Advanced Computer Networks

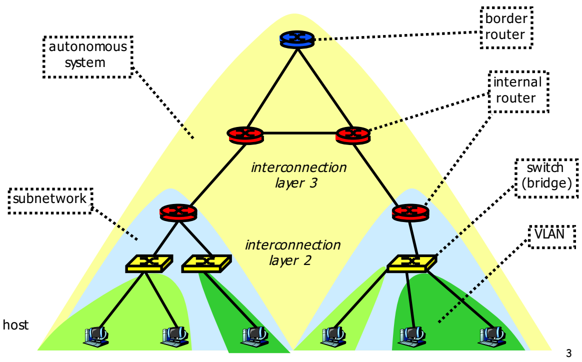
External Routing - BGP protocol

Prof. Andrzej Duda
duda@imag.fr

http://duda.imag.fr

Contents

Autonomous systems



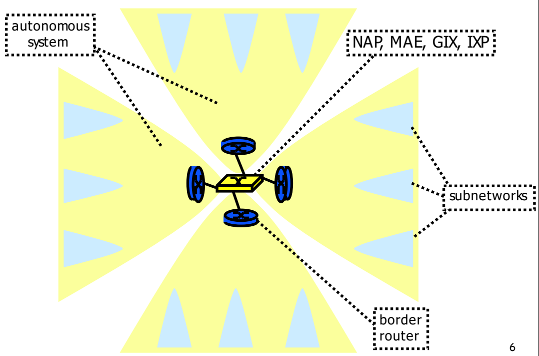
Autonomous Systems

- Routing domain under one single administration
  - one or more border routers
  - all subnetworks should be connected - run an interior gateway protocol (IGP like OSPF) to be able to forward packets within the AS
  - should learn about all other prefixes - use an exterior gateway protocol (EGP like BGP) to route packets to other AS
  - autonomy of management

AS numbers

- AS number
  - 16 bits, extended to 32 bits: x.y
  - 0.y – old 16 bits numbers, 1.y - reserved
  - public: 1 - 64511
  - private: 64512 - 65535
  - ASs that do not need a number are typically those with a default route to the rest of the world
- Examples
  - AS1942 - CICG-GRENOBLE, AS1717, AS2200 - Renater
  - AS559 - SWITCH Teleinformatics Services (EPFL)
  - AS5511 - OPENTRANSIT

Interconnection of AS



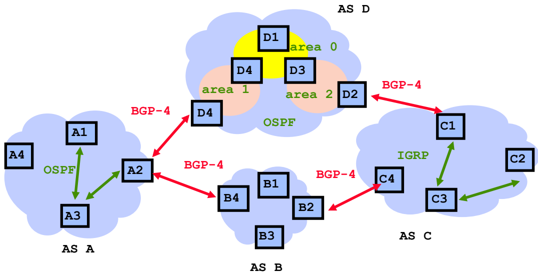
Interconnection of AS

- Border routers
  - interconnect AS
  - advertise routes to internal subnetworks
    - AS accepts the traffic
    - there is an internal route to the destination - AS is able to forward packets to the destination, otherwise - black hole
  - learn routes to external subnetworks
- Interconnection point
  - NAP (Network Access Point), MAE (Metropolitan Area Ethernet), CIX (Commercial Internet eXchange), GIX (Global Internet eXchange), IXP, SFINX, LINX
  - exchange of traffic - peering contract between ASs
- High-speed local area network connecting border routers of ASs

7

Example interconnection

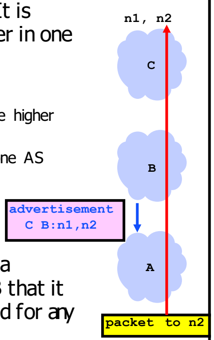
- AS can be transit (B and D), stub (A) or multihomed (C). Only non stub AS needs a number.



8

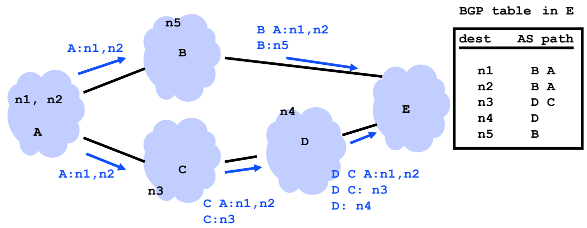
What does BGP do?

