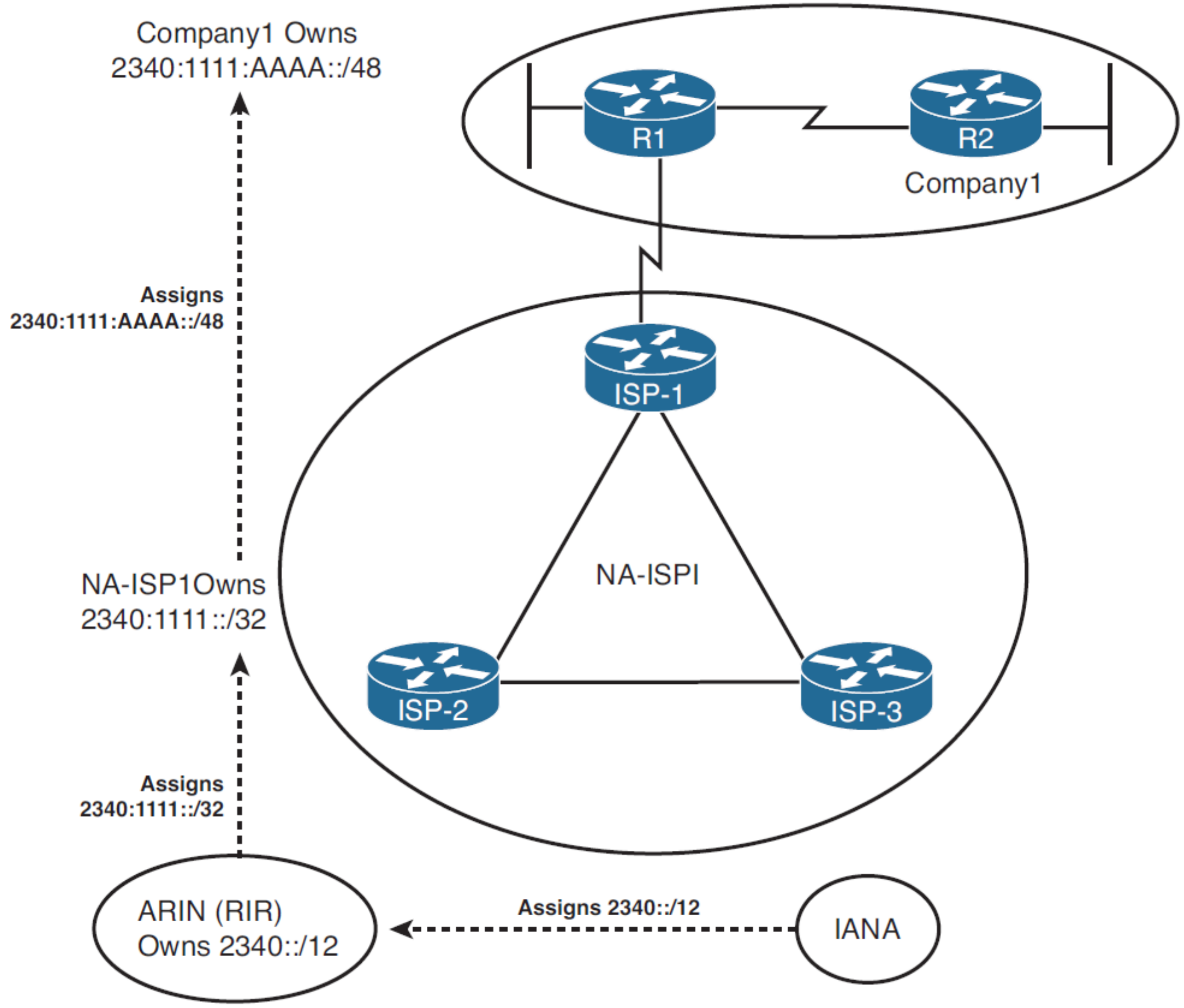
10011110	11000001	10001010	00100000

IPv6 prefix



- IPv6 uses a classless view of addressing, with no concept of classful addressing
- E.g. `2001:db8:1234:5678:1234:5678:9abc:1111/64`

Example of address assignment



How to subnet IPv6 prefix?

- Depends on enterprise requirements, however, there are several rules
- LAN – always /64
- Point-to-point links – /64 or /127
- It is wise to keep nibble boundaries for hierarchical routing

```
2001:67c:1220:900::/56
```

Basic Configuration

- IPv6 has to be enabled first

```
Router(config)# ipv6 unicast-routing
Router(config)# ipv6 cef
```

- IPv6 is configured in the same manner as IPv4
  - All commands use **ipv6** instead of **ip**

```
Router(config)# ipv6 route 2000::/3 2001:4118:300:122::1
Router(config)# interface fa0/0
Router(config-if)# ipv6 address 2001:4118:300:123::1/64
```

IPv6 Address Configuration ①

- Static address:

```
Router(config-if)# ipv6 address 2001:4118:300:123::1/64
```

- IPv6 Address using EUI-64:

```
Router(config-if)# ipv6 address 2001:4118:300:123::/64 eui-64
```

- Stateless autoconfiguration (RS/RA):

```
Router(config-if)# ipv6 address autoconfig
```

Router Functionality

- Router uses longest prefix match algorithm to determine destination network/interface
  - This decision repeats on every router independently
  - Only from routers point of view
  - Decision in forward direction doesn’t affect backward direction
- Router stores list of destination networks in its **routing table**
  - *What minimally is in every routing table?*
  - NetID and subnet mask
  - IP address of next-hop
    - IGP: address of adjacent neighbor
    - EGP: address of border router of AS
  - Additional information for route (metric, AD)

Routing Table

- *There's no way how to store whole path!*
- Internally sorted descendant by subnet mask
  - `show ip route` is sometimes sorted differently but remember it doesn't matter
- The most specific NetID is used for routing decision
  - A.k.a. **longest prefix match**

Routing Table ②

- Next-hop L3 addresses are translated by appropriate protocol to L2 addresses of neighbor
  - *Which protocols do you know?*
  - ARP, InvARP, dialer mapping, NDP…
  - Next-hop addresses are never used in IP header unless router is intended recipient of packet
- In some cases only outgoing interface could be used without next-hop address
  - Suitable only for point-to-point links
  - Deathtrap for multi-access interfaces!

