BGP Basics

WEBINAR COURSE

Overview

- What is BGP?
- BGP Features
- Path Vector Routing Protocol
- BGP General Operation
- BGP Terminology
- Inserting Prefixes into BGP

What is BGP?

Interior Gateway Protocol	Exterior Gateway Protocol
OSPF, IS-IS …	BGP

- **B**order **G**ateway **P**rotocol
- A Routing Protocol used to exchange routing information between different networks
  - Exterior gateway protocol
- Described in RFC4271
  - RFC4276 gives an implementation report on BGP
  - RFC4277 describes operational experiences using BGP

BGP Features

- Path Vector Protocol
- Incremental Updates
- Many options for policy enforcement
- Widely used for Internet backbone
- Autonomous systems
- Classless Inter Domain Routing (CIDR)

What is an Autonomous System?

- Group of Internet Protocol-based networks with the same routing policy
  - Usually under single ownership, trust or administrative control
- The AS is used both in the exchange of exterior routing information (between neighboring ASes) and as an identifier of the AS itself
- The Autonomous System is the cornerstone of BGP
  - It is used to uniquely identify networks with a common routing policy

Autonomous System Number (ASN)

- Globally unique identifiers for IP networks
- ASN uniquely identifies each network on the Internet, allocated to each Autonomous System (AS) for use in BGP routing
- 2-byte only AS number range : 0 – 65535
- 4-byte only AS number range : 65,536 - 4,294,967,295