- BGP is a routing protocol between AS. It is used to establish routes from one router in one AS to any network prefix in the world
- There are two levels in BGP:
  - Inter-domain: one AS is a virtual node in the higher layer
  - Intra-domain: distribution of routes inside one AS
- The method of routing is
  - Path vector
  - With policy
- A route advertisement from B to A for a destination prefix is an agreement by B that it will forward packets sent via A destined for any destination in the prefix.



9

Path Vector routing

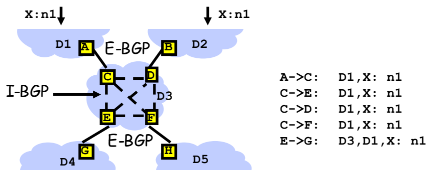


- AS maintains a table of best paths known so far
- Table updated using local rules
- Suitable when
  - no global meaning for costs can be assumed (heterogeneous environments)
  - global topology is fairly stable

10

Border Routers, E-BGP and I-BGP

- E-BGP: BGP runs on *border routers = "BGP speakers"* belonging to one AS only
  - two border routers per boundary (OSPF - one per area boundary)
- I-BGP: BGP speakers talks to each other inside the AS using "Internal-BGP"
  - full mesh called the "BGP mesh"
  - I-BGP is the same as E-BGP except for one rule: routes learned from a neighbour in the mesh are not repeated inside the mesh




```
A->C:  D1,X: n1
C->E:  D1,X: n1
C->D:  D1,X: n1
C->F:  D1,X: n1
E->G:  D3,D1,X: n1
```

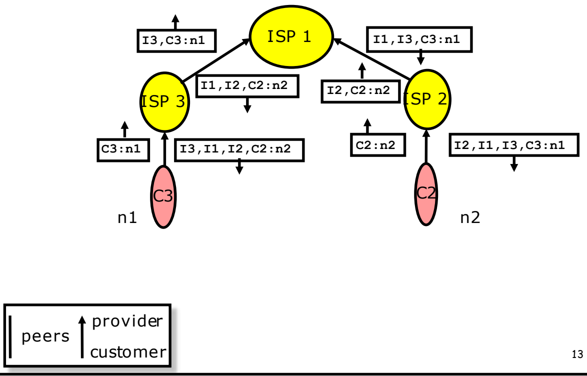
11

Policy Routing

- Mainly 3 types of relations depending on money flows
  - **customer**: EPFL is customer of Switch. EPFL pays Switch
  - **provider**: Switch is provider for EPFL; Switch is paid by EPFL
  - **peer**: EPFL and CERN are peers: costs of interconnection is shared
- Type of relation is negotiated in bilateral agreements there is no architecture rule, just business