Routing Table ③

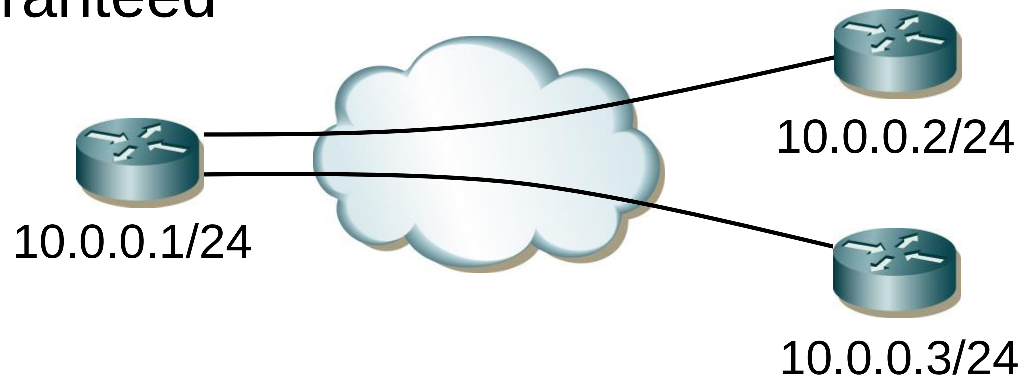
- Conditions to insert network into routing table:
  1. `IF` destination network is directly connected
     `THEN` outgoing interface MUST be „up, line protocol up"
  2. `IF` destination network is accessible via next-hop
     `THEN` it MUST be possible to recursively find out next-hop outgoing interface
  - *In other words, every record in routing table must point on up and working interface (even after recursive lookup)*
- `IF` the one of these condition become invalid
  `THEN` destination network is removed from routing table

Recursive Lookup

```
R1# show ip route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 1 subnets
C       10.0.0.0 is directly connected, Serial1/0
S    11.0.0.0/8 [1/0] via 10.0.0.2
S    12.0.0.0/8 [1/0] via 11.0.0.2
S    13.0.0.0/8 [1/0] via 12.0.0.2
S    14.0.0.0/8 [1/0] via 13.0.0.2

R1# configure terminal
R1(config)# no ip route 12.0.0.0 255.0.0.0

R1(config)# do show ip route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 1 subnets
C       10.0.0.0 is directly connected, Serial1/0
S    11.0.0.0/8 [1/0] via 10.0.0.2
```

NBMA Networks ①

- **Non-Broadcast**
  - Used L2 technology has no means how to deliver broadcasts
  - Sender has to guarantee broadcast distribution on its own
  - Usually on point-to-point circuits (ATM, X.25, Frame Relay, Dynamic Multipoint VPN)
- **Multi-Access**
  - Other routers are available on the same network through one router's interface
- Transitivity is not guaranteed



NBMA Networks ②

- *It is necessary to know who with whom might/would like to communicate in NBMA networks!*
- Multiple routing protocols need additional configuration to be properly working in NBMA networks
  - Split-horizon rule correction
  - Defining directly connected neighbors
  - Correction of next-hop router addresses
  - For OSPF also influencing of DR/BDR election

Where to Seek Further???

- Doc ID 8651: „Route Selection in Cisco Routers“
- Doc ID 5212: „How Does Load Balancing Work?“
- Doc ID 16448: „Configuring a Gateway of Last Resort Using IP Commands“

Few Facts about Routing Protocols

- *The main goal of routing protocols is to feed routing table with available routes with the best metrics!*
- Each routing protocol has its own topology database from where routes are installed to routing table
- Routing protocol sends in updates:
  - directly connected networks specified with **`network`** command
  - other networks learned from same routing protocol neighbors
- Content of routing table is the result of running routing algorithm above routes in topology database

Administrative Distance ①

- Every routing protocol inserts to routing table routes with lowest possible metric
  - Metric is criteria for decision which route is best
  - *Lower means better*
- Multiple different routing protocols could run on router
  - *…but theirs metrics are incomparable*
- *This is the reason why* **administrative distance** *exists!*
  - AD is measurement of trustworthiness of information about network
  - Lesser AD is, more trustworthy is information
- `IF` there are multiple sources of network information which satisfy condition to insert route into the routing table `THEN`
  - firstly AD is compared
  - afterwards the best metric is resolved

Administrative Distance ②

Route origin	Cisco default ADs
Directly connected	0
Static	1
EIGRP summary	5
BGP external	20
EIGRP internal	90
OSPF	110
IS-IS	115
RIP	120
ODR	160
EIGRP external	170
BGP internal	200
DHCP	254
Totally unreliable source	255

Asymmetric Routing

- Routing protocol could insert multiple same route records into the routing table
  - Typically when they have the same (and lowest) metric
  - EIGRP could insert routes with different metrics
  - *Why should they do this?*
- Multiple records to the same network could be used for load-balancing
  - Maximally 16 records per one network (IOS and platform dependent)
  - IGP has 4 records per network by default
    - Could be changed with command **`maximum-paths`**
  - BGP has only 1 record by default

