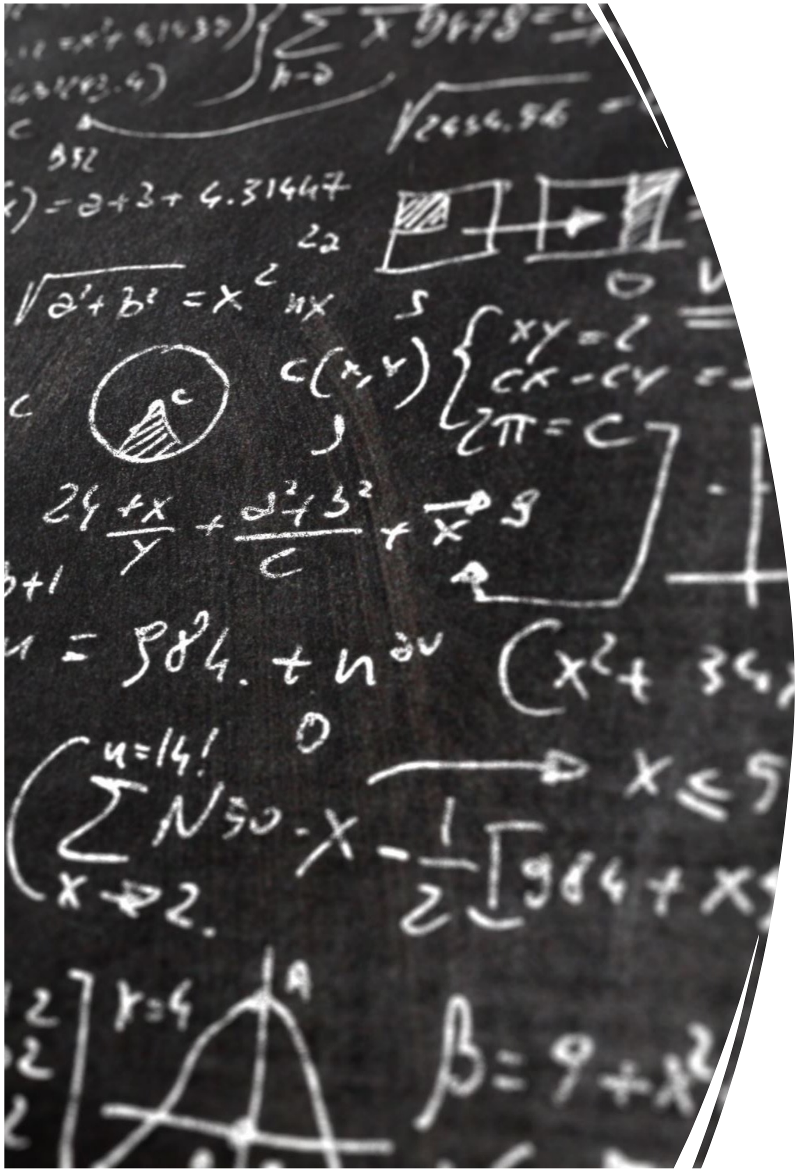
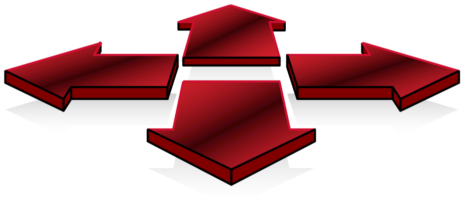


INTRODUCTION TO BGP

Routing versus Forwarding

- Routing = building maps and giving directions
- Forwarding = moving packets between interfaces according to the “directions”



Routing Table/RIB

Store of routing information

ROUTING TABLE IS MANAGED BY A ROUTING PROTOCOL (E.G. OSPF OR BGP)

OFTEN CALLED THE RIB – ROUTING INFORMATION BASE

EACH ROUTING PROTOCOL HAS ITS OWN WAY OF MANAGING ITS OWN ROUTING TABLES

EACH ROUTING PROTOCOL HAS A WAY OF EXCHANGING INFORMATION BETWEEN ROUTERS USING THE SAME PROTOCOL

Forwarding Table/FIB

Forwarding table determines how packets are sent through the router

Often called the FIB – Forwarding Information Base

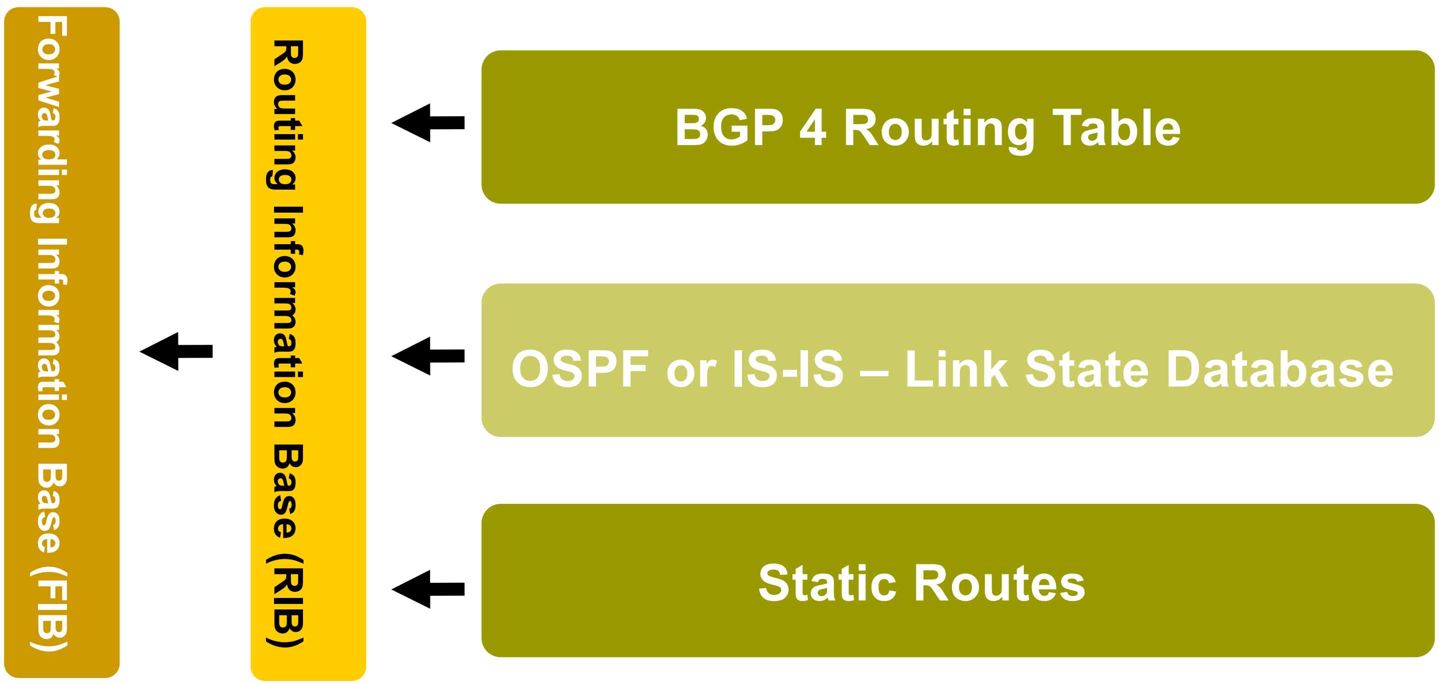
Made from routing table built by routing protocols

- Best routes from routing tables are installed

Performs the lookup to find next-hop and outgoing interface

Switches the packet with new encapsulation as per the outgoing interface

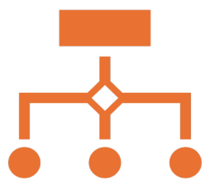
Routing Tables Feed the Forwarding Table



IP Routing

- Each router or host makes its own routing decisions
- Sending machine does not have to determine the entire path to the destination
- Sending machine just determines the next-hop along the path (based on destination IP address)
  - This process is repeated until the destination is reached, or there’s an error
- Forwarding table is consulted (at each hop) to determine the next-hop

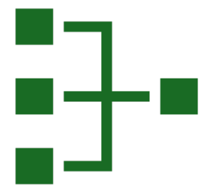
IP Routing



Classless routing

route entries include

- destination
- next-hop
- mask (prefix-length) indicating size of address space described by the entry



Longest match

for a given destination, find longest prefix match in the routing table

example: destination is 35.35.66.42

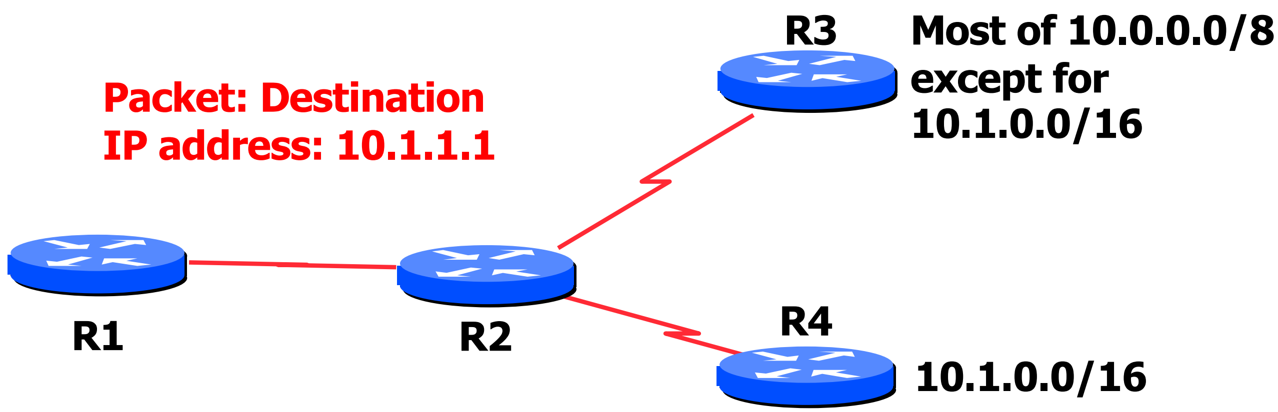
- routing table entries are 35.0.0.0/8, 35.35.64.0/19 and 0.0.0.0/0
- All these routes match, but the /19 is the longest match

IP routing

- Default route
  - where to send packets if there is no entry for the destination in the routing table
  - most machines have a single default route
  - often referred to as a default gateway

  - 0.0.0.0/0
    - matches all possible destinations, but is usually not the longest match

IP route lookup:
Longest match routing

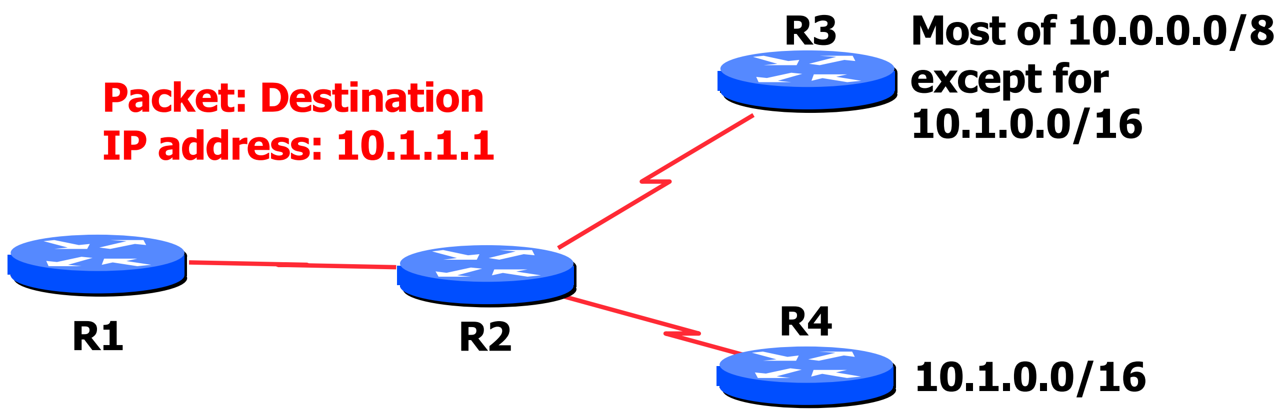


Based on destination IP address

R2's IP forwarding table

10.0.0.0/8 → R3
10.1.0.0/16 → R4
20.0.0.0/8 → R5
0.0.0.0/0 → R1

IP route lookup: Longest match routing

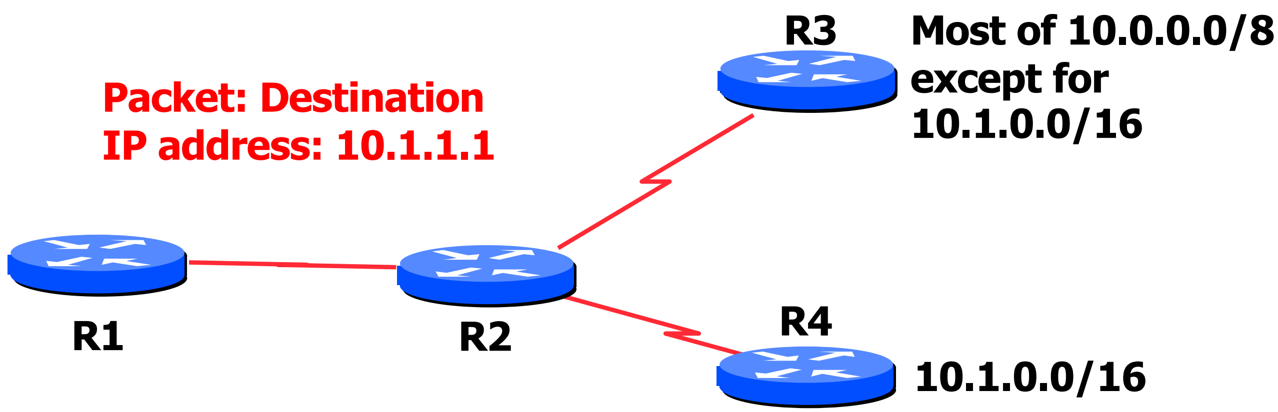


Based on destination IP address

R2's IP forwarding table

10.0.0.0/8 → R3 10.1.0.0/16 → R4 20.0.0.0/8 → R5 0.0.0.0/0 → R1	→ **10.1.1.1 & FF.00.00.00** **vs.** **10.0.0.0 & FF.00.00.00** **Match! (length 8)**