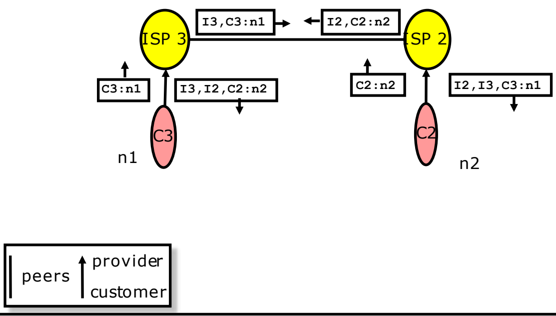
12

Customer-provider



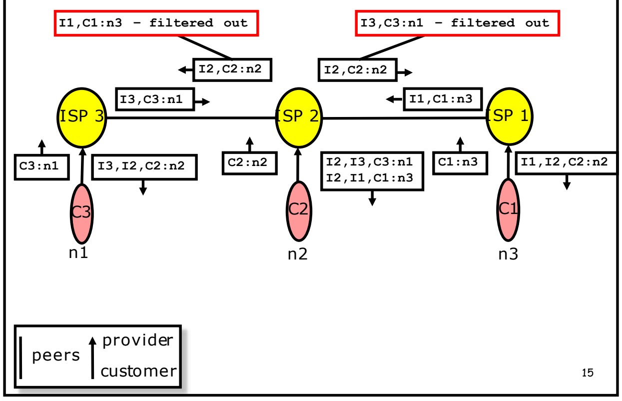
13

Peers



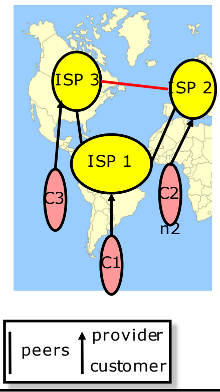
14

Peers



15

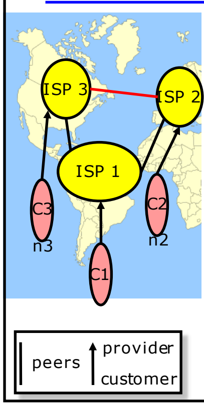
Goal of Policy Routing



- Example:
  - ISP3 - ISP2 is transatlantic link, cost shared between ISP2 and ISP 3
  - ISP 3 - ISP 1 is a local, inexpensive link
  - Ci is customer of ISPi, ISPs are peers
- It is advantageous for ISP3 to send traffic to n2 via ISP1
- ISP1 does not agree to carry traffic from C3 to C2
  - ISP1 offers a "transit service" to C1 and a "non-transit" service to ISP 2 and ISP3
- The goal of "policy routing" is to support this and other similar requirements

16

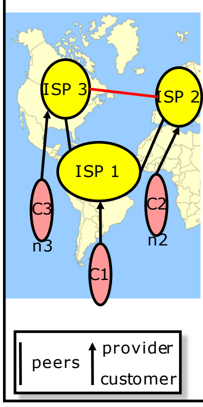
How does Policy Routing Work ?



- Implemented by rules followed by BGP speakers
  - refuse to import or announce some routes
  - modify the attributes that control which route is preferred (see later)
- Example
  - ISP 1 announces to ISP 3 all networks of C1
  - ISP 1 announces to C1 all routes it has learnt from ISP3 and ISP2
  - ISP2 announces "ISP2 n2" to ISP3 and ISP1; assume that ISP1 announces "ISP1 ISP2 n2" to ISP3.
  - ISP 3 has two routes to n2: "ISP2 n2" and "ISP1 ISP2 n2"; assume that ISP3 prefers "ISP1 ISP2 n2"
  - packets from n3 to n2 are routed via ISP1 – undesired
  - solution: ISP 1 announces to ISP3 only routes to ISP1's customers (not "ISP1 ISP2 n2")

17

Typical Policy Routing Rules



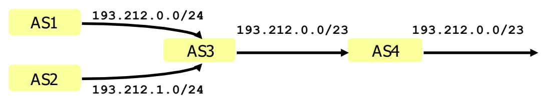
- Provider (ISP1) to customer (C1)
  - announce all routes learnt from other ISPs
  - import only routes that belong to C1 example: import from IMAG only one route 129.88/16
- Customer (C1) to Provider (ISP1)
  - announce all routes that belong to C1
  - import all routes
- Peers (ISP1 to ISP3)
  - announce only routes to all customers of ISP1
  - import only routes to ISP3's customer
  - these routes are defined as part of peering agreement
- The rules are defined by every AS and implemented in all BGP speakers in one AS

18

Prefix Aggregation

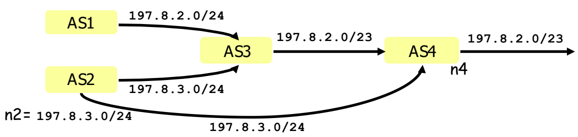
- AS that does not have a default route (i.e. all transit ISPs) must know all routes in the world (> 280 000 prefixes)
  - in IP routing tables unless default routes are used
  - in BGP announcements
- Aggregation is a way to reduce the number of routes