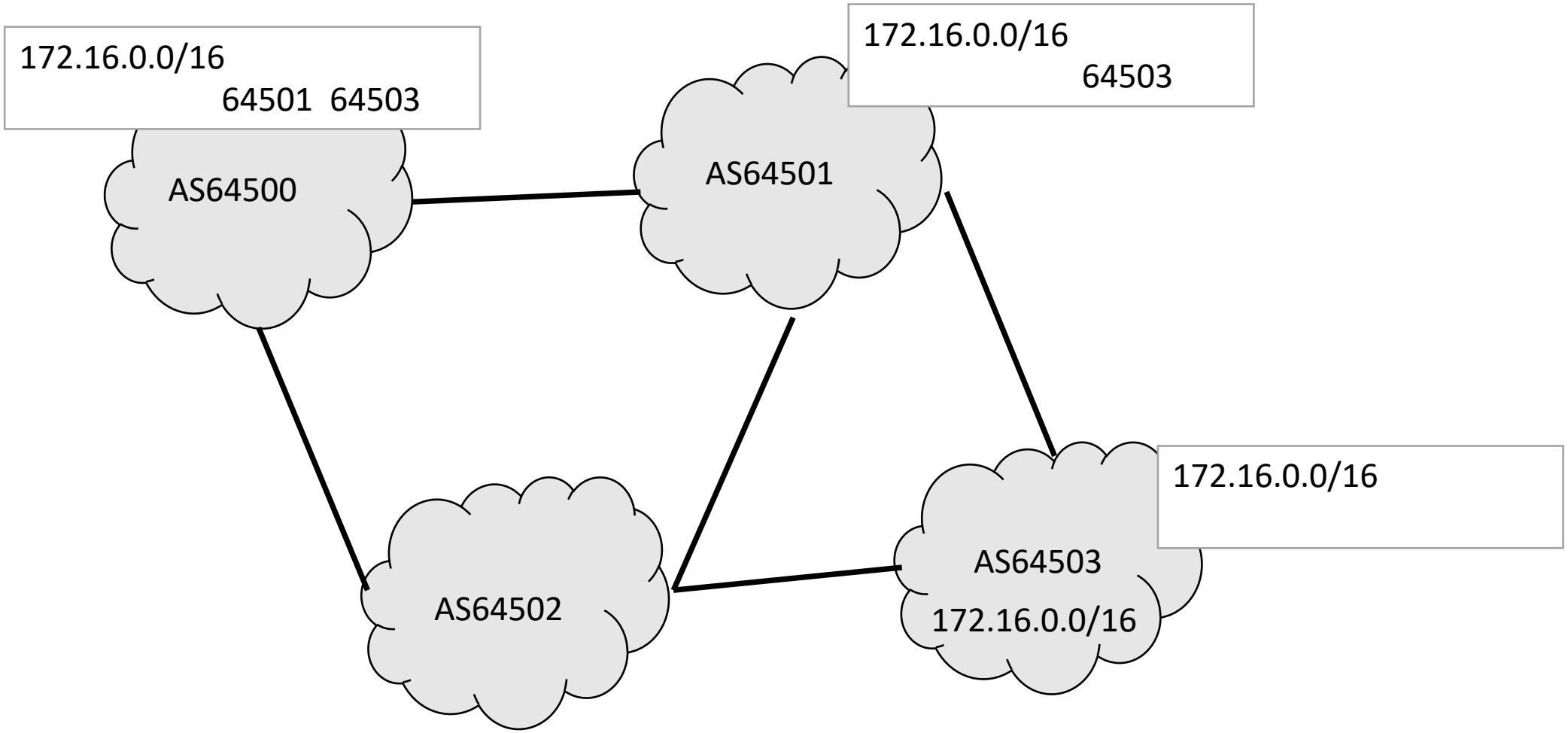
What is Path Vector Routing Protocol

- A path vector routing protocol is used to span different autonomous systems
- It defines a route as a collection of a number of AS that it passes through from source AS to destination AS, i.e.

  {65001 65002 65003 65007}

- This list of AS numbers is called AS path and used to avoid routing loop
- AS path is also used to select path to destination