IP route lookup: Longest match routing



Based on destination IP address

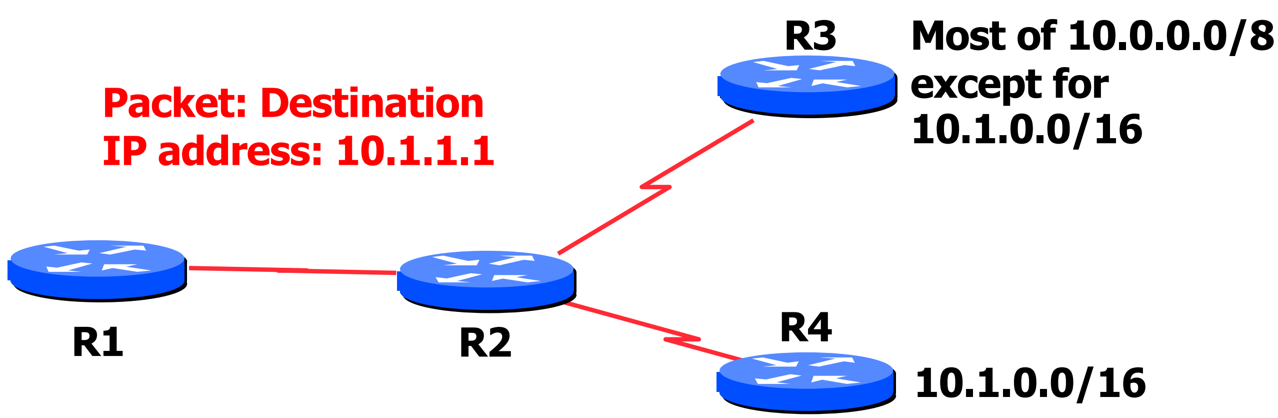
R2's IP forwarding table

10.0.0.0/8 → R3
10.1.0.0/16 → R4
20.0.0.0/8 → R5
0.0.0.0/0 → R1

→

10.1.1.1 & FF.FF.00.00
vs.
10.1.0.0 & FF.FF.00.00
Match! (length 16)

IP route lookup:
Longest match routing



Based on destination IP address

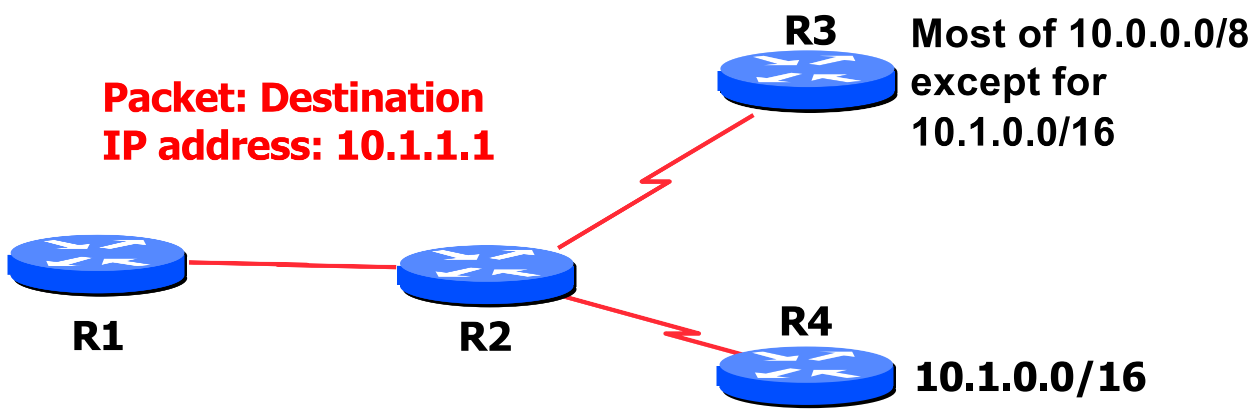
R2's IP forwarding table

10.0.0.0/8 → R3
10.1.0.0/16 → R4
20.0.0.0/8 → R5
0.0.0.0/0 → R1

→

10.1.1.1 & FF.00.00.00
vs.
20.0.0.0 & FF.00.00.00
No Match!

IP route lookup:
Longest match routing



Based on destination IP address

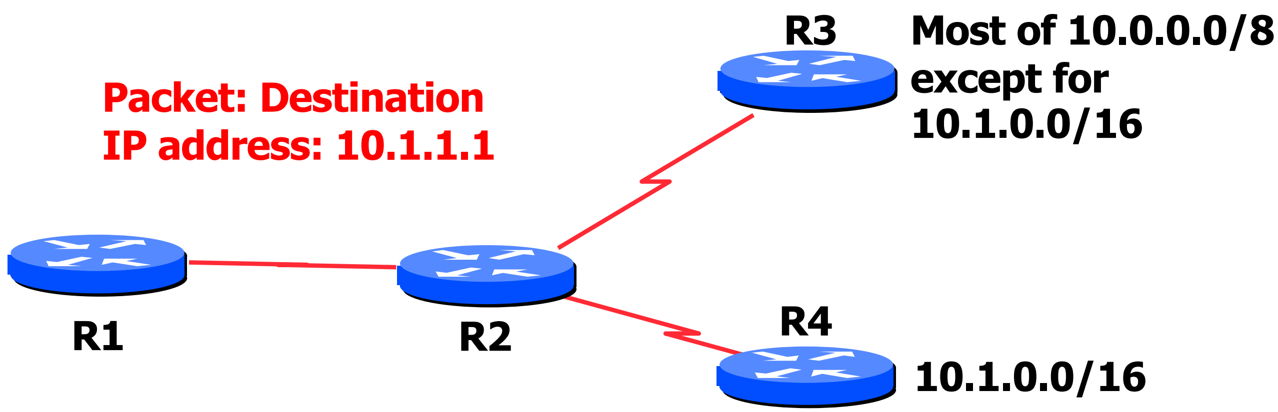
R2's IP forwarding table

10.0.0.0/8 → R3
10.1.0.0/16 → R4
20.0.0.0/8 → R5
0.0.0.0/0 → R1

→

10.1.1.1 & 00.00.00.00
vs.
0.0.0.0 & 00.00.00.00
Match! (length 0)

IP route lookup:
Longest match routing



Based on destination IP address

R2's IP forwarding table

10.0.0.0/8 → R3
10.1.0.0/16 → R4
20.0.0.0/8 → R5
0.0.0.0/0 → R1

This is the longest matching prefix (length 16). "R2" will send the packet to "R4".

IP route lookup: Longest match routing

- Most specific/longest match always wins!!
  - Many people forget this, even experienced ISP engineers
- Default route is 0.0.0.0/0
  - Can handle it using the normal longest match algorithm
  - Matches everything. Always the shortest match.

Static vs. Dynamic routing

- Static routes
  - Set up by administrator
  - Changes need to be made by administrator
  - Only good for small sites and star topologies
  - Bad for every other topology type

Dynamic routes
- Provided by routing protocols
- Changes are made automatically
- Good for network topologies which have redundant links (most!)

Dynamic Routing

ROUTERS COMPUTE ROUTING TABLES DYNAMICALLY BASED ON INFORMATION PROVIDED BY OTHER ROUTERS IN THE NETWORK

ROUTERS COMMUNICATE TOPOLOGY TO EACH OTHER VIA DIFFERENT PROTOCOLS

ROUTERS THEN COMPUTE ONE OR MORE NEXT HOPS FOR EACH DESTINATION – TRYING TO CALCULATE THE MOST OPTIMAL PATH

AUTOMATICALLY REPAIRS DAMAGE BY CHOOSING AN ALTERNATIVE ROUTE (IF THERE IS ONE)

Routing Protocols

Interior and Exterior Routing

Interior vs. Exterior Routing Protocols

- Interior gateway protocol (IGP)
  - Automatic neighbour discovery
  - Under control of a single organisation
  - Generally trust your IGP routers
  - Routes go to all IGP routers
  - Usually not filtered

Exterior gateway protocol (EGP)
- Specifically configured peers
- Connecting with outside networks
- Neighbours are not trusted
- Set administrative boundaries
- Filters based on policy

IGP

- Interior Gateway Protocol
- Within a network/autonomous system
- Carries information about internal prefixes
- Examples – OSPF, ISIS, EIGRP, RIP

EGP

- Exterior Gateway Protocol
- Used to convey routing information between networks/ASes
- De-coupled from the IGP
- Current EGP is BGP4

Why Do We Need an EGP?

Scaling to large network
- Hierarchy
- Limit scope of failure

Define administrative boundary

Policy
- Control reachability to prefixes

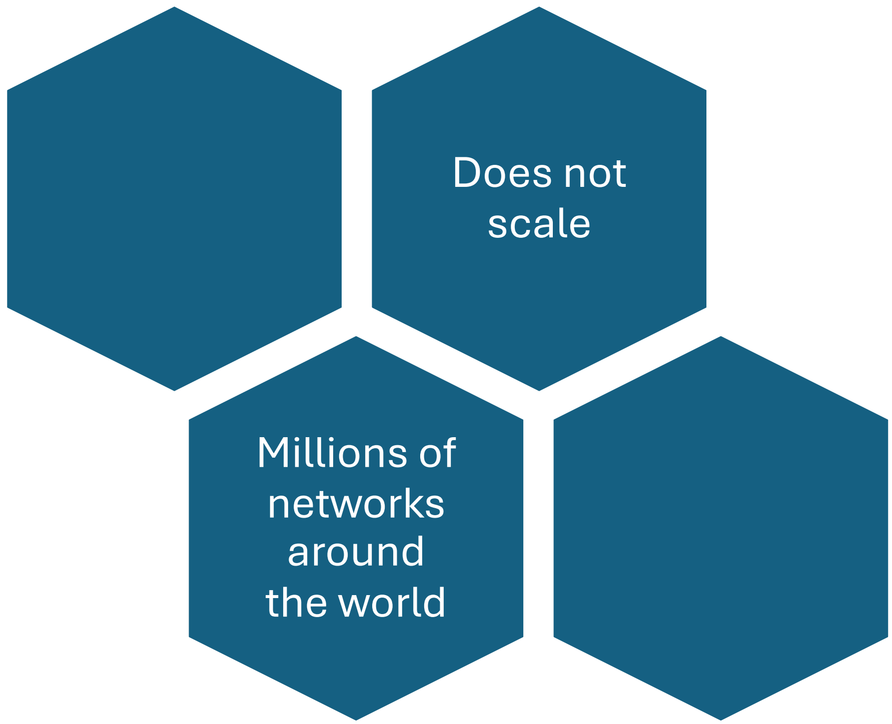
Scalability and policy issues

Just getting direct line is not enough

Need to work out how to do routing

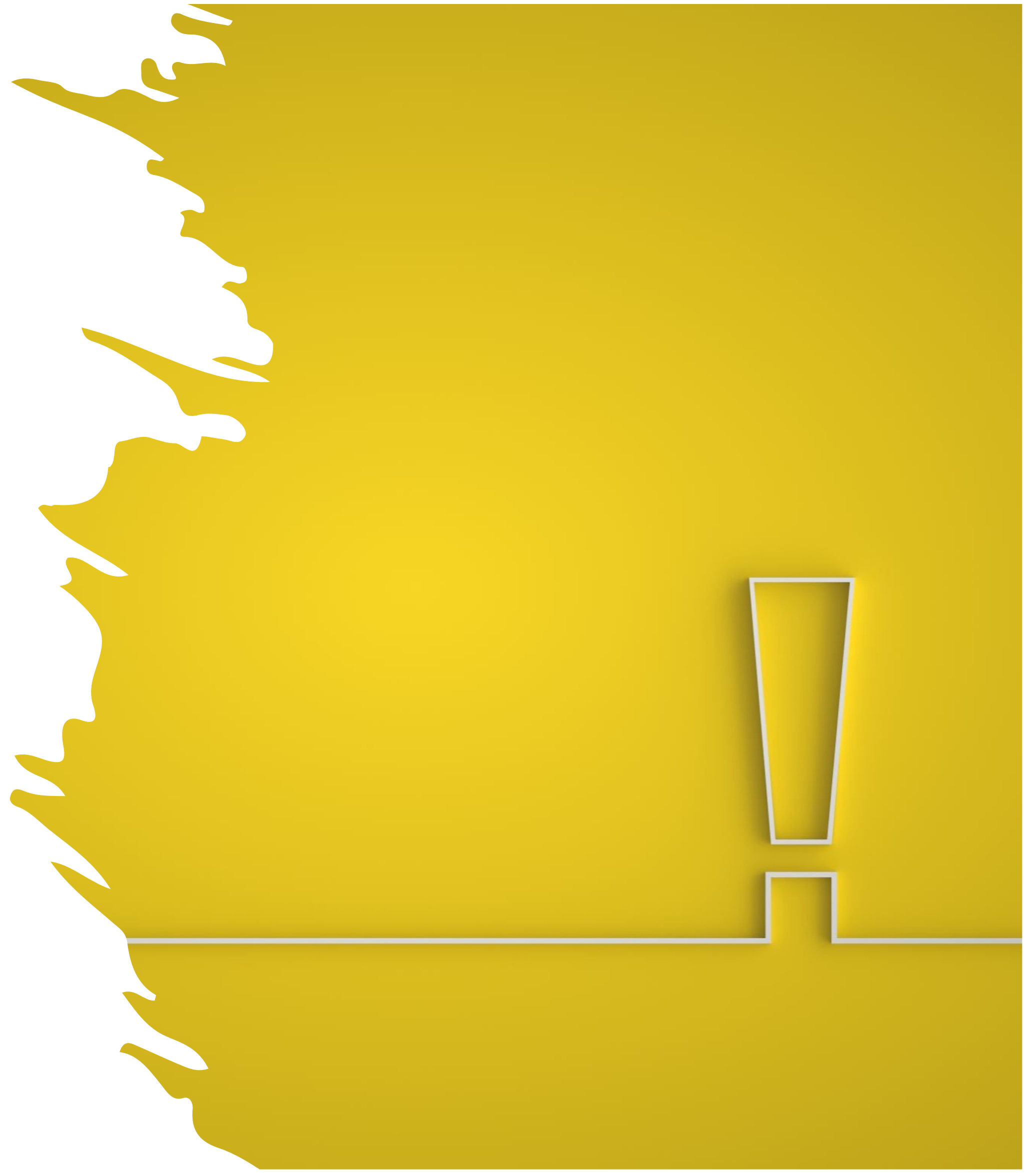
Need to get local traffic between ISP's/peers	Need to make sure the peer ISP doesn't use us for transit	Need to control what networks to announce, what network announcements to accept to upstreams and peers