Aggregation Example 1



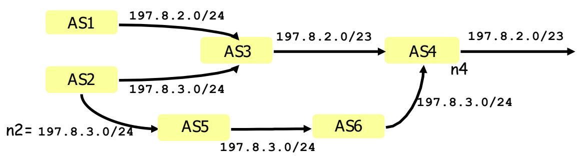
- Assume AS3 aggregates the routes received from AS1 and AS2
  - `AS1: 193.212.0.0/24      AS_PATH: 1`
  - `AS2: 193.212.1.0/24      AS_PATH: 2`
  - `AS3: 193.212.0.0/23      AS_PATH: 3 {1 2}`
  - `AS4: 193.212.0.0/23      AS_PATH: 4 3 {1 2}`

Aggregation Example 2



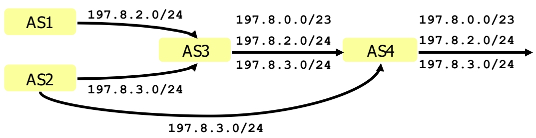
- AS4 receives
  - `197.8.2.0/23      AS_PATH: 3 {1 2}`
  - `197.8.3.0/24      AS_PATH: 2`
- What happens to packets from n4 to n2?
  - if AS4 puts two entries: `197.8.2.0/23, 197.8.3.0/24`
  - if AS4 puts one entry: `197.8.2.0/23`

Aggregation Example 3



- AS4 receives
  - `197.8.2.0/23      AS_PATH: 3 {1 2}`
  - `197.8.3.0/24      AS_PATH: 6 5 2`
- What happens to packets from n4 to n2?
  - if both routes are used: `197.8.2.0/23, 197.8.3.0/24`
  - if the shortest AS path is used: `197.8.2.0/23`

Example Without Aggregation



- AS3 has `197.8.0.0/23`
- If AS3 does not aggregate, what are the routes announced by AS 4?
  - `197.8.0.0/23      AS_PATH: 4 3`
  - `197.8.2.0/24      AS_PATH: 4 3 1`
  - `197.8.3.0/24      AS_PATH: 4 2`
- There is no benefit since all routes go via AS 4 anyhow. AS4 should aggregate to `197.8.0.0/22`.

Morality

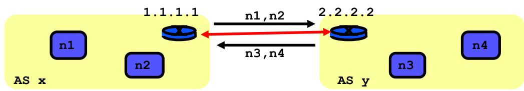
- Aggregation should be performed whenever possible
  - when all aggregated prefixes have the same path (example 1)
  - when all aggregated prefixes have the same path before the aggregation point (examples 2 to 4)
- An AS can decide to
  - Aggregate several routes when exporting them
  - But still maintain different routing entries inside its domain (example 2)

BGP (Border Gateway Protocol)

- BGP-4, RFC 1771
- AS border router - BGP speaker
  - peer-to peer relation with another AS border router
  - connected communication
    - on top of a TCP connection, port 179 (vs. datagram (RIP, OSPF))
  - external connections (E-BGP)
    - with border routers of different AS
  - internal connections (I-BGP)
    - with border routers of the same AS
  - BGP only transmits modifications

25

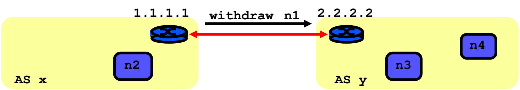
BGP principles



- Establish BGP session
- Update
  - list of destinations reachable via each router
  - path attributes such as degree of preference for a particular route