Static Routing

Static Routing

- *It's root of all routing…*
- Content of routing table is defined by administrator
- Unfortunately in this case routing table is NOT flexible, it doesn't converge according to current network topology
- Useful for stub networks
- Configuration snippet:

```
Router(config)#
① ip route NET MASK NEXTHOP [AD] [permanent]
② ip route NET MASK IFACE [AD] [permanent]
③ ip route NET MASK NEXTHOP IFACE [AD] [permanent]
```

Outgoing Interface in Static Route ①

- *DO NOT DO THAT!!!*
- Technically it advertises that destination network is directly connected to this outgoing interface…
  - …which is usually not true and could lead to awful troubles
- Ethernet example
  - For every recipient router consults its ARP cache
  - Whenever there's no record in ARP cache, router generates ARP Request and awaits ARP Response
  - If router couldn't resolve IP/MAC packet router would drop packet
- *What if Proxy ARP is turned on?*
  - Proxy ARP isn't solution – big ARP traffic means huge ARP cache

Outgoing Interface in Static Route ②

- Multipoint Frame Relay
  - IP/DLCI map table lookup for every IP address
  - `IF` there's no match `THEN` packet is drop
- ISDN BRI (Legacy DDR)
  - Works good only for default route
  - Any other network couldn't be translated to telephone number and therefor the packet is dropped
- Conclusion:
  - *Once again do not do that! Only exception could be point-to-point links. But why bother when there's working equivalent?*

Floating Static Route

- = is static record with AD purposely higher then usual
- Leaked to routing table only when the route with lower AD becomes invalid
- Typically used for backup links
- *What if there is tie between static and dynamic route?*
  - Static routes are more preferred than routes learned via routing protocol
  - *Why?*
  - Static route records have internal metric 0

RIPv2 and RIPng

Routing Information Protocol

- *Grandfather of all distance-vector protocols*
- Currently there are three versions available
  - RIPv1: Historic, classful, RFC 1058
  - RIPv2: RFC 2453
  - RIPng: RFC 2080
- It's still used because of it's ease of deployment, it's also open standard and it has wide vendor support
- Despite gossips and false prophets that *"RIP is dead! R.I.P"*
  - It's ideal for small networks
  - Perfect for CE/PE information exchanges