Scalability:
Not using static routes



Scalability: Not using IGP (OSPF/ISIS)

- Serious operational consequences:
  - If the other ISP has a routing problem, you will have problems too
  - Your network prefixes could end up in the other ISP's network — and vice-versa
  - Very hard to filter routes so that we don't inadvertently give transit



Using BGP instead

BGP = Border Gateway Protocol

BGP is an **exterior** routing protocol

Focus on routing **policy**, not topology

BGP can make ‘groups’ of networks (Autonomous Systems)

Good route filtering capabilities

Ability to isolate from others' problems

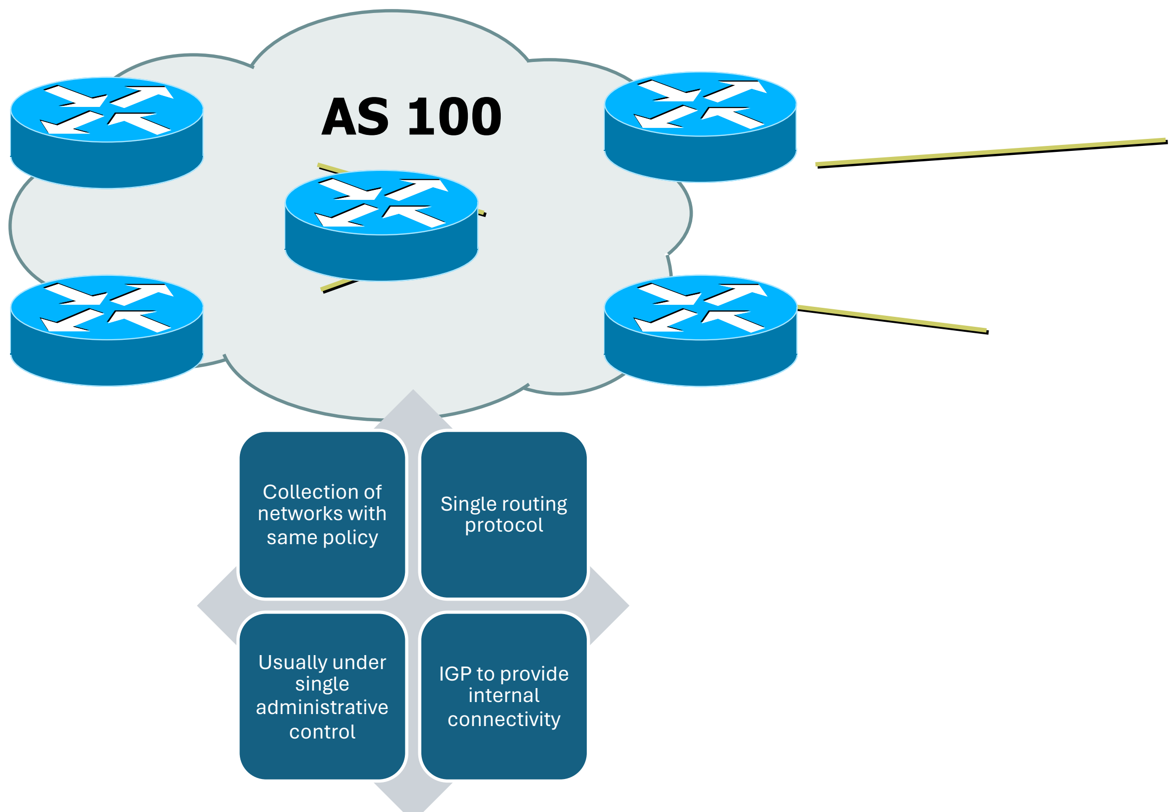
Border Gateway Protocol (BGP)

- A Routing Protocol used to exchange routing information between networks
  - exterior gateway protocol
- Described in RFC4271
  - RFC4276 gives an implementation report on BGP-4
  - RFC4277 describes operational experiences using BGP-4
- The Autonomous System is BGP's fundamental operating unit
  - It is used to uniquely identify networks with a common routing policy

BGP Building Blocks

- Autonomous System (AS)
- Types of Routes
- IGP/EGP
- DMZ
- Policy
- Egress
- Ingress

Autonomous System (AS)



Autonomous System (AS)

Autonomous systems is a misnomer

Not much to do with freedom, independence, ...

Just a handle for a group of networks that is under the same administrative control

Identified by an AS number

Autonomous System (AS)

- Identified by 'AS number'
  - example: AS16907 (ISPKenya)
- Examples:
  - Service provider
  - Multi-homed customers
  - Anyone needing policy discrimination for networks with different routing policies
- Single-homed network (one upstream provider) does not need an AS number
  - Treated like part of upstream AS

Autonomous System Number (ASN)

- Two ranges
  - 0-65535 (original 16-bit range)
  - 65536-4294967295 (32-bit range - RFC4893)
- Usage:
  - 0 and 65535 (reserved)
  - 1-64495 (public Internet)
  - 64496-64511 (documentation - RFC5398)
  - 64512-65534 (private use only)
  - 23456 (represent 32-bit range in 16-bit world)
  - 65536-65551 (documentation - RFC5398)
  - 65552-4294967295 (public Internet)
- 32-bit range representation specified in RFC5396
  - Defines “asplain” (traditional format) as standard notation

Autonomous System Number (ASN)

ASNs are distributed by the Regional Internet Registries

They are also available from upstream ISPs who are members of one of the RIRs

The RIRs do not make distinctions between 2byte and 4byte ASNs when they allocate resources.

See www.iana.org/assignments/as-numbers

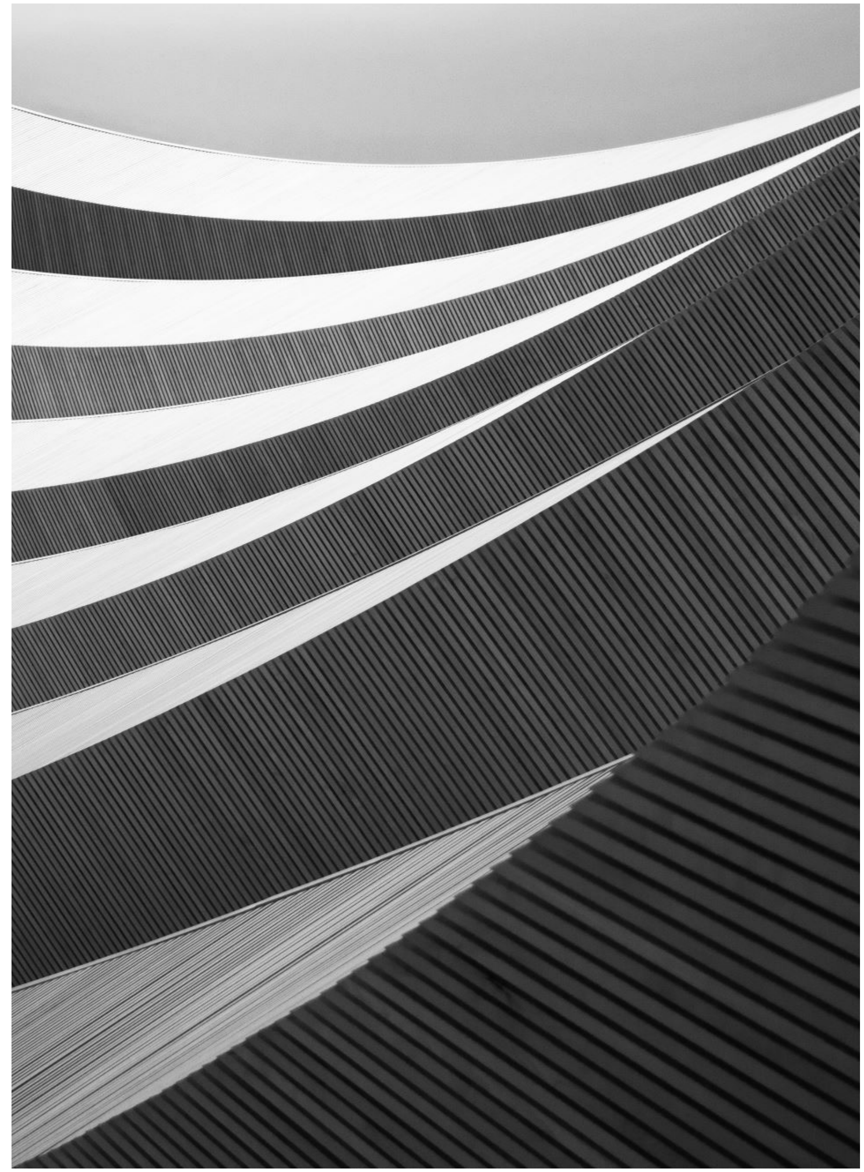
Using AS numbers

BGP can filter on AS numbers

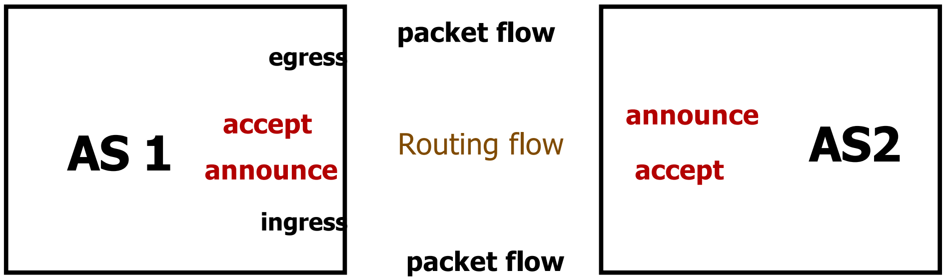
- Get all networks of the other ISP using one handle
- Include future new networks without having to change routing filters
  - AS number for new network will be same
- Can use AS numbers in filters with regular expressions

↓

BGP actually does routing computation on IP numbers



Routing flow and packet flow



- For networks in AS1 and AS2 to communicate:
  - AS1 must announce routes to AS2
  - AS2 must accept routes from AS1
  - AS2 must announce routes to AS1
  - AS1 must accept routes from AS2

Egress Traffic

- Packets exiting the network
- Based on:
  - Route availability (what others send you)
  - Route acceptance (what you accept from others)
  - Policy and tuning (what you do with routes from others)
  - Peering and transit agreements

Ingress Traffic

Packets entering your network

Ingress traffic depends on:

What information you send and to whom

Based on your addressing and ASes

Based on others' policy (what they accept from you and what they do with it

Difficult for you to influence what others choose to do, so load balancing ingress traffic is difficult

Types of Routes

- Static Routes
  - configured manually
- Connected Routes
  - created automatically when an interface is ‘up’
- Interior Routes
  - Routes within an AS
  - learned via IGP (e.g. OSPF)
- Exterior Routes
  - Routes exterior to AS
  - learned via EGP (e.g. BGP)

Basics of a BGP route

Seen from output of “show ip bgp”

Prefix and mask — what IP addresses are we talking about?