Path Vector Routing Protocol

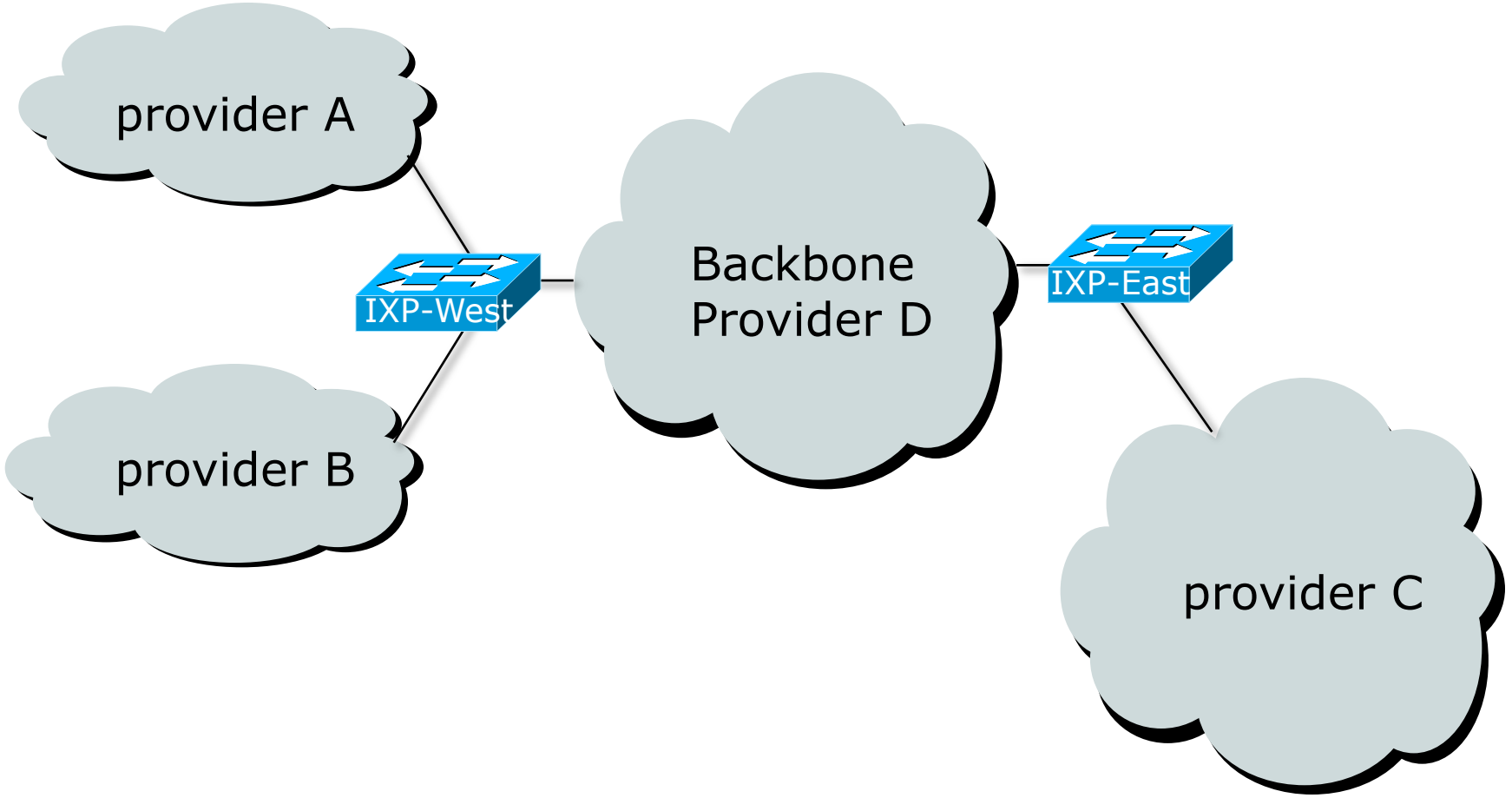


Definitions

- Transit
  - carrying traffic across a network, usually for a fee
- Peering
  - exchanging routing information and traffic

Peering and Transit example

A and B can peer, but need transit arrangements with D to get packets to/from C



BGP General Operation

- Learns multiple paths via internal and external BGP speakers
- Picks the best path and installs it in the routing table (RIB)
- Best path is sent to external BGP neighbours
- Policies are applied by influencing the best path selection

BGP Attributes

- Well-known attributes – must be supported by every BGP implementation
  - Mandatory attributes – must be included with every route entry. If one attribute is missing, it will result in an error message
  - Ex: ORIGIN, AS_PATH, NEXT_HOP
  - Discretionary attributes – every BGP router must recognize, but they don't have to be present with every route entry
  - Ex. ATOMIC_AGGREGATE, LOCAL_PREF
- Optional attributes – not necessarily supported by all BGP implementations. It can be either transitive or non-transitive.
  - COMMUNITY, AGGREGATOR, MULTI_EXIT_DISC

Internal & External BGP

- **eBGP** used to:
  - Exchange networks/routes between ASes
    - Aggregates and sub-aggregates
  - Implement routing policies
    - To manipulate inbound and outbound traffic
- **iBGP** is used to:
  - Carry customer networks/prefixes
  - Internet routes (some or all) across the AS backbone

BGP Message Types

- **Open:**
  - After a TCP connection has been established between two BGP routers, an Open message is sent
    - Once the open message is confirmed (keepalive), the BGP session is established – become BGP peers/neighbors!
  - Contains:
    - Sender’s ASN
    - BGP version
    - BGP router ID
    - Hold-time (3 x keepalive interval)

BGP Message Types

- **Keepalive:**
  - Exchanged initially to acknowledge Open messages
  - Exchanged periodically (60 secs) to maintain BGP session
    - Dataless packet
- **Update:**
  - BGP peers exchange network information through Update messages
    - One update for each path!
  - Contains:
    - Withdrawn routes – no more reachable
    - Path attributes – attributes for this path to reach the destinations specified by the NLRI
    - NLRI – list of networks reachable through this path <prefix, length>

BGP Message Types

- **Notification:**
  - Sent when an error condition is detected
  - The BGP session is torn down immediately!
  - Contains:
    - Error code
    - Error sub-code
    - Data related to error

BGP Neighbor States

- A BGP router goes through six different states
  - Idle
    - The router is looking for a route to its neighbor
  - Connect
    - BGP router moves from Idle to Connect state if it has found a route to its neighbor, and has started the TCP handshake
    - If the TCP session successful, sends an Open message (and transitions to OpenSent)
    - Else, move to Active state