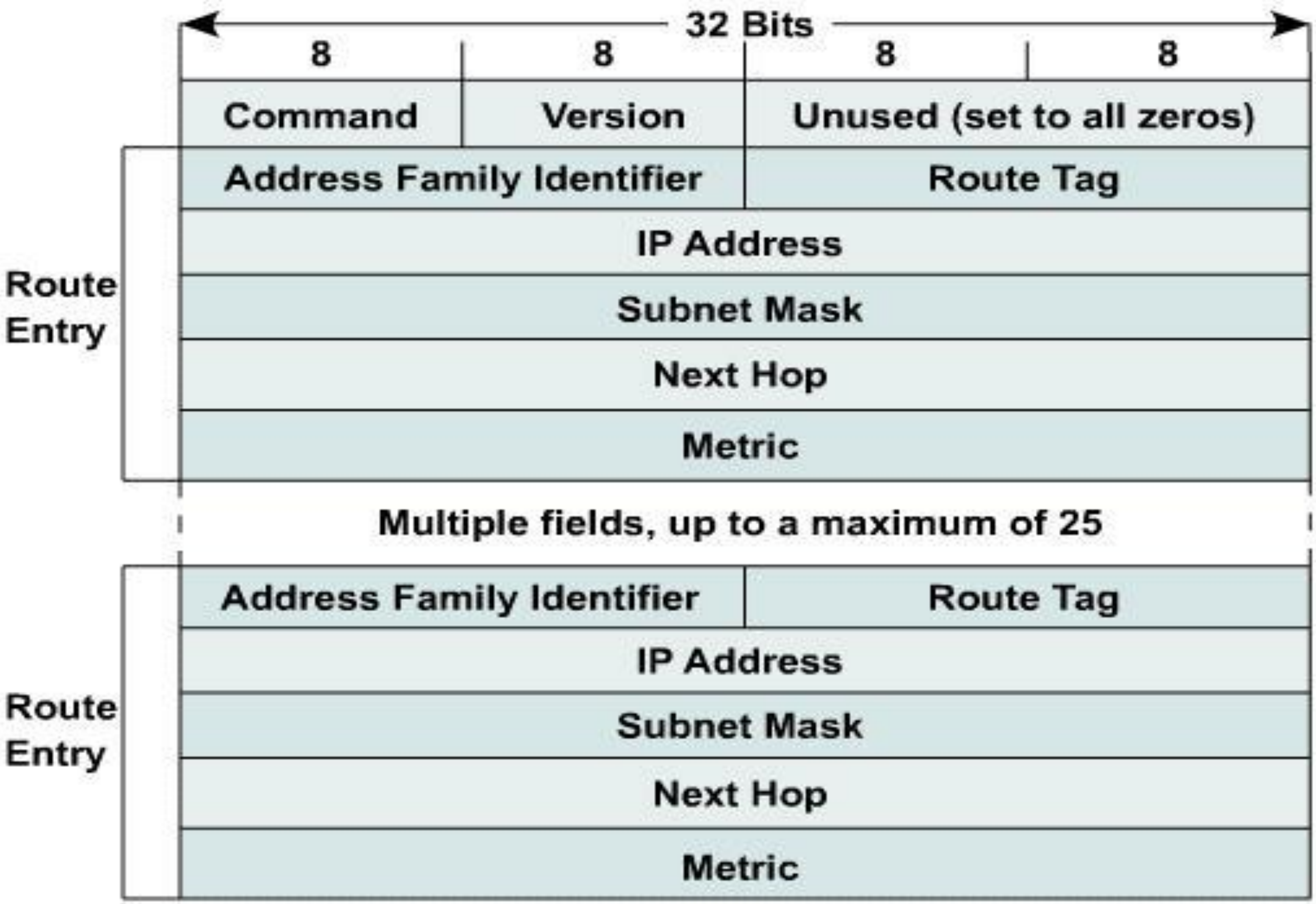
RIPv1 and RIPv2 Compare

- RIPv1:
  - Classful (Document ID 13723)
  - Metric is number of hops – 15 maximally
  - UDP/520, updates send periodically every 30 seconds as limited broadcast on address 255.255.255.255
- RIPv2 key changes:
  - Classless
  - UDP/520, updates send periodically every 30 seconds on multicast address 224.0.0.9
  - Authentication
  - Route tagging

RIPng Compare

- RIPng key changes:
  - Classless as RIPv2
  - UDP/**521**, updates send periodically every 30 seconds on multicast address FF02::9
  - Authentication is not supported – the idea was that IPSec will be used
  - Route tagging

RIPv2 Packet Format



RIPv2 Configuration

- Basic configuration guide:

```
Router(config)# router rip
Router(config-router)# no auto-summary
Router(config-router)# version 2
Router(config-router)# network …
Router(config-router)# network …
```

- Meaning of the **`network`** command:
  - To which directly connected network RIP sends packets
  - From which directly connected network RIP accepts packets
  - Which directly connected network RIP advertises to neighbors
- Distance-vector protocols consider even static routes with outgoing interface as “directly connected networks”

RIPv2 Default Route

- RIP enables to distribute default route
- Configuration snippet:

```
Router(config)# router rip
Router(config-router)# default-information originate
```

- Router with this configuration generates this route DESPITE the fact whether it has default route in its routing table or not
- Configure it only on border routers which interconnect our network with other one
  - Inner routers chose route to the closest border router
- Known bug in IOS RIP implementation when it stucks and not generate default route:

```
Router# clear ip route *
```

Compatibility of RIPv2 with RIPv1

- Backward compatible
  - Without **version** command:
    - Sending version 1
    - Accepting version 1 and also 2
  - With **version** command:
    - Send and accept just configured version
- Use following configuration whenever it's necessary to enforce preferred version on interface:

```
Router(config-if)# ip rip send version {1 | 2 | 1 2}
Router(config-if)# ip rip receive version {1 | 2 | 1 2}
```

RIPv2 Authentication ①

- *Without authentication of sender RIP blindly trust every packet it accepts!*
- Authentication
  - Every packet is “signed” by mutual agreed password
  - By RFC two forms of authentication – plain text or MD5 hash
- Configuration guide:
  1. Creation of “keychain” – list of keys
  2. Activation of authentication form on interface
  3. Activation of keychain on interface

RIPv2 Authentication ②

1. Creation of keychain:

```
Router(config)# key chain NAME
Router(config-keychain)# key NUMBER
Router(config-keychain-key)# key-string PASSWORD
```

2. Activation of authentication form:

```
Router(config-if)# ip rip authentication mode {md5|text}
```

3. Activation of keychain:

```
Router(config-if)# ip rip authentication key-chain NAME
```

RIPv2 Authentication ③

- Key rings names could differ but key numbers MUST be identical (key number is part of every message)!
- Every RIP message sent/received on interface is signed/checked with appropriate key
  - On multi-access segment all routers have to have same key
- *But what about case when we're using multiple keys?*
  - Every key has tuple of parameters
    - send-lifetime – validity of key for signing outgoing messages
    - accept-lifetime – validity of key for checking incoming messages
  - Whenever there are multiple keys valid for sending (their send-lifetimes are sounding), the key with lowest number will be used

RIPv2 Authentication ④

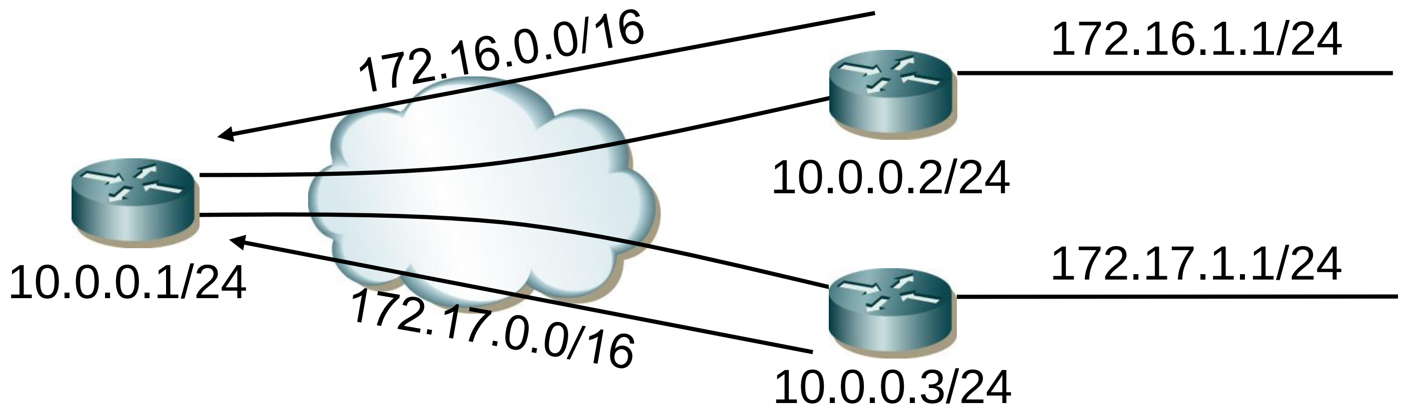
- Design-guide how to swap for new key:
  1. Add new key to key ring with right password string and higher number on all affected routers
     - Routers are still authenticating with the same old key all outgoing and incoming RIP messages
  2. Set send-lifetime of old key to past on all routers
     - One by one routers start using new key to authenticate outgoing packets
     - Nevertheless not yet reconfigured routers are working - because they are sending messages with old key and receiving messages with new key
     - At the end of 2nd step all routers are using new key and none is using the old one
  3. Delete old key from key ring on all routers