192.168.0.0/16 or 192.168.0.0/255.255.0.0

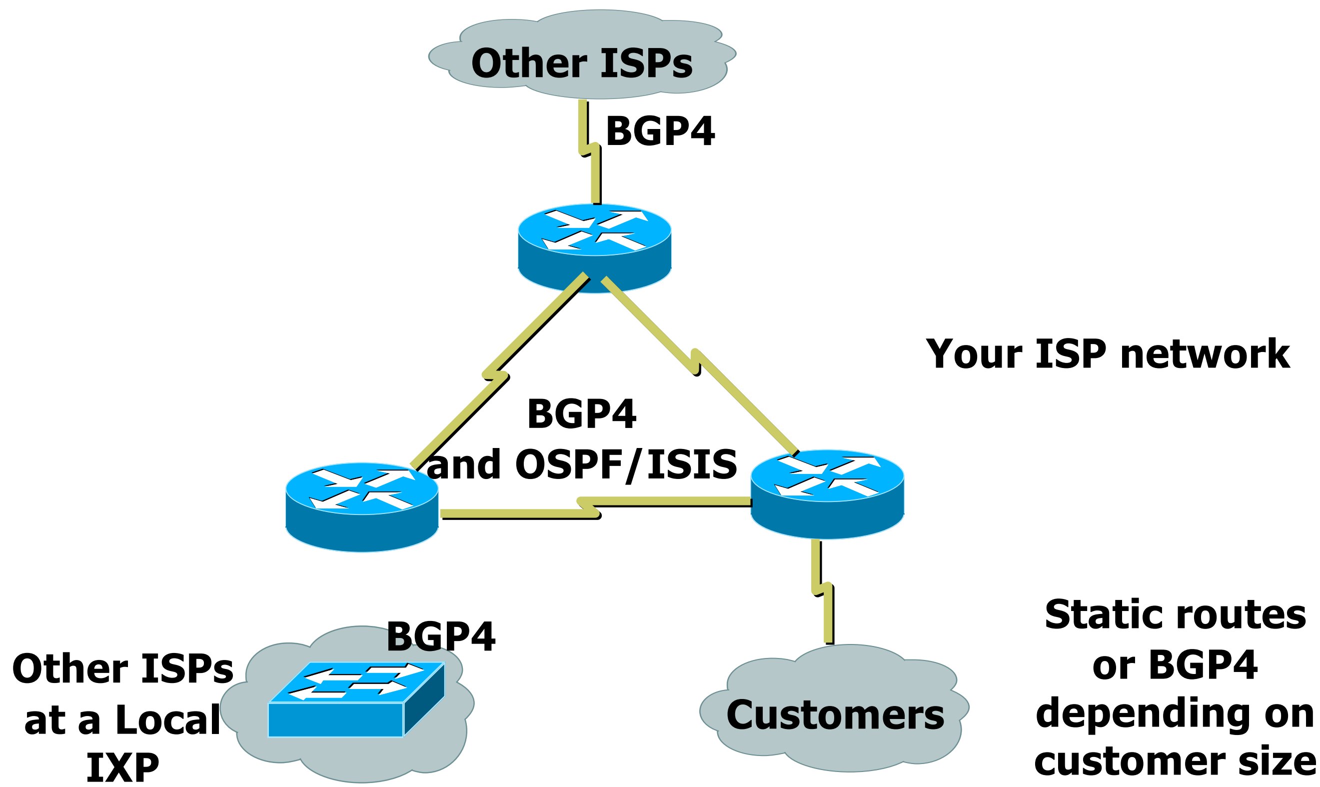
Origin – How did the route originally get into BGP?

“?” — incomplete, “e” — EGP, “i” — IGP

AS Path – what ASes did the route go through before it got to us?

“701 3561 1”

Hierarchy of Routing Protocols



BGP ATTRIBUTES and BEST PATH SELECTION

BGP Path Attributes: Why ?

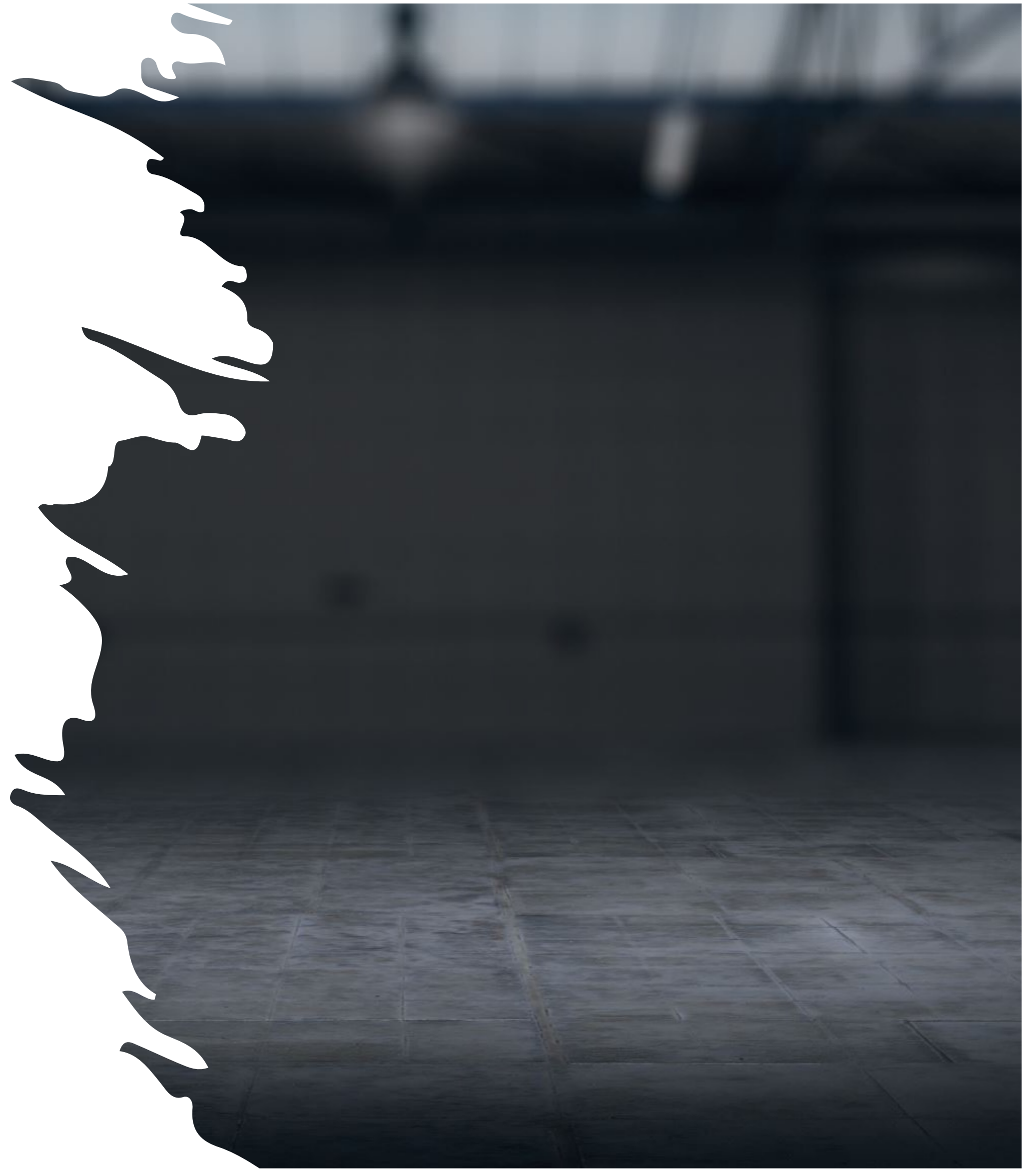
- Encoded as Type, Length & Value (TLV)
- Transitive/Non-Transitive attributes
- Some are mandatory
- Used in path selection
- To apply policy for steering traffic

BGP Attributes

- Used to convey information associated with NLRI
- AS path
- Next hop
- Local preference
- Multi-Exit Discriminator (MED)
- Community
- Origin
- Aggregator

Local Preference

- Not used by eBGP, mandatory for iBGP
- Default value of 100 on Cisco IOS
- Local to an AS
- Used to prefer one exit over another
- Path with highest local preference wins

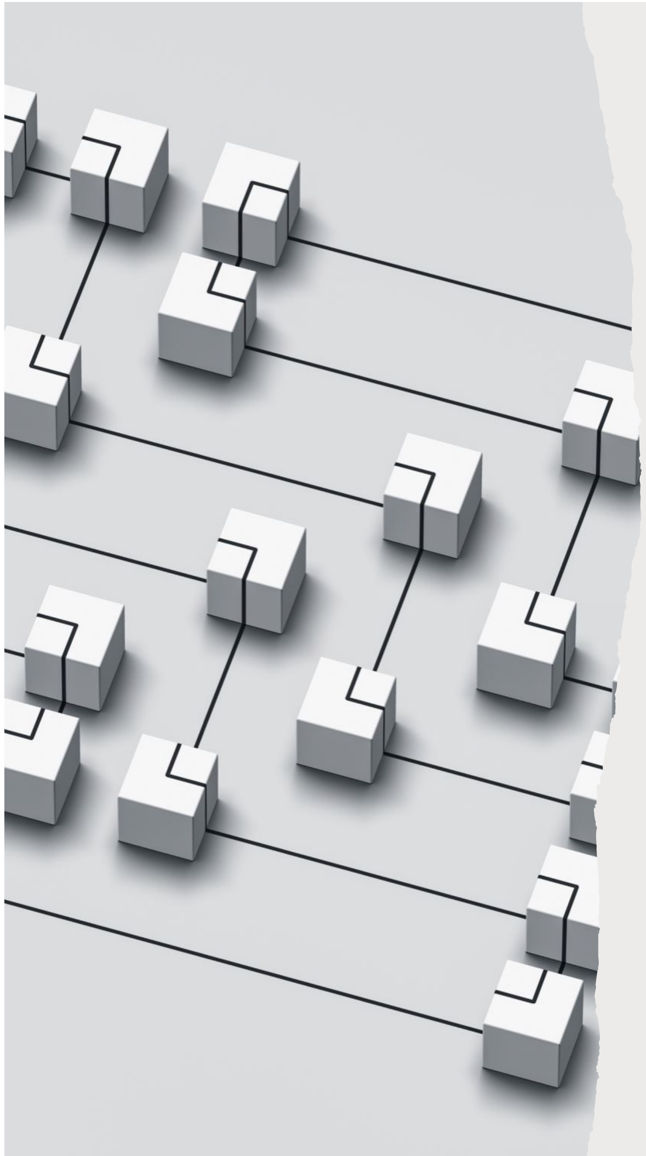


Multi-Exit Discriminator

- Non-transitive and optional
- Represented as a numerical value
- Range 0x0 – 0xffffffff (if unset 0 is assumed)
- **Used to convey relative preference of entry points to an AS**
- Comparable if the paths are from the same AS
- **bgp always-compare-med** allows comparisons of MEDs from different ASes
- Path with the lowest MED wins
- IGP metric can be conveyed as MED

Origin

- Conveys the origin of the prefix
- Historical attribute
- Three values:
  - IGP – from BGP network statement
  - E.g. – *network 35.0.0.0*
  - EGP – redistributed from EGP (not used today)
  - Incomplete – redistributed from another routing protocol
  - E.g. – *redistribute static*
- IGP < EGP < incomplete
  - Lowest origin code wins



Weight

- Not really an attribute (Cisco proprietary)
- Used when there is more than one route to same destination
- Local to the router on which it is assigned, and not propagated in routing updates
- Default is 32768 for paths that the router originates and zero for other paths
- Routes with a higher weight are preferred when there are multiple routes to the same destination

Communities

- Transitive, Non-mandatory
- Represented as a numeric value
- 0x0 – 0xffffffff
- Internet convention is ASN:<0-65535>
- Used to group destinations
- Each destination could be member of multiple communities
- Flexibility to scope a set of prefixes within or across AS for applying policy

Well-Known Communities

- Several well known communities
  - www.iana.org/assignments/bgp-well-known-communities
- no-export 65535:65281
- do not advertise to any eBGP peers
- no-advertise 65535:65282
- do not advertise to any BGP peer
- no-export-subconfed 65535:65283
- do not advertise outside local AS (only used with confederations)
- no-peer 65535:65284
- do not advertise to bi-lateral peers (RFC3765)

Administrative Distance

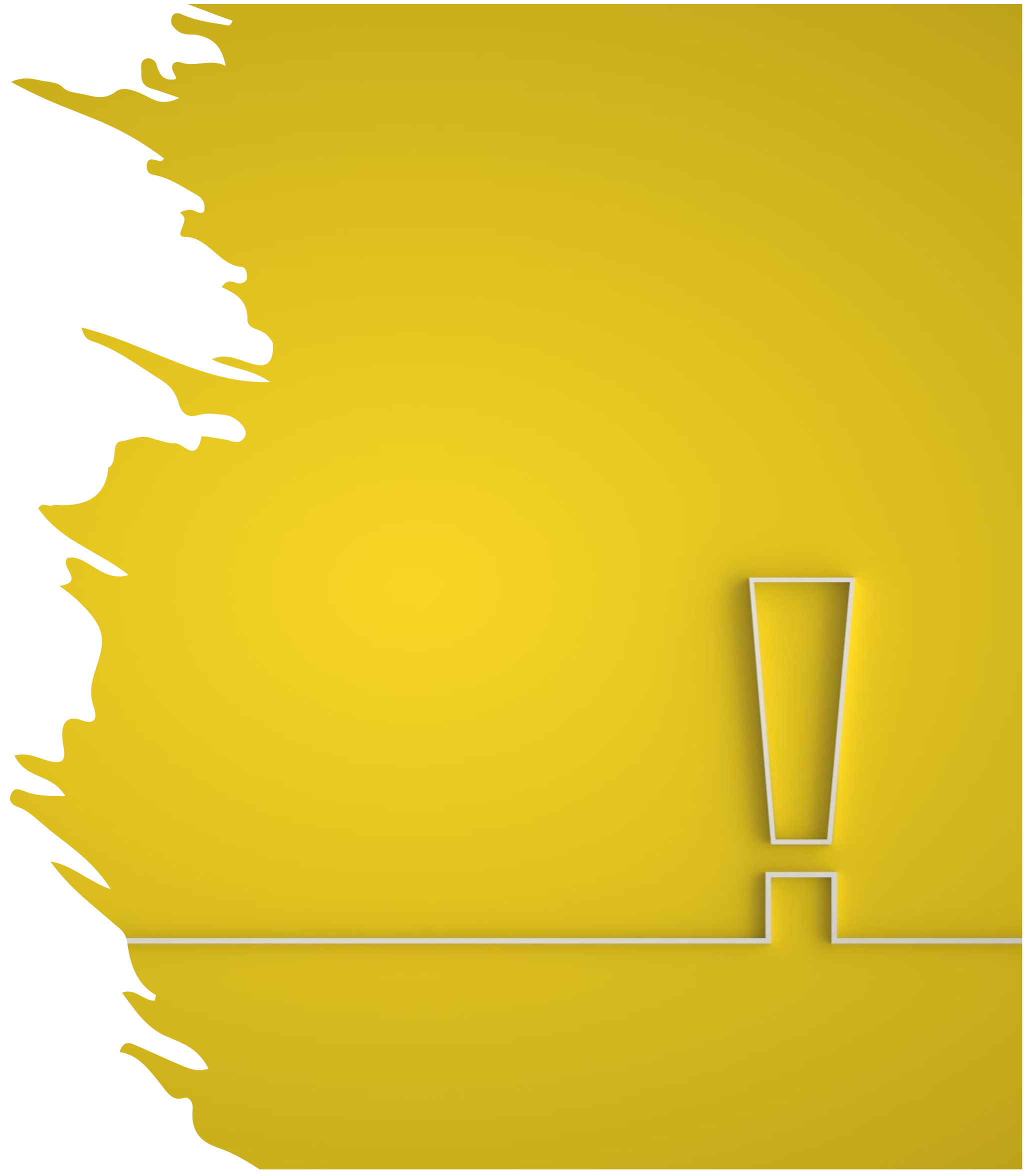
- Routes can be learned via more than one protocol
- Used to discriminate between them
- Route with lowest distance installed in forwarding table
- BGP defaults
- Local routes originated on router: 200
- iBGP routes: 200
- eBGP routes: 20
- Does not influence the BGP path selection algorithm but influences whether BGP learned routes enter the forwarding table