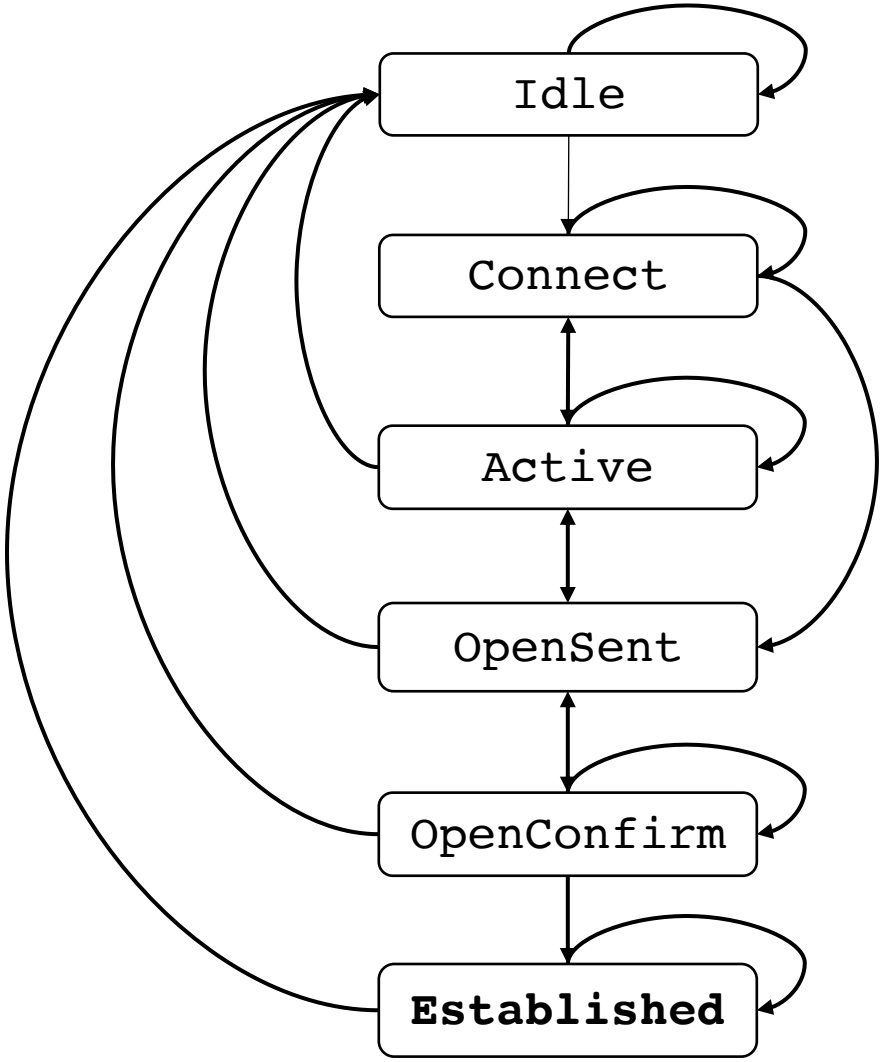
BGP Neighbor States

- Active
  - A router transitions to Active state if the initial TCP connection was not successful (in Connect state)
  - Restarts the TCP connection
  - If successful, sends an Open message
  - Else, falls back to Idle state
- OpenSent
  - An Open message has been sent to the neighbour
  - Waiting for Open message from neighbour
  - If it receives an Open message and there are no mismatches (version, source addr same as TCP addr, ASN, router-ID, TTL, md5), sends KeepAlive, moves to OpenConfirm
  - Else (if mismatches/errors), sent Notification and falls back to Idle

BGP Neighbor States

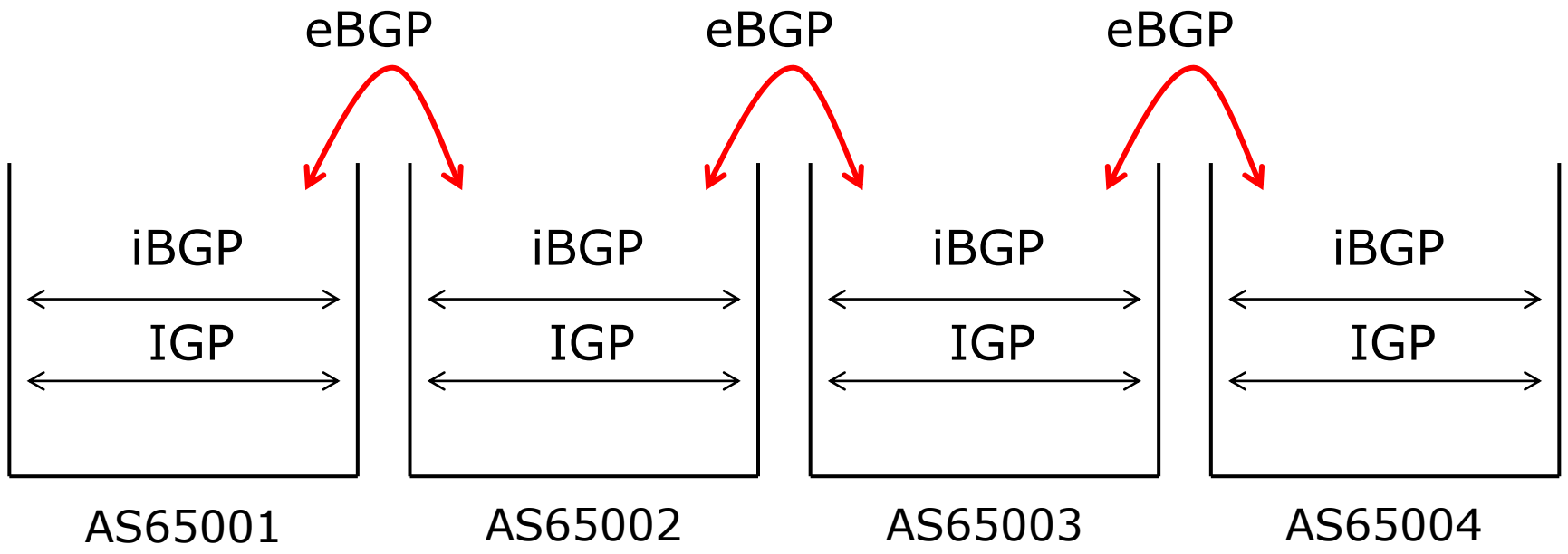
- OpenConfirm
  - waiting for the initial KeepAlive
  - If received, transitions to Established
  - If holdtimer expires or Notification received, moves to Idle
- Established
  - The BGP neighbor relationship (session) is established!
  - Routing information can now be exchanged
  - If holdtimer expires/error, moves back to Idle