26

BGP principles



- `n1` no longer reachable
- Incremental update
  - withdraw `n1`

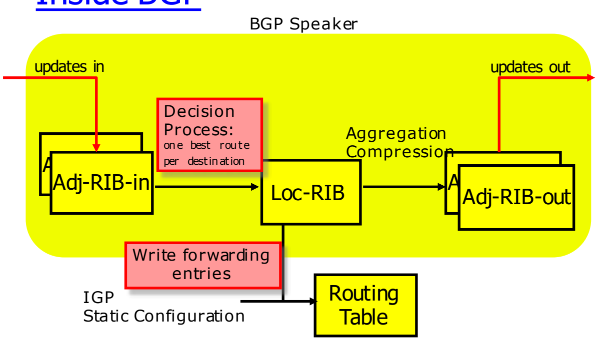
27

Operation of a BGP speaker

- BGP speaker
  - stores received routes in **Adj-RIB-in**
    - one per BGP peer (internal or external)
  - applies **decision process** and stores results in **Loc-RIB** (global to BGP speaker)
    - decide which routes to accept, how to rank them (set LOCAL-PREF), which routes to export and with which attributes
  - dispatches results per outgoing interface into **Adj-RIB-out** (one per BGP peer), after aggregation and information reduction
  - maintains adjacency to peers: open, keep-alive
  - sends updates when Adj-RIB-out changes
  - **write forwarding entries** in its routing table; redistributes routes learnt from E-BGP from Loc-RIB into IGP and vice-versa, unless other mechanisms are used (see examples)

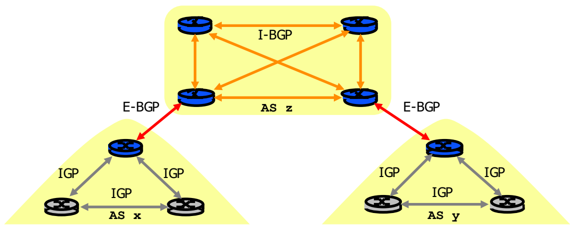
28

Inside BGP



29

E-BGB and I-BGP



- Border routers of different AS exchange route information using External BGP (E-BGP)
  - peer border routers should be on the same subnetwork
- Border routers of AS exchange route information using Internal BGP (I-BGP)

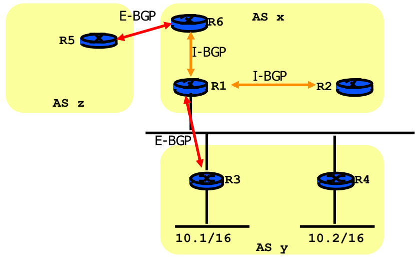
30

Routes have attributes

- **Route** - unit of information; contains:
  - destination (subnetwork prefix)
  - path to the destination (AS-PATH)
  - attributes
    - Well-known Mandatory
      - – ORIGIN (route learnt from IGP, BGP or static)
      - – AS-PATH
      - – NEXT-HOP (see later)
    - Well-known Discretionary
      - – LOCAL-PREF (see later)
      - – ATOMIC-AGGREGATE (= route cannot be dis-aggregated)
    - Optional Transitive
      - – MULTI-EXIT-DISC (MED) (see later)
      - – AGGREGATOR (who aggregated this route)
    - Optional Nontransitive
      - – WEIGHT (see later)

31

NEXT_HOP



- R3 advertises `10.2/16` to R1, NEXT_HOP = R4 IP address
- R6 advertises `10.1/16` to R5, NEXT_HOP = R6 IP address

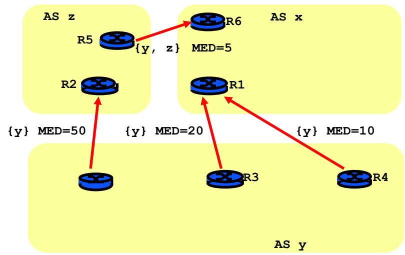
32

Preference attributes

- When multiple routes exist, choose one route to put into the BGP routing table
- Preference information
  - passed to other ASs - MED
  - local to an AS - LOCAL_PREF
  - local to a BGP router - WEIGHT

33

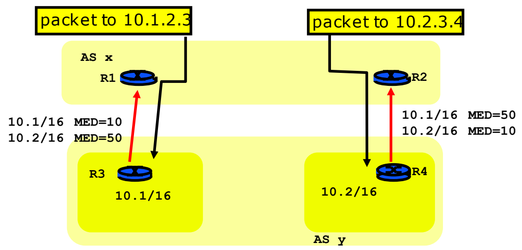
MULTI-EXIT-DISC (MED)



- Preference for a prefix list when there are several exit routers from an AS
  - AS y advertises its prefixes with MED 10, 20, 50
  - AS x will accept the prefix with the smallest MED

34

MULTI-EXIT-DISC (MED)