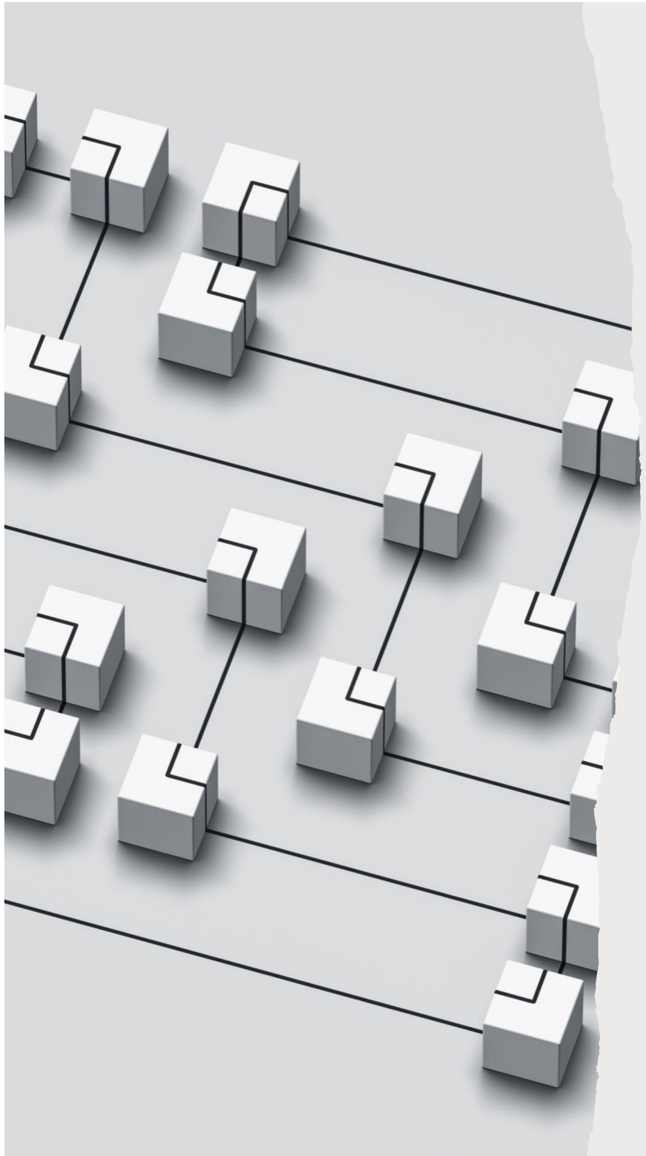
FYI: Default Administrative Distances

Route Source	Cisco	Juniper	Huawei	Brocade	Nokia/ALU
Connected Interface	**0**	**0**	**0**	**0**	**0**
Static Route	**1**	**5**	**60**	**1**	**1**
EIGRP Summary Route	5	N/A	?	N/A	N/A
External BGP	**20**	**170**	**255**	**20**	**170**
Internal EIGRP Route	90	N/A	?	N/A	N/A
IGRP	100	N/A	?	N/A	N/A
OSPF	**110**	**10**	**10**	**110**	**10**
IS-IS	**115**	**18**	**15**	**115**	**18**
RIP	120	100	100	120	100
EGP	140	N/A	N/A	N/A	N/A
External EIGRP	170	N/A	?	N/A	N/A
Internal BGP	**200**	**170**	**255**	**200**	**130**
Unknown	255	255	?	255	?

Synchronization

- In Cisco IOS, BGP does not advertise a route before all routers in the AS have learned it via an IGP
  - Default in IOS prior to 12.4; very unhelpful to most ISPs
- Disable synchronization if:
  - AS doesn’t pass traffic from one AS to another, or
  - All transit routers in AS run BGP, or
  - iBGP is used across backbone
- You should always use iBGP
  - so, always use “no synchronization”





BGP route selection (bestpath)

- Prefix must be in forwarding table
- Next-hop has to be accessible
- Next-hop must be in forwarding table
- Largest weight
- Largest local preference

BGP route selection (bestpath)

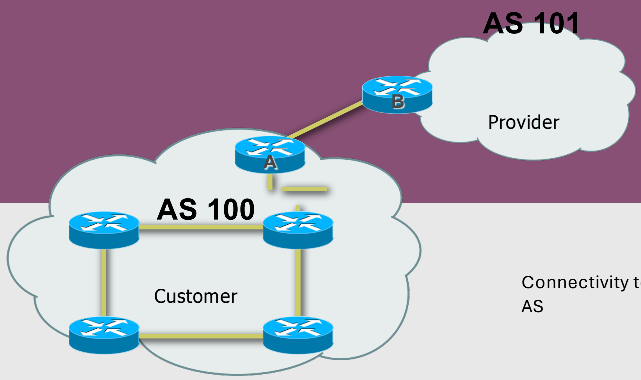
- Locally sourced
- Via redistribute or network statement
- Shortest AS path length
- Number of ASes in the AS-PATH attribute
- Lowest origin
- IGP < EGP < incomplete
- Lowest MED
- Compared from paths from the same AS

BGP route selection (bestpath)

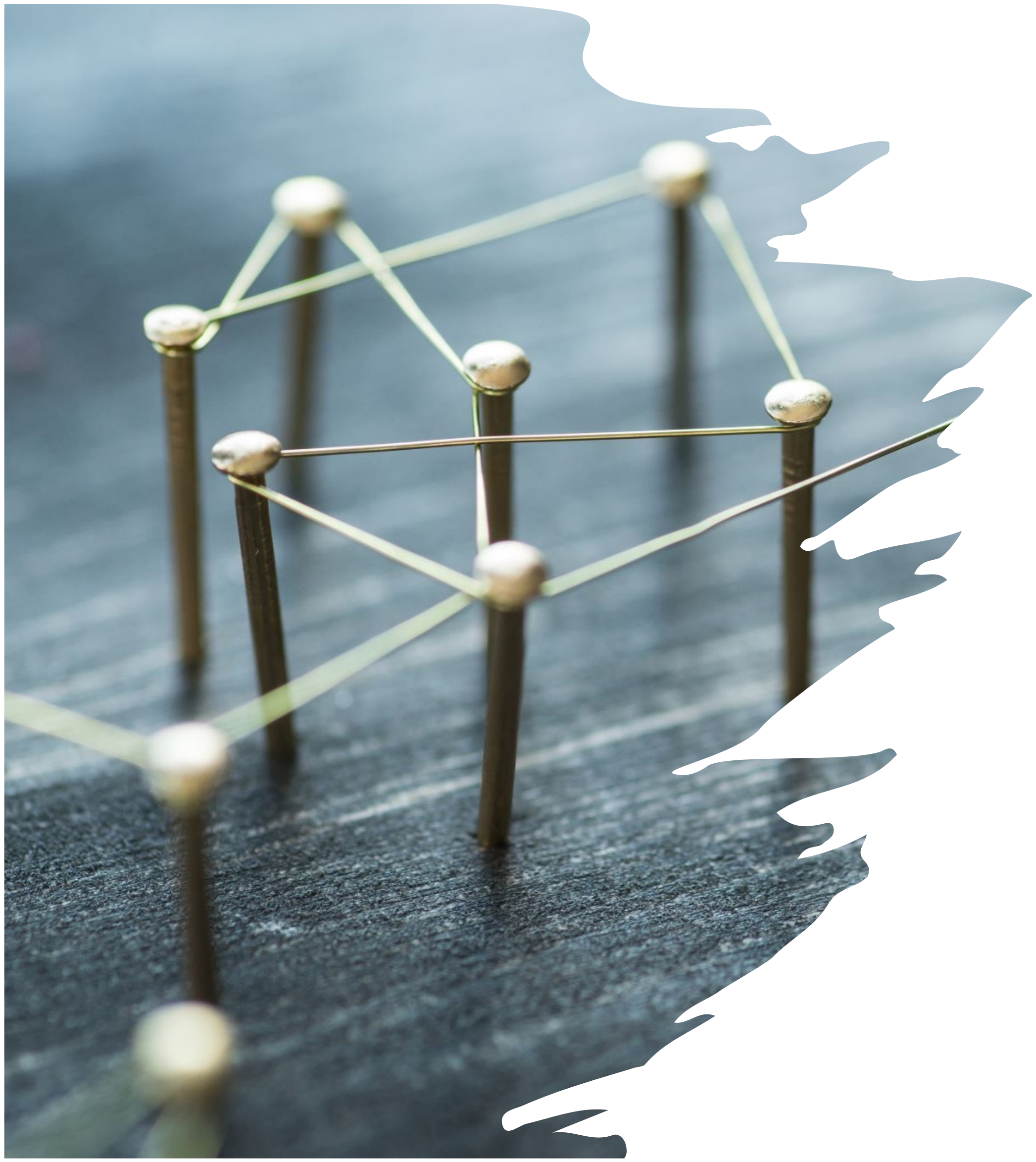
- External before internal
- Choose external path before internal
- Closest next-hop
- Lower IGP metric, nearest exit to router
- Lowest router ID
- Lowest IP address of neighbour

BGP and Network Design

Stub AS



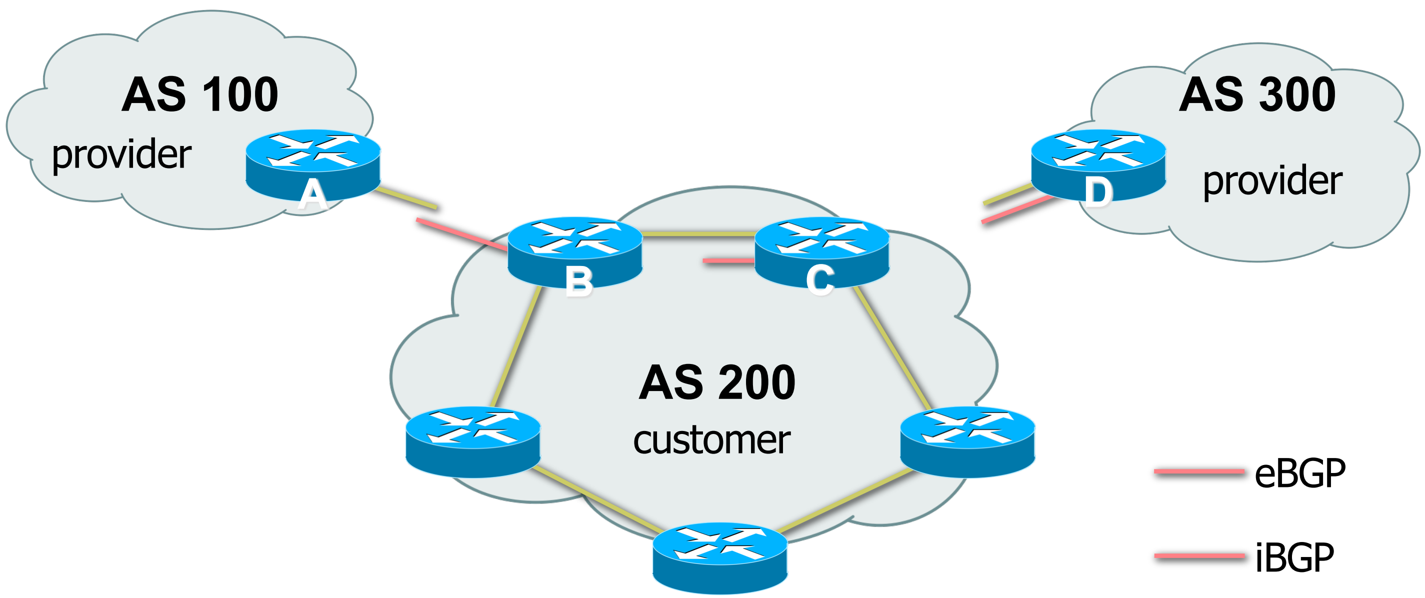
Connectivity to only one other AS



Stub AS

- Enterprise network, or small ISP
- Typically no need for BGP
- Point default towards the ISP
- ISP advertises the stub network to Internet
- Policy confined within ISP policy