BGP State Machine



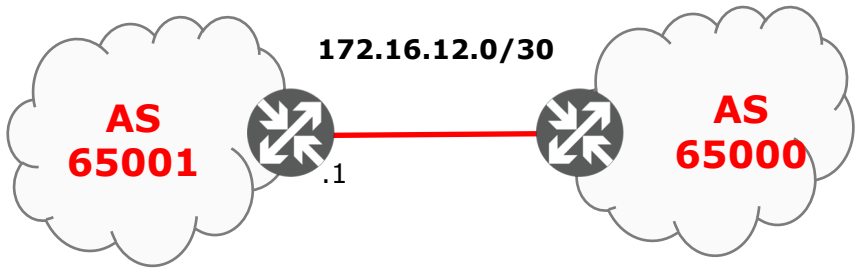
BGP/IGP model used in ISP networks

- BGP is used internally (iBGP) and externally (eBGP)
- iBGP – used to carry some/all Internet prefixes across ISP backbone and ISP's customer prefixes
- eBGP – used to exchange prefixes with other ASes and implement routing policy



eBGP Neighbor Relationship

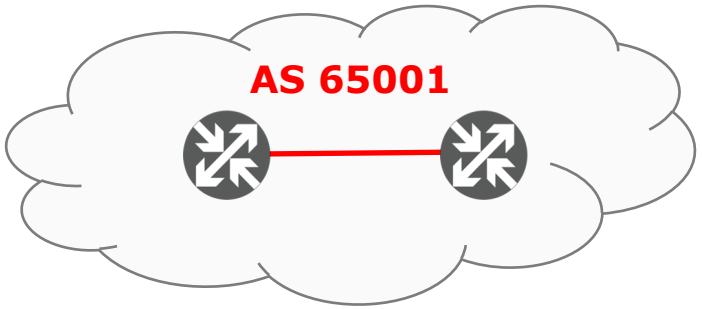
- eBGP neighbors/peers
  - BGP session established between routers in different ASes
  - Generally directly connected!
    - Session established using directly connected interface IP
    - Peering address must match the TCP session!
  - Else, we need a static route to reach the neighbor and change the eBGP TTL value (default 1)




```
router bgp 65001
 neighbor 172.16.12.2 remote-as 65000
!
 address-family ipv4
  neighbor 172.16.12.2 activate
 !
```

iBGP Neighbor Relationship

- iBGP neighbors/peers
  - BGP session established between routers within the same AS
  - Does not need to be directly connected
    - IGP ensure reachability (TCP connection)
  - Generally using loopback addresses