Summarization

- Multiple more specific networks (components) are described by one less specific (summary) record
- Summarization could effectively reduce size of routing tables when used together with right address plan
- Summarization happens when sending routing information, never when receiving them!
- Types of summarization on Cisco devices:
  - **Automatic**
  - **Manual**

RIPv2 Automatic Summarization

- Major network summarization (according to IP address class)
- *Router substitutes component with summary record whenever sending information about component of the one major network through interface to another major network!*



RIPv2 Manual Summarization ①

- *Router substitutes advertised network with configured summary network address and subnet mask*
- Networks without summary configured are sent unchanged
- Limitations of Cisco RIP implementation:
  - Every summary network address MUST belong to different major network
  - Supernetting (aggregation of classful networks) isn’t allowed

RIPv2 Manual Summarization ②

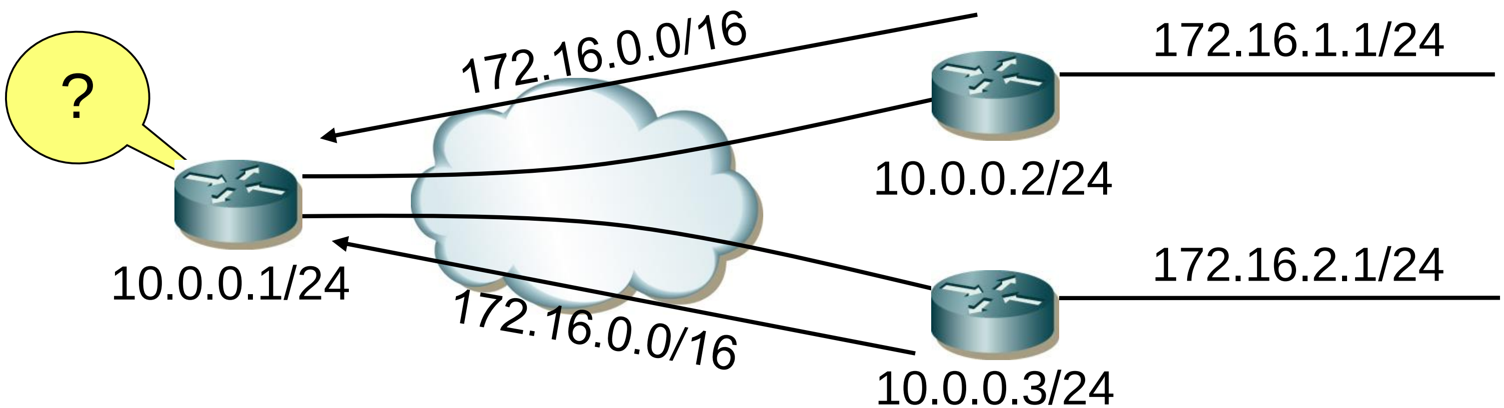
- Configuration snippet of manual summarization:

```
Router(config-if)# ip summary-address rip NET MASK

Router(config-if)# router rip
Router(config-router)# no auto-summary
```

- Automatic summarization MUST be turned off otherwise it has priority above manual summarization
- `no auto-summary` *is strongly advised as first step of distance-vector routing protocols configuration!!!*

RIPv2 Network Discontinuity

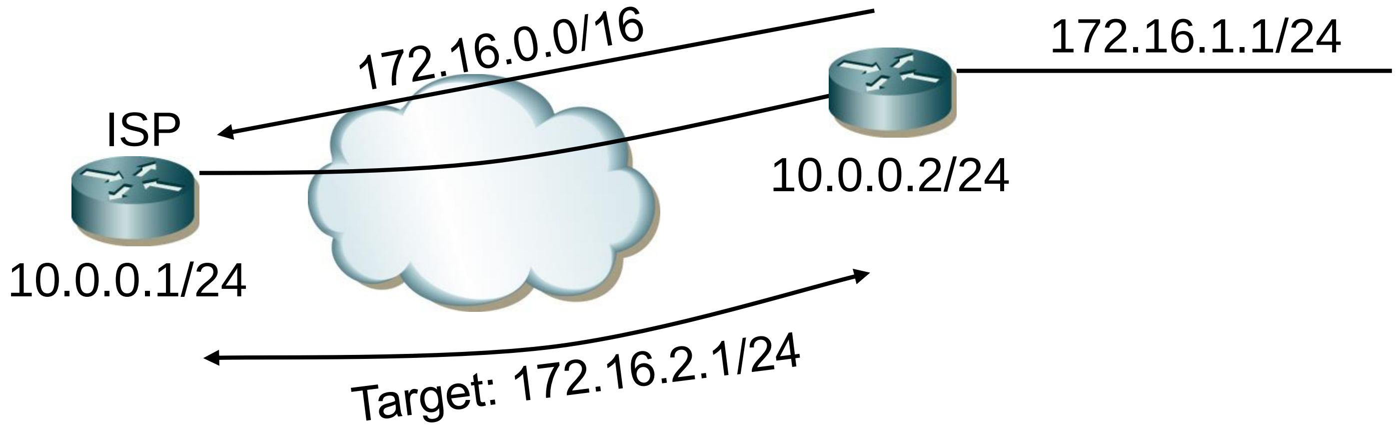
- Happens when improper (or even automatic) summarization is configured
- **Network discontinuity** is state when components of the one major network are located behind other intermediate major network