Multi-homed AS



- More details on multihoming coming up…

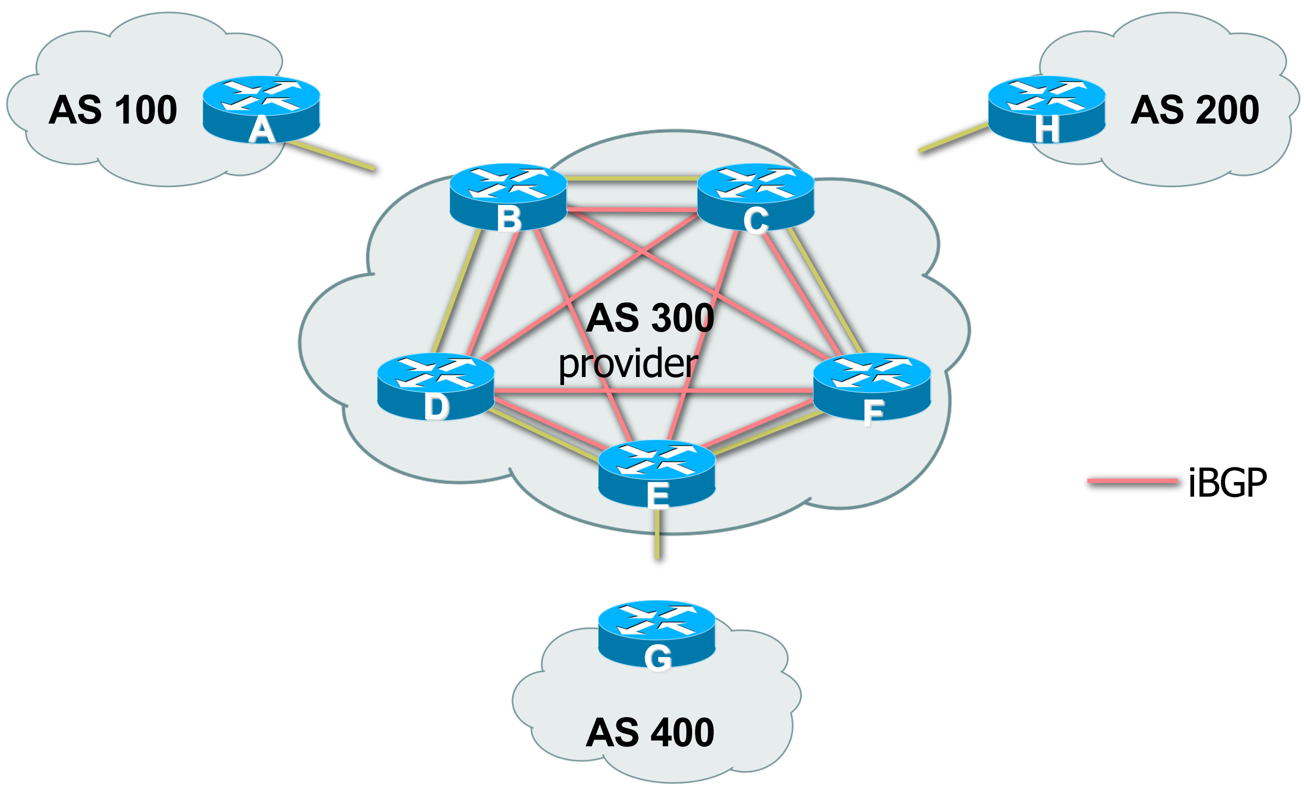
Multihomed AS

- Enterprise network or small ISP
- Only border routers speak BGP
  - And others on direct path between them
- iBGP only between border routers
- Rest of network either has:
  - exterior routes redistributed in a controlled fashion into IGP...
  - ...or use defaults (much preferred!)

Service Provider Network

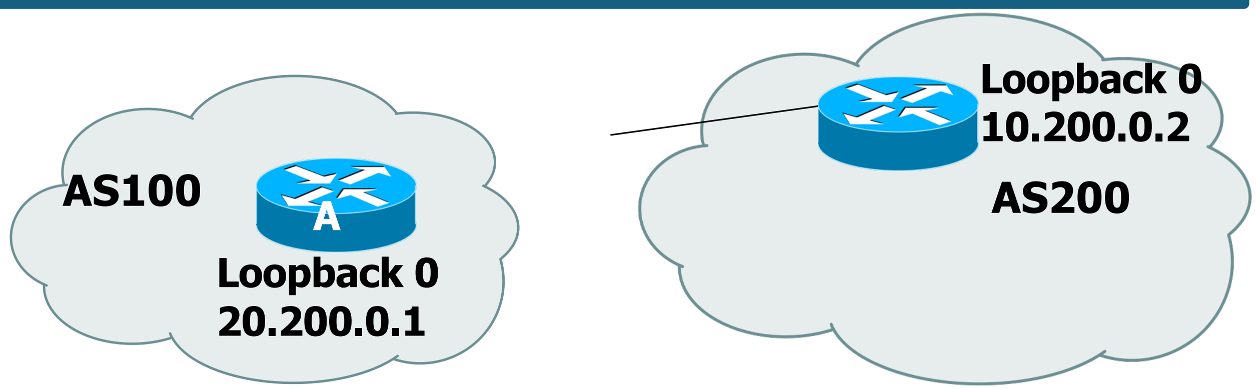
- iBGP used to carry exterior routes
  - No redistribution into IGP
- IGP used to track topology inside your network
- Full iBGP mesh required
  - Every router in ISP backbone should talk iBGP to every other router
  - This has scaling problems, and solutions (e.g. route reflectors)

Common Service Provider Network



Load-sharing – single path

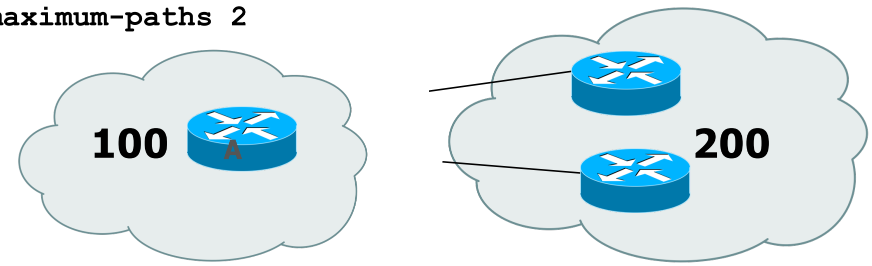
```
Router A:
interface loopback 0
ip address 20.200.0.1 255.255.255.255
!
router bgp 100
neighbor 10.200.0.2 remote-as 200
neighbor 10.200.0.2 update-source loopback0
neighbor 10.200.0.2 ebgp-multihop 2
!
ip route 10.200.0.2 255.255.255.255 <DMZ-link1>
ip route 10.200.0.2 255.255.255.255 <DMZ-link2>
```



Load-sharing – multiple paths from the same AS

- Router A:

```
router bgp 100
 neighbor 10.200.0.1 remote-as 200
 neighbor 10.300.0.1 remote-as 200
 maximum-paths 2
```

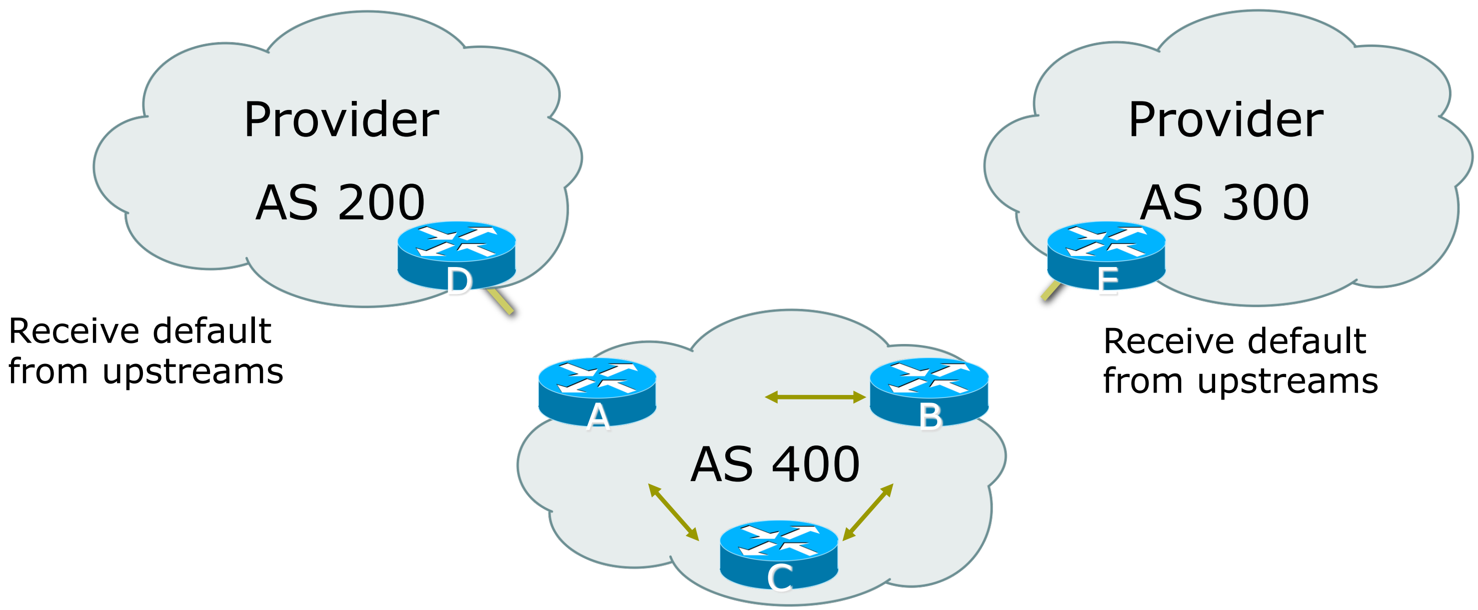


Note: A still only advertises one "best" path to iBGP peers

Redundancy – Multi-homing

- Reliable connection to Internet
- 3 common cases of multi-homing
  - default from all providers
  - customer + default from all providers
  - full routes from all providers
- Address Space
  - comes from upstream providers, or
  - allocated directly from registries