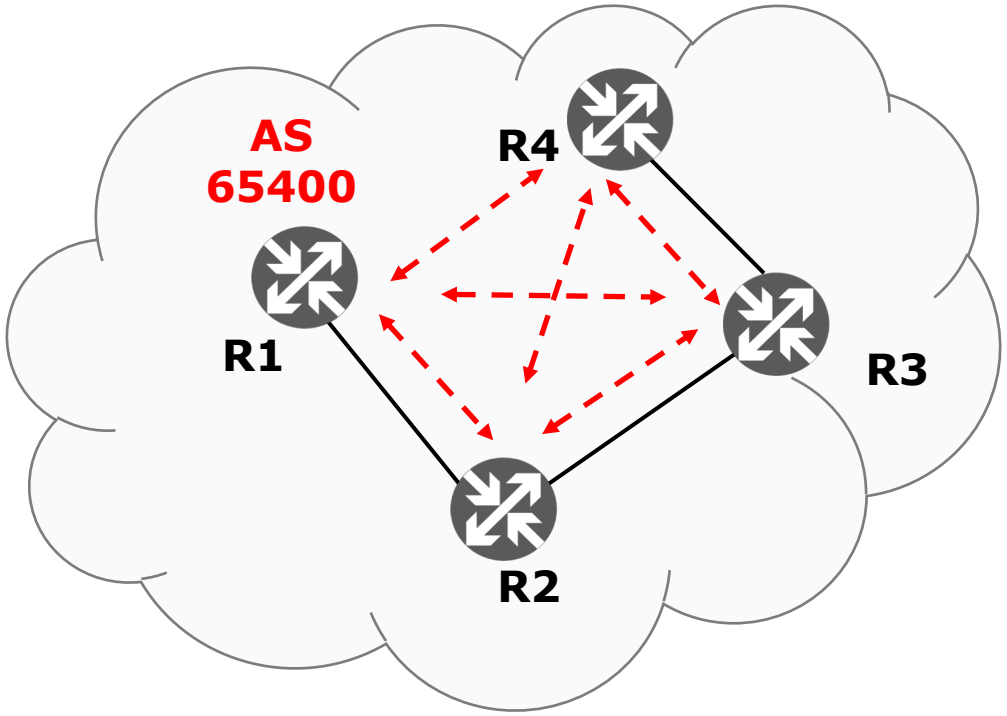



```
router bgp 65001
 neighbor 10.10.10.2 remote-as 65001
!
```

iBGP Operation

- iBGP routers must:
  - Originate directly connected routes
- Carry routes learned from outside the AS to all routers within the AS
  - Fully-meshed instead of redistributing!
  - Advertise routes learned from eBGP peers to all iBGP peers!
- To prevent routing loops (in a fully-meshed network)
  - iBGP routers are not allowed to advertise iBGP learned routes to other iBGP peers!

iBGP full-mesh



Example configuration on R1 and R2

R1:

```
router bgp 65400
 neighbor 10.10.10.2 remote-as 65400
 neighbor 10.10.10.3 remote-as 65400
 neighbor 10.10.10.4 remote-as 65400
 !
```

R2:

```
router bgp 65400
 neighbor 10.10.10.1 remote-as 65400
 neighbor 10.10.10.3 remote-as 65400
 neighbor 10.10.10.4 remote-as 65400
!
```

Sourcing iBGP from Loopback

- By default, routers use the exit-interface address as the source address for locally originated packets (updates)
  - If the BGP TCP session was established using any other interface (loopbacks) addresses, the source address for BGP updates must match!
- In Cisco IOS, the update-source loopback command achieves this

```
router bgp 65400
 neighbor 10.10.10.1 remote-as 65400
 neighbor 10.10.10.1 update-source loopback 0
!
```

Insert Prefixes into BGP

Examples in IOS	Function
`network 192.168.1.0 mask 255.255.255.0`	Add the specific route 192.168.1.0/24 into BGP routing table.
`redistribute OSPF`	redistribute all the routes in OSPF routing table into BGP routing table.

Inserting prefixes into BGP – network command

- Configuration Example

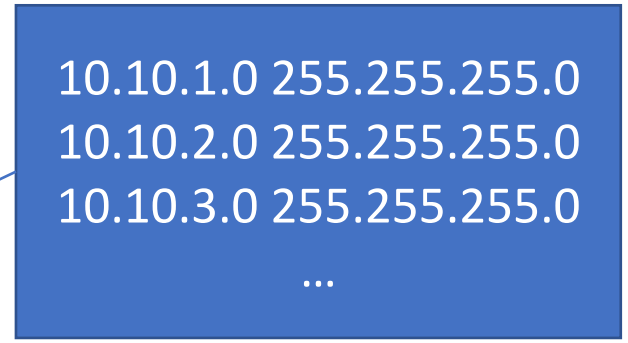
```
router bgp 65400
  network 10.10.32.0 mask 255.255.254.0
ip route 10.10.32.0 255.255.254.0
```