- *Routing table inconsistency is more than obvious consequence!*

RIPv2 Discard Route ①

- Scenario:
  1. Company router sends summary network towards ISP but the one of its component doesn't exist
  2. ISP isn't aware of this fact because of summarization. ISP is sending packets to this nonexistent network through company router
  3. Company router doesn't recognized component hence it's returning packets back to ISP via default route



RIPv2 Discard Route ②

- This routing loop could be eliminated with static routing by adding **discard route**:

```
Router(config)# ip route NETWORK MASK Null0
```

  where *NETWORK* and *MASK* are identical with summary

- Other routing protocols (EIGRP, OSPF, IS-IS, BGP) are adding discard route automatically
  - *Other nonsensical limitation of Cisco RIP implementation* ☹

RIPv2 NBMA Networks ①

- RIPv2 is sending messages on multicast address
  - *Why?*
  - Because it’s not necessary to know how many routers with whatever addresses are on same segment
- NBMA are by principle unable to deliver (and spread) multicast frames
- In that case it’s mandatory to configure all directly connected RIPv2 neighbors

```
Router(config)# router rip
Router(config-router)# neighbor …
```

RIPv2 NBMA Networks ②

- Theoretically it's not necessary to define all neighbors on point-to-point or multipoint FR links where IP/DLCI has flag broadcast
  - To configure neighbors is not a configuration fault
  - *„Premature optimization is the root of all evil." – D. E. Knuth*
- On multipoint FR links is important not to forget turn off split-horizon

```
Router(config-if)# no ip split-horizon
```

- Split-horizon for RIP is by default
  - disabled on physical FR interface
  - enabled on point-to-point a multipoint FR subinterfaces

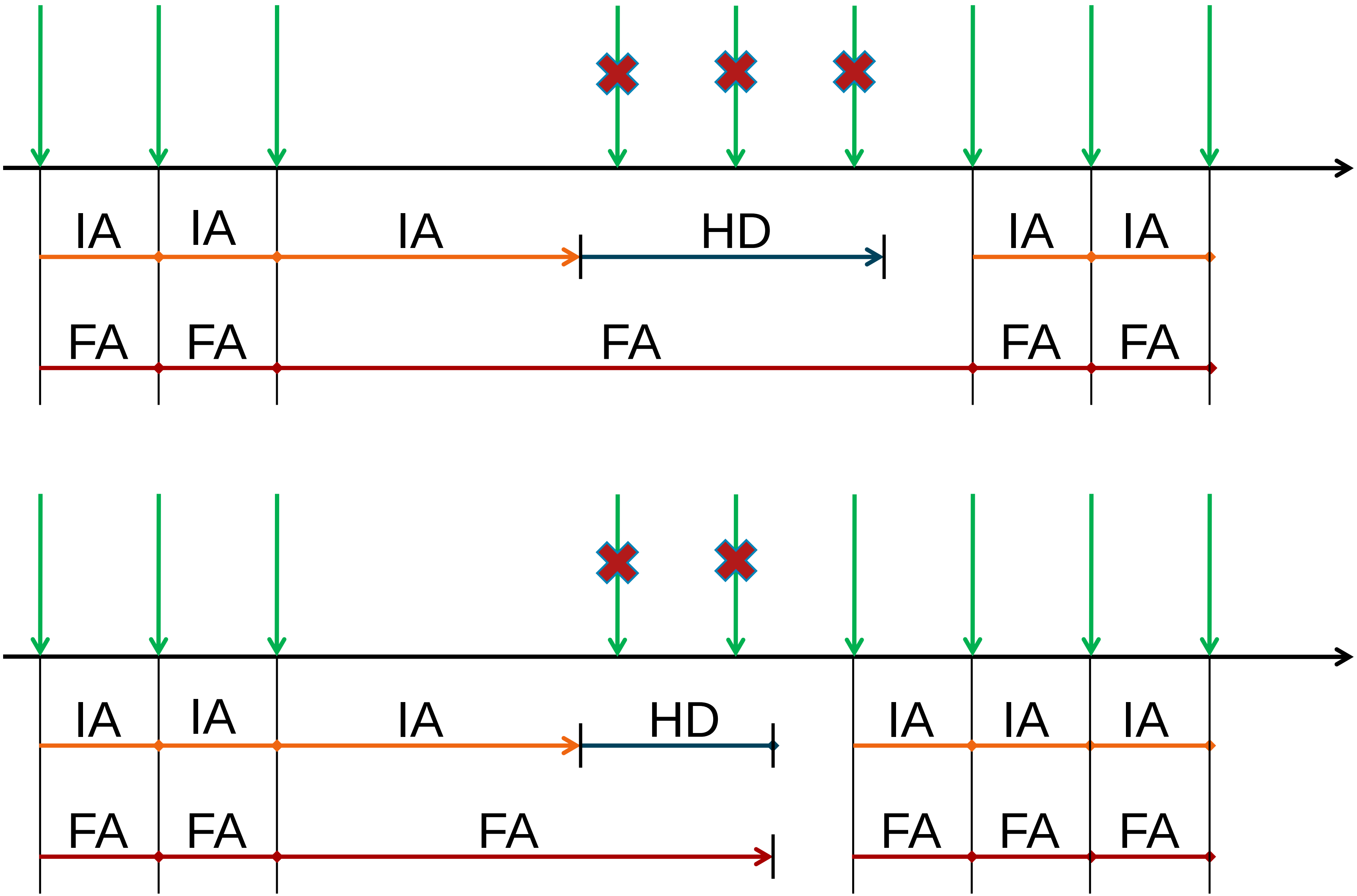
RIPv2 NBMA Networks ③

- Hub is advertising spoke networks with IP address of spoke as next-hop in hub-and-spoke topologies
  - *Bug or feature???*
  - There are no PVCs between spoke hence they are unable to communicate directly despite routing table is saying so
- Solution is to configure static IP/DLCI mapping on every spoke router via hub router