Default from all providers



Default from all providers

Low memory/CPU solution

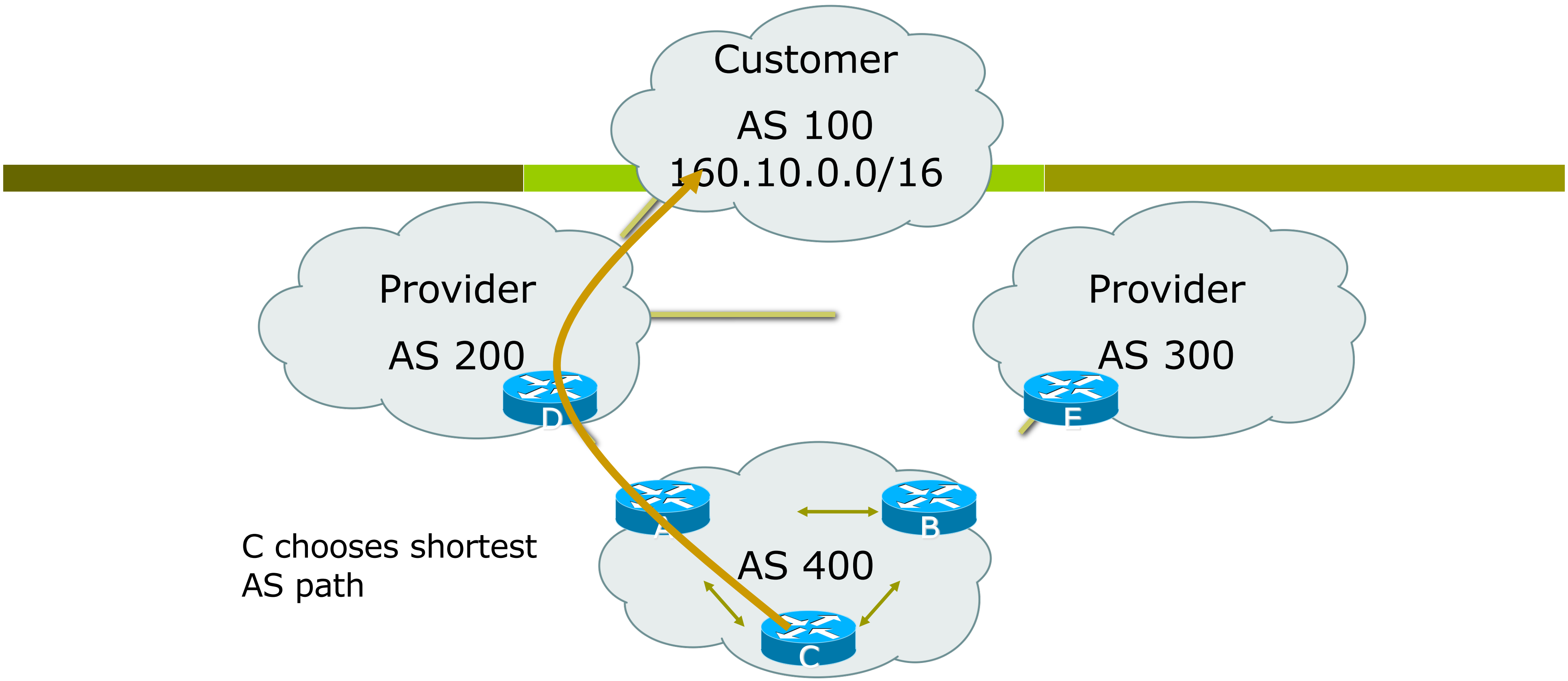
Provider sends BGP default

provider is selected based on IGP metric

Inbound traffic decided by providers' policy

Can influence using outbound policy, example: AS-path prepend

Customer routes from all providers



C chooses shortest AS path

Customer prefixes plus default from all providers

Medium memory and CPU solution

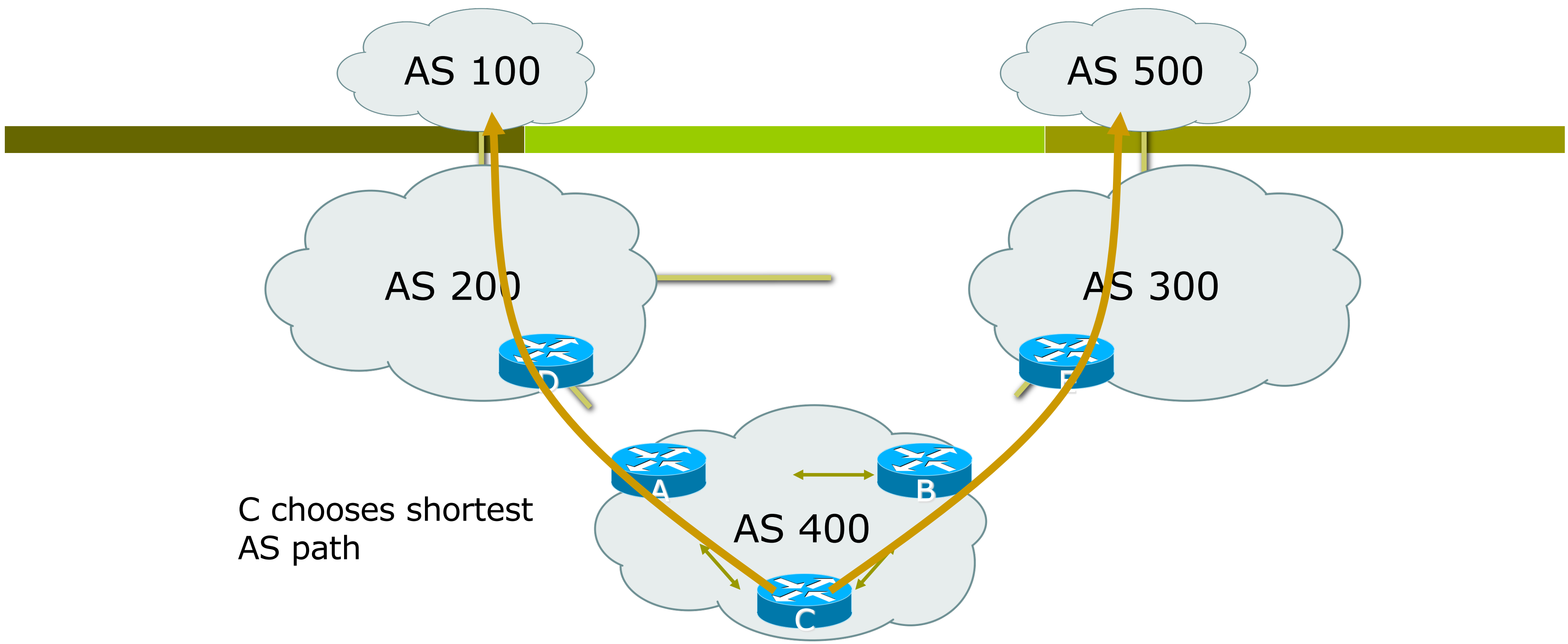
Granular routing for customer routes, default for the rest

Route directly to customers as those have specific policies

Inbound traffic decided by providers' policies

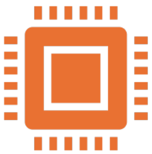
Can influence using outbound policy

Full routes from all providers



C chooses shortest AS path

Full routes from all providers



More memory/CPU

Fine grained routing control

Usually transit ASes take full routes

Usually pervasive BGP

Best Practices
IGP in Backbone

IGP connects your backbone together, not your clients' routes

- Clients' routes go into iBGP
- Hosting and service LANs go into iBGP
- Dial/Broadband/Wireless pools go into iBGP

IGP must converge quickly

- The fewer prefixes in the IGP the better

IGP should carry netmask information – OSPF, IS-IS, EIGRP

Best Practices
iBGP in Backbone

iBGP runs between all routers in backbone

Configuration essentials:

Runs between loopbacks

Next-hop-self

Send-community

Passwords

All non-infrastructure prefixes go here

Best Practices...
Connecting to a customer

Static routes

- You control directly
- No route flaps

Shared routing protocol or leaking

- Strongly discouraged
- You must filter your customers info
- Route flaps

BGP for multi-homed customers

- Private AS for those who multihome on to your backbone
- Public AS for the rest

Best Practices...
Connecting to other ISPs

Advertise only what you serve

Take back as little as you can

Take the shortest exit

Aggregate your routes!!

Consult RIPE-399 document for recommendations:

http://www.ripe.net/docs/ripe-399.html

FILTER! FILTER! FILTER!

BGP

Configuring BGP

Basic commands

Getting routes into BGP

Basic BGP commands

Configuration commands

router bgp <AS-number>

no auto-summary

no synchronization

neighbor <ip address> remote-as <as-number>

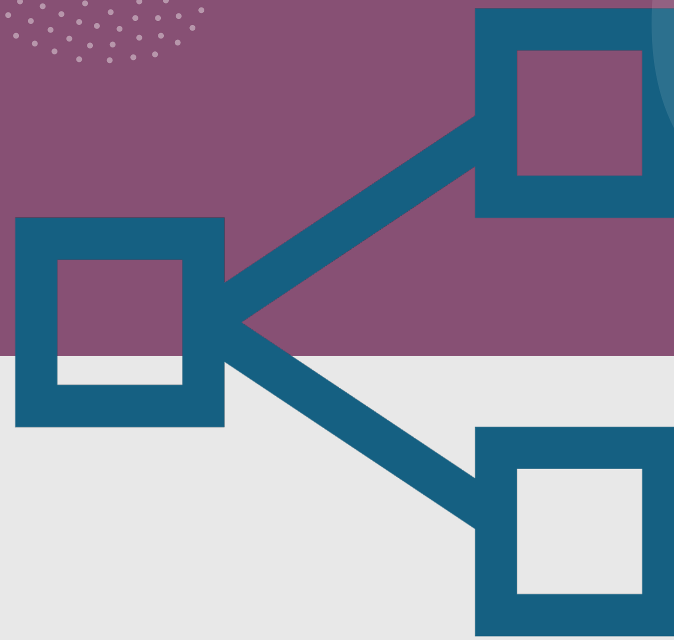
Show commands

show ip bgp summary

show ip bgp neighbors

show ip bgp neighbor <ip address>

Configuring BGP



- Router A:

```
router bgp 100
 neighbor 1.1.1.1 remote-as 101
```

- Router B:

```
router bgp 101
 neighbor 1.1.1.2 remote-as 100
```

Inserting prefixes into BGP

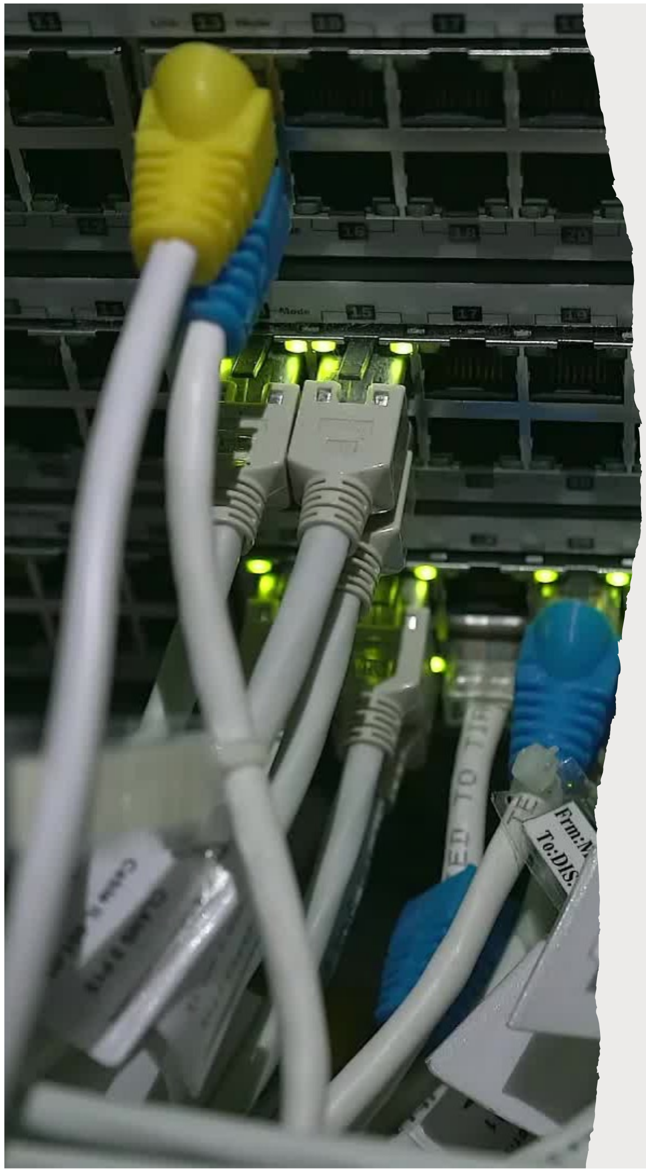
Two main ways to insert prefixes into BGP

network command

redistribute static

Both require the prefix to be in the routing table

You can't add a prefix to BGP unless the prefix is also in some other routing protocol (e.g. static route, IS-IS, OSPF)

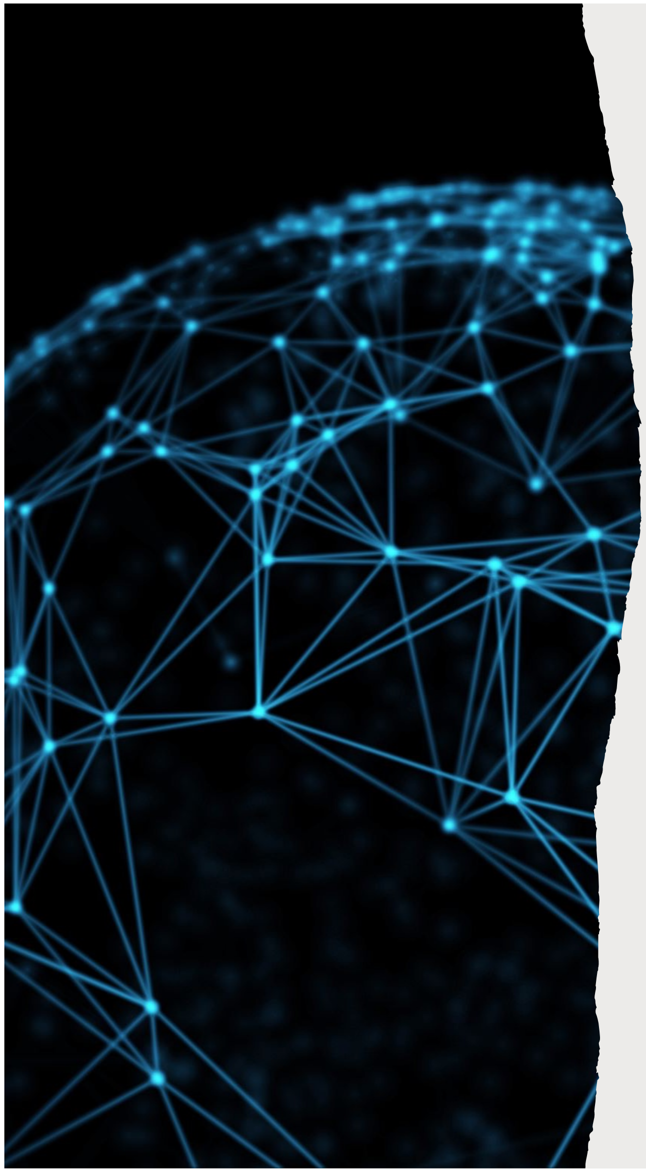


“network” command

- Configuration Example

```
router bgp 1
 network 105.32.4.0 mask 255.255.254.0
ip route 105.32.4.0 255.255.254.0 serial 0
```

- Matching route must exist in the routing table before network is announced!
- Prefix will have Origin code set to “IGP”