- A matching route must exist in the routing table before the network is announced
- Forces origin to be “IGP”

Configuring Aggregation – Network Command

- Configuration Example

```
router bgp 64500
 network 10.10.0.0 mask 255.255.0.0
ip route 10.10.0.0 255.255.0.0 null0
```



- A matching route must exist in the routing table before the network is announced
- Easiest and best way of generating an aggregate

APNIC Helpdesk Chat

Helpdesk

APNIC Helpdesk provides assistance to all on matters related to APNIC Services, such as membership and IP address enquiries.

APNIC Helpdesk offers (through prior arrangement) multi-language phone support for the following: Bahasa Indonesia, Bahasa Malaysia, Burmese, Cantonese, English, Filipino (Tagalog), Hindi, Japanese, Mandarin, Sinhalese, Tamil and Telugu.

You may also find our FAQs helpful with your enquiries.

Contact details

Helpdesk hours	09:00 to 21:00 (UTC +10) Monday – Friday (closed for some public holidays)
Chat	APNIC Live Chat Online Click here to chat
Skype	Call ID: apnic-helpdesk
Email	helpdesk@apnic.net
Phone	+61 7 3858 3188
VoIP	helpdesk@voip.apnic.net Using VoIP
Fax	+ 61 7 3858 3199

Service Updates

Upgrade edge router firmware

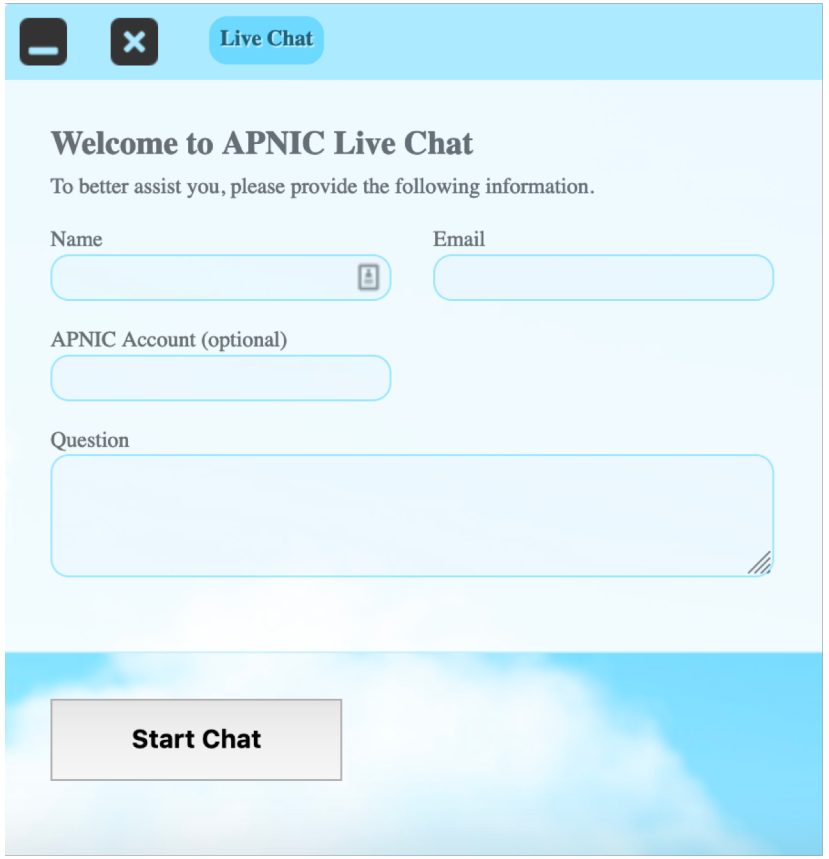
Start: Thursday, 31 January 2019 07:00 AM (UTC +10)
End: Thursday, 31 Jan 2019 08:00 AM (UTC +10)

This maintenance is required to upgrade our edge router firmware in DC2. There may be one or two interruptions to the services listed above for a maximum of 30 minutes within the change window.

More Updates

Subscribe to APNIC Service Announcements

Learn more about system maintenance



Acknowledgements

Some materials used in this course were originated from the Cisco ISP/IXP Workshop Programme developed by Philip Smith & Barry Greene.