RIPv2 Timers ①

- **Update** (by default 30 seconds)
  - Period between two updates
- **Invalid after** (by default 180 seconds)
  - Maximal time between two consecutively received updates about same network after which route is considered unreachable
- **Holddown** (by default 180 seconds)
  - Interval of time during no updates about network is accepted
  - Route record remains in routing table and is being used but it is advertised as unreachable to neighbors
- **Flushed after** (by default 240 seconds)
  - Maximal time between two consecutively received updates about same network before it’s removed from routing table

RIPv2 Timers ②



RIPv2 Timers ③

- Timers have to be identical on all routers
- Configuration snippet for manipulating with timers:

```
Router(config)# router rip
Router(config-router)# timers basic UPD INV HOL FLU
```

$$Flushed\,after < Invalid\,after + Holddown$$

RIPv2 Useful SHOW Commands

```
show ip protocols
show ip interface
show ip rip database
show ip route A.B.C.D
show key chain
debug ip rip
debug ip routing
```

RIPng packet structure

- Packet structure is different compare to RIPv2
- Address family is missing, next-hop si optional

```
▷ Internet Protocol Version 6, Src: fe80::240:5ff:fea0:8e08, Dst: fe80::200:86ff:fe05:80fa
▷ User Datagram Protocol, Src Port: 521 (521), Dst Port: 52539 (52539)
◢ RIPng
    Command: Response (2)
    Version: 1
    Reserved: 0000
  ▷ Route Table Entry: IPv6 Prefix: 3ffe:501:404:100::/64 Metric: 4
  ▷ Route Table Entry: IPv6 Prefix: 3ffe:501:404::/64 Metric: 3
  ▷ Route Table Entry: IPv6 Prefix: 3ffe:501:405::/64 Metric: 3
  ▷ Route Table Entry: IPv6 Prefix: 2001:200:0:1c2b::1/128 Metric: 7
  ◢ Route Table Entry: IPv6 Prefix: 2001:200:0:1c2b::2/128 Metric: 7
      IPv6 Prefix: 2001:200:0:1c2b::2
      Route Tag: 0x0000
      Prefix Length: 128
      Metric: 7
  ▷ Route Table Entry: IPv6 Prefix: 2001:200:0:3c01::/64 Metric: 7
  ▷ Route Table Entry: IPv6 Prefix: 3ffe:501:0:1c01::/64 Metric: 6
```

RIPng configuration

- Configuration is done on interfaces, e.g.

```
Router(config-if)# ipv6 rip NAME enable
```

- Name, is the name of the RIP process
- If *NAME* doesn‘t exist, it is created, however, you can create it manually

```
Router(config)# ipv6 router rip NAME
```

RIPng configuration

- **`neighbor`** command and passive interface features are not supported
- Split-horizon can be disabled only for whole RIPng process