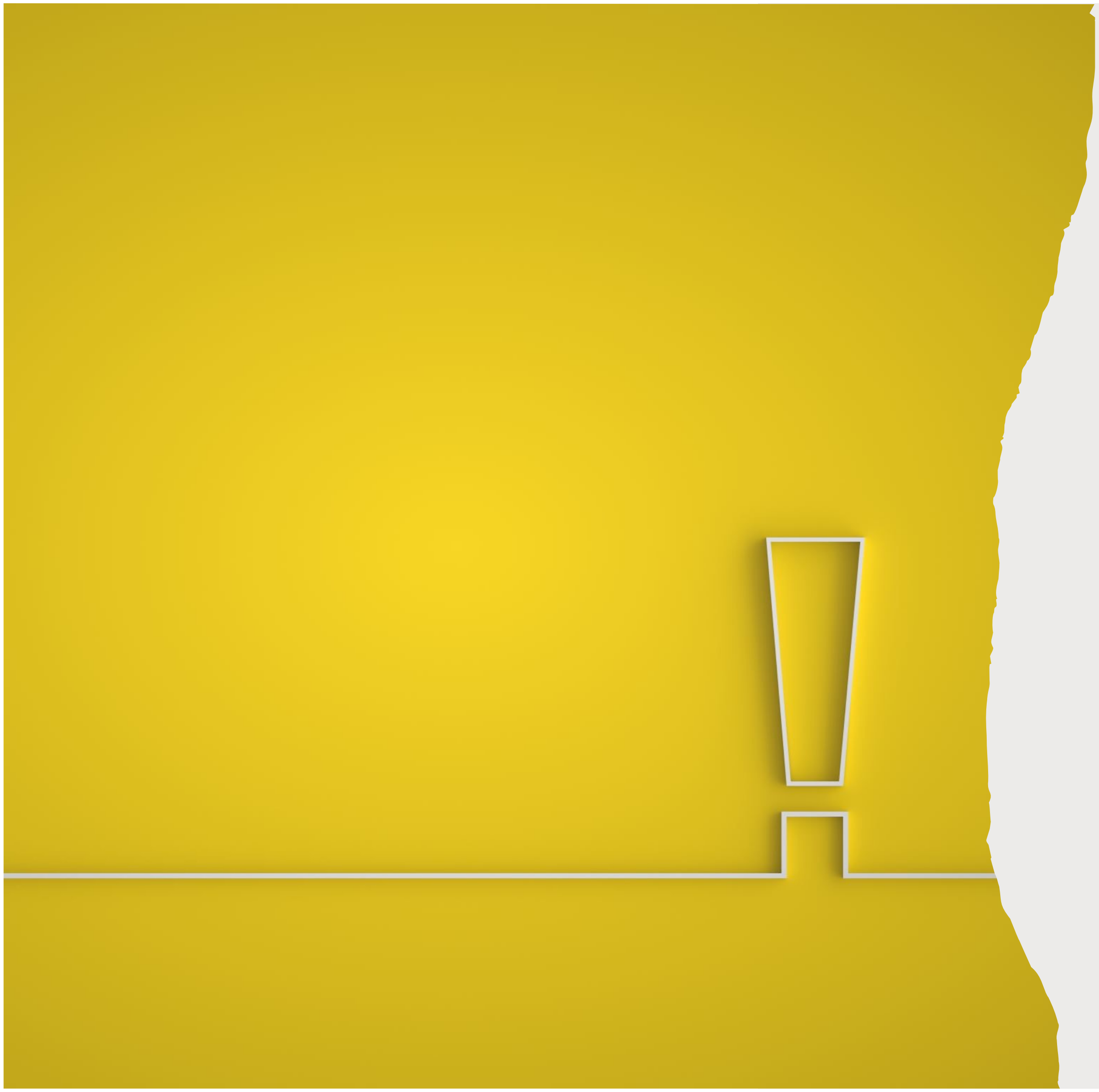


“redistribute static”

- Configuration Example:

```
router bgp 1
 redistribute static
ip route 105.32.4.0 255.255.254.0 serial0
```

- Static route must exist before redistribute command will work
- Forces origin to be “incomplete”
- Care required!
  - This will redistribute all static routes into BGP
  - **Redistributing without using a filter is dangerous; you could accidentally get many unwanted routes**



“redistribute static”

- Care required with redistribution
  - redistribute <routing-protocol> means everything in the <routing-protocol> will be transferred into the current routing protocol
  - will not scale if uncontrolled
  - best avoided if at all possible
  - redistribute normally used with “route-maps” and under tight administrative control
    - “route-map” is used to apply policies in BGP, so is a kind of filter

Aggregates and Null0

- Remember: matching route must exist in routing table before it will be announced by BGP

```
router bgp 1
 network 105.32.0.0 mask 255.255.0.0
ip route 105.32.0.0 255.255.0.0 null0 250
```

- Static route to null0 often used for aggregation
  - Packets will be sent here if there is no more specific match in the routing table
  - Distance of 250 ensures last resort
- Often used to nail up routes for stability
  - Can’t flap! ☺

Introducing IPv6…

Adding IPv6 to BGP…

- RFC4760
  - Defines Multi-protocol Extensions for BGP4
  - Enables BGP to carry routing information of protocols other than IPv4
    - e.g. MPLS, IPv6, Multicast etc
  - Exchange of multiprotocol NLRI must be negotiated at session startup
- RFC2545
  - Use of BGP Multiprotocol Extensions for IPv6 Inter-Domain Routing

RFC4760

- New optional and non-transitive BGP attributes:
  - MP_REACH_ NLRI (Attribute code: 14)
    - Carry the set of reachable destinations together with the next-hop information to be used for forwarding to these destinations (RFC4760)
  - MP_UNREACH_NLRI (Attribute code: 15)
    - Carry the set of unreachable destinations
- Attribute contains one or more Triples:
  - AFI Address Family Information
  - Next-Hop Information (must be of the same address family)
  - NLRI Network Layer Reachability Information

RFC2545

- IPv6 specific extensions
  - Scoped addresses: Next-hop contains a global IPv6 address and/or potentially a link-local address
  - NEXT_HOP and NLRI are expressed as IPv6 addresses and prefix
  - Address Family Information (AFI) = 2 (IPv6)
    - Sub-AFI = 1 (NLRI is used for unicast)
    - Sub-AFI = 2 (NLRI is used for multicast RPF check)
    - Sub-AFI = 3 (NLRI is used for both unicast and multicast RPF check)
    - Sub-AFI = 4 (label)

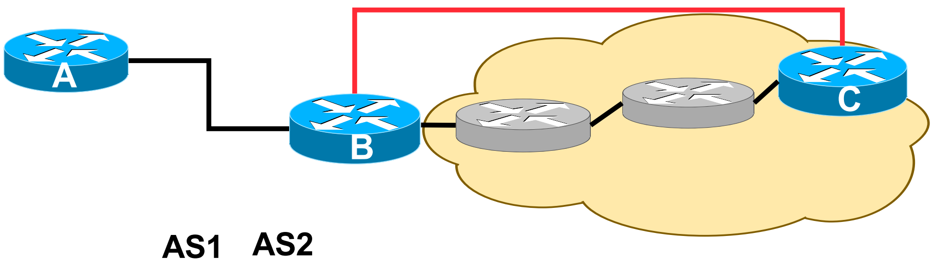
BGP Considerations

- Rules for constructing the NEXTHOP attribute:
  - When two peers share a common subnet, the NEXTHOP information is formed by a global address and a link local address
  - Redirects in IPv6 are restricted to the usage of link local addresses

Routing Information

- Independent operation
  - One RIB per protocol
    - e.g. IPv6 has its own BGP table
  - Distinct policies per protocol
- Peering sessions **can** be shared when the IPv4 and IPv6 topologies are congruent

BGP next-hop attribute



- Next-hop contains a global IPv6 address
  - (and potentially a link local address)
- Link local address is only set as a next-hop if the BGP peer shares the subnet with both routers (advertising and advertised)

More BGP considerations

- TCP Interaction
  - BGP runs on top of TCP
  - This connection could be set up either over IPv4 or IPv6
- Router ID
  - When no IPv4 is configured, an explicit bgp router-id needs to be configured
    - BGP identifier is a 32 bit integer currently generated from the router identifier – which is generated from an IPv4 address on the router
  - This is needed as a BGP identifier, this is used as a tie breaker, and is sent within the OPEN message

BGP Configuration

- IOS default is to assume that all configured peers are unicast IPv4 neighbours
  - If we want to support IPv6 too, this isn't useful
  - So we disable the default assumption

```
no bgp default ipv4-unicast
```

  - This means that we must explicitly state which address family the peer belongs to

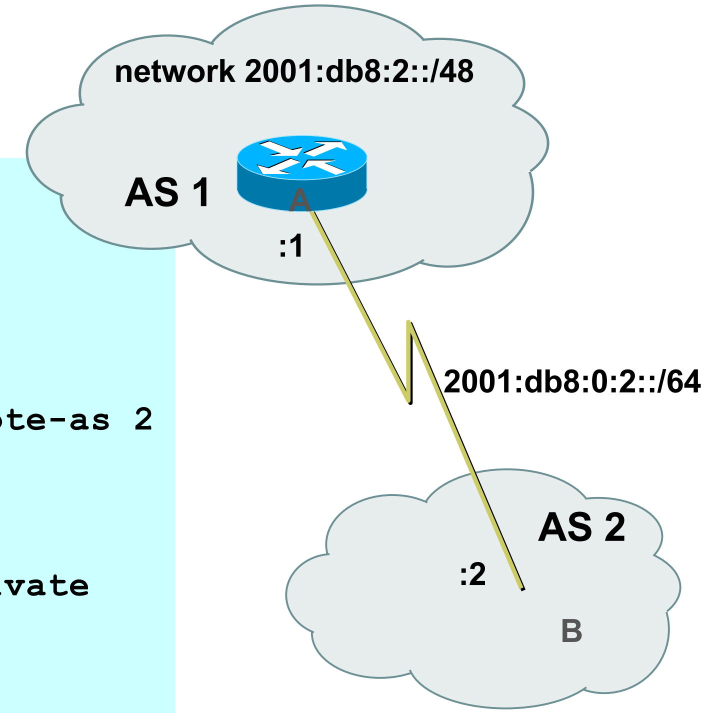
BGP Configuration

- Two options for configuring BGP peering
- Using link local addressing
  - ISP uses FE80:: addressing for BGP neighbours
  - **NOT RECOMMENDED**
    - There are plenty of IPv6 addresses
    - Unnecessary configuration complexity
- Using global unicast addresses
  - As with IPv4
  - **RECOMMENDED**

Regular BGP Peering

Router A

```
router bgp 1
 no bgp default ipv4 unicast
 neighbor 2001:db8:0:2::2 remote-as 2
!
 address-family ipv6
 neighbor 2001:db8:0:2::2 activate
 network 2001:db8:2::/48
!
```



Link Local Peering

Router A

```
interface fastethernet 0/0
 ipv6 address 2001:db8:0:1::1/64
!
router bgp 1
 no bgp default ipv4 unicast
 neighbor fe80::260:3eff:c043:1143 remote-as 2
!
address-family ipv6
 neighbor fe80::260:3eff:c043:1143 activate
 neighbor fe80::260:3eff:c043:1143 route-map next-hop in
!
route-map next-hop permit 5
 set ipv6 next-hop 2001:db8:0:1::1
!
```

DO NOT DO THIS

BGP Configuration IPv4 and IPv6

- When configuring the router, recommendation is:
  - Put **all** IPv6 configuration directly into IPv6 address family
  - Put **all** IPv4 configuration directly into IPv4 address family
- Router will sort generic from specific address family configuration when the configuration is saved to NVRAM or displayed on the console
- Example follows…
  - Notice how **`activate`** is required to indicate that the peering is activated for the particular address family

IPv4 and IPv6

- router bgp 10
- no bgp default ipv4-unicast
- neighbor 2001:db8:1:1019::1 remote-as 20
- neighbor 172.16.1.2 remote-as 30
- !
- address-family ipv4
- neighbor 172.16.1.2 activate
- neighbor 172.16.1.2 prefix-list ipv4-ebgp in
- neighbor 172.16.1.2 prefix-list v4out out
- network 172.16.0.0
- exit-address-family
- !
- address-family ipv6
- neighbor 2001:db8:1:1019::1 activate
- neighbor 2001:db8:1:1019::1 prefix-list ipv6-ebgp in
- neighbor 2001:db8:1:1019::1 prefix-list v6out out
- network 2001:db8::/32
- exit-address-family
- !

BGP Address Families Applied Configuration

```
router bgp 10
 no bgp default ipv4-unicast
!
address family ipv4
 neighbor 172.16.1.2 remote-as 30
 neighbor 172.16.1.2 prefix-list ipv4-ebgp in
 neighbor 172.16.1.2 prefix-list v4out out
 neighbor 172.16.1.2 activate
 network 172.16.0.0
!
 address-family ipv6
 neighbor 2001:db8:1:1019::1 remote-as 20
 neighbor 2001:db8:1 ... ipv6-ebgp in
 neighbor 2001:db8:1:1019::1 prefix-list v6out out
 neighbor 2001:db8:1:1019::1 activate
 network 2001:db8::/32
!
ip prefix-list ipv4-ebgp permit 0.0.0.0/0 le 32
ip prefix-list v4out permit 172.16.0.0/16
ipv6 prefix-list ipv6-ebgp permit ::/0 le 128
ipv6 prefix-list v6out permit 2001:db8::/32
```

Generic Configuration

Specific Configuration

BGP Address Families
End result – line order has changed

```
router bgp 10
 no bgp default ipv4-unicast
 neighbor 2001:db8:1:1019::1 remote-as 20
 neighbor 172.16.1.2 remote-as 30
!
 address-family ipv4
 neighbor 172.16.1.2 activate
 neighbor 172.16.1.2 prefix-list ipv4-ebgp in
 neighbor 172.16.1.2 prefix-list v4out out
 network 172.16.0.0
 exit-address-family
!
 address-family ipv6
 neighbor 2001:db8:1:10
 neighbor 2001:db8:1:1019::1 prefix-list ipv6-ebgp in
 neighbor 2001:db8:1:1019::1 prefix-list v6out out
 network 2001:db8::/32
 exit-address-family
!
ip prefix-list ipv4-ebgp permit 0.0.0.0/0 le 32
ip prefix-list v4out permit 172.16.0.0/16
ipv6 prefix-list ipv6-ebgp permit ::/0 le 128
ipv6 prefix-list v6out permit 2001:db8::/32
```

Generic Configuration

Specific Configuration

SPECIAL THANKS TO AFNOG FOR THE SLIDES.

QUESTIONS ?