Appendix: Other interesting techniques and technologies

IP Unnumbered

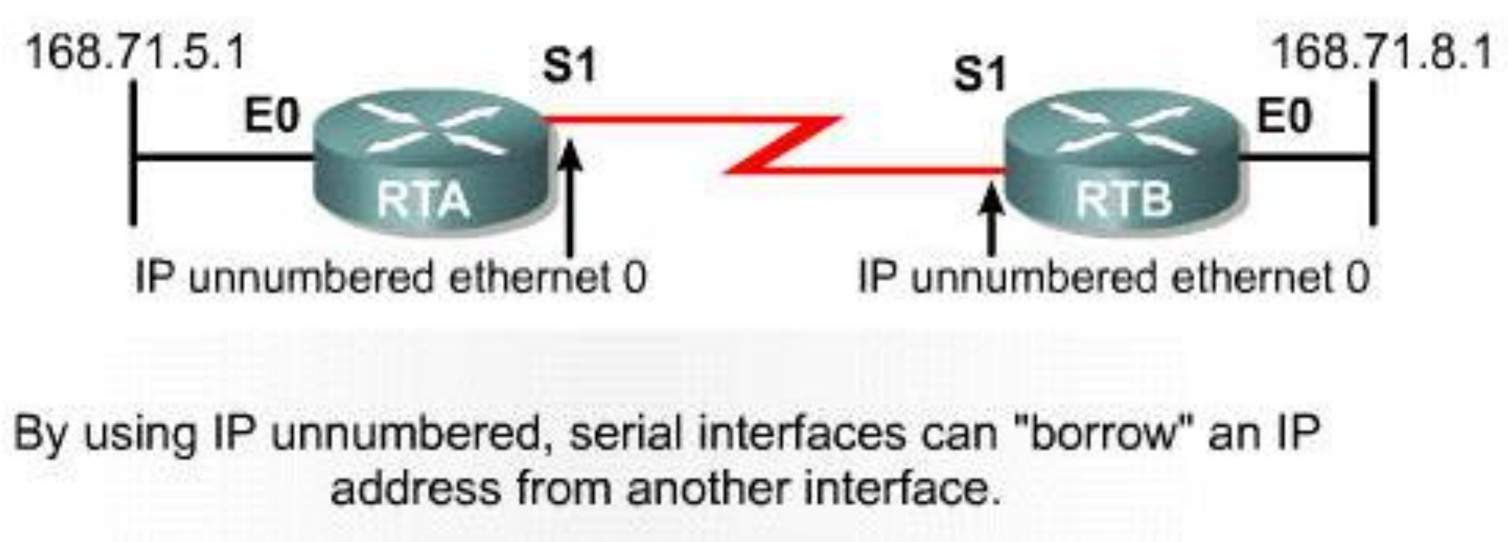
IP Unnumbered ①

- Document ID: 13786
- Point-to-point interfaces has specific nature
  - Recipient of data is certain – the one on the other side of cable
  - *Hence, interfaces theoretically doesn't even need IP address*
- **IP Unnumbered** is feature of point-to-point interface allowing them to borrow IP address from other interface
  - Effective usage of IP address space
  - Destination networks use name of outgoing interface as next-hop
- Disadvantages:
  - State of IP Unnumbered interface is dependent on state of "master" interface – ideally is to use Loopback
  - *You cannot test unnumbered interface! How to ping something that does not have even address?*

IP Unnumbered ②

- Configuration example:

```
RTA(config)# interface e0
RTA(config-if)# ip address 168.71.5.1 255.255.255.0
RTA(config-if)# no shutdown
RTA(config-if)# interface s1
RTA(config-if)# ip unnumbered e0
```



IP Unnumbered ③

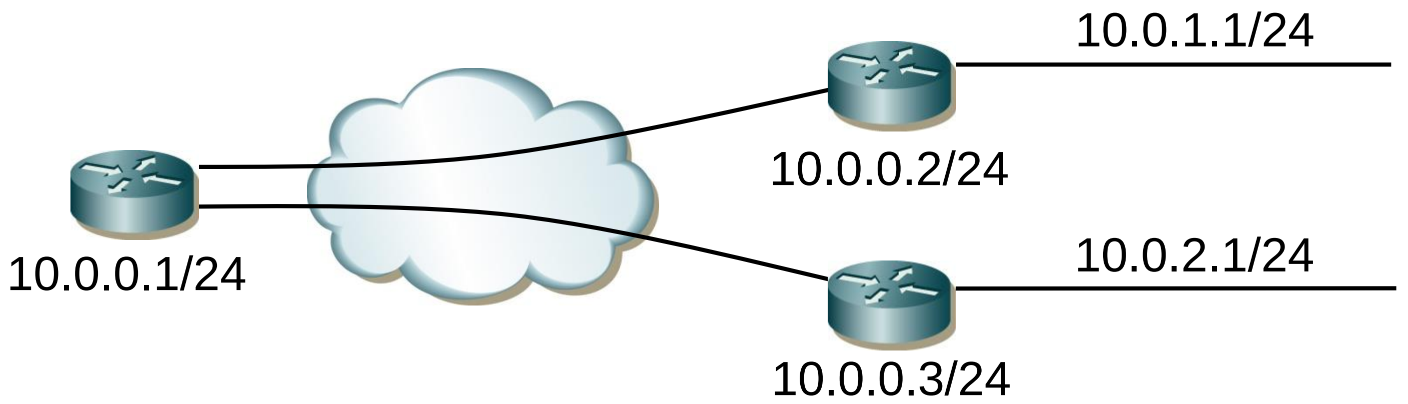
- IP Unnumbered is useful on following types of interfaces
  - Tunnel interface in MPLS-TE
  - Virtual Template interface from which other interfaces are cloned dynamically (e.g. PPPoE, PPPoA)

- Notice that IP Unnumbered technically allows that both ends of link could be in different networks

On-Demand Routing

On Demand Routing ①

- *Surprisingly many networks are designed in hub-and-spoke topology (the simplest star topology design)*
- **Spoke router**
  - Behind this router are **stub networks**
  - This kind of router needs just default route
- **Hub router**
  - Has list of all networks connected via stub routers



On Demand Routing ②

- Document ID: 13710, 13716
- Cisco proprietary limited routing ability inside CDP protocol
- Principle
  - Hub router sends default route to spokes
  - Spoke routers send hub list of all directly connected networks
- ODR is exclusively configured only on hub router
- Spoke routers NEED NOT to run any routing protocol
- Configuration snippet:

```
Hub(config)# router odr
Hub(config-router)# network …
```

On Demand Routing ③

- There's no option to redistribute routing protocol into ODR
- ODR is dependent on CDP
  - To fasten its convergence use **`cdp timer 5`**
  - On client side of network turn it off with **`no cdp run`**
  - Open standard variant of CDP is called LLDP
- Frame Relay ODR considerations
  - CDP is disabled on multipoint links by default
  - CDP is enabled on Point-to-Point links by default

/31 Mask on Point-to-Point Links

Mask /31 on Point-to-Point Links

- Serial links are usually addressed with /30 mask
  - *It's awful wasting and travesty – there's no need for broadcast on link with just two devices (one sender and one receiver)!*
- RFC 3021 specifies /31 mask address which allows to configure network with just and only two endpoints

```
Router(config-if)# ip address A.B.C.D 255.255.255.254
```

- E.g.:
  - 10.0.0.0/31 a 10.0.0.1/31
  - 192.0.2.254/31 a 192.0.2.255/31
- This feature is available since IOS version 12.2(2)T
  - No special configuration requirements
  - Warning on multiple-access links

Slides adapted by Matěj Grégr and Vladimír Veselý
partially from official Cisco course materials
but the most of the credit goes to CCIE#23527 Ing. Peter Palúch, Ph.D.

Last update: 2016-02-16