- One AS connected to another over several links
  - ex: multinational company connected to worldwide ISP
  - AS y advertises its prefixes with different MEDs (low = preferred)
  - If AS x accepts to use MEDs put by AS y: traffic goes on preferred link

35

MED Example

- Q1: by which mechanisms will R1 and R2 make sure that packets to ASy use the preferred links?
  - R1 and R2 exchange their routes to AS y via I-BGP
  - R1 has 2 routes to 10.1/16, one of them learnt over E-BGP; prefers route via R1; injects it into IGP
  - R1 has 2 routes to 10.2/16, one of them learnt over E-BGP; prefers route via R2; does not inject a route to 10.2/16 into IGP
- Q2: router R3 crashes; can 10.1/16 still be reached ? explain the sequence of actions.
  - R1 clears routes to AS y learnt from R3 (keep-alive mechanism)
  - R2 is informed of the route suppression by I-BGP
  - R2 has now only 1 route to 10.1/16 and 1 route to 10.2/16;. keeps both routes in its local RIB and injects them into IGP since both were learnt via E-BGP
  - traffic to 10.1/16 now goes to R2

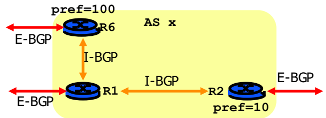
36

MED Question

- Q1: Assume now AS x and AS y are peers (ex: both are ISPs). Explain why AS x is not interested in taking MED into account.
  A: AS x is interested in sending traffic to AS y to the nearest exit, avoiding transit inside AS x as much as possible. Thus AS x will choose the nearest route to AS y and will ignore MEDs
- Q2: By which mechanisms can AS x pick the nearest route to AS y?
  A: it depends on the IGP. With OSPF: all routes to AS y are injected into OSPF by means of type 5 LSAs. These LSAs say: send to router R3 or R4. Every OSPF router inside AS x knows the cost (determined by OSPF weights) of the path from self to R3 and R4. Packets to 10.1/16 and 10.2/16 are routed to the nearest among R3 and R4 (nearest = lowest OSPF cost).

37

LOCAL-PREF



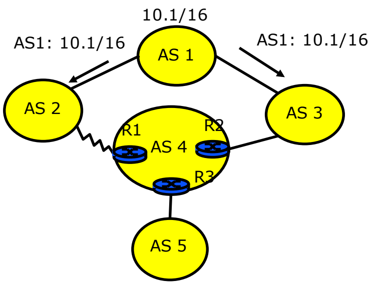
- Used inside an AS to select the best *AS path*
- Assigned by border router when receiving route over E-BGP
  - Propagated without change over I-BGP
- Example
  - R6 associates pref=100, R2 pref=10
  - R1 chooses the largest preference

```
bgp default local-preference pref-value
```

38

LOCAL-PREF Example

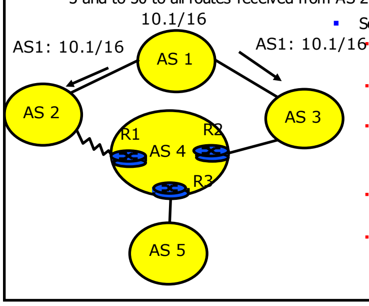
- Q1: The link AS2-AS4 is expensive. How should AS 4 set local-prefs on routes received from AS 3 and AS 2 in order to route traffic preferably through AS 3 ?
- Q2:Explain the sequence of events for R1, R2 and R3.



39

LOCAL-PREF Example

- Q1: The link AS2-AS4 is expensive. How should AS 4 set local-prefs on routes received from AS 3 and AS 2 in order to route traffic preferably through AS 3 ?
  A: for example: set LOCAL-PREF to 100 to all routes received from AS 3 and to 50 to all routes received from AS 2



- Sequence of events
  - R1 receives the route AS2 AS1 10.1/16 over E-BGP; sets LOCAL-PREF to 50
  - R2 receives the route AS3 AS1 10.1/16 over E-BGP; sets LOCAL-PREF to 100
  - R3 receives AS2 AS1 10.1/16, LOCAL-PREF=50 from R1 over I-BGP and
    AS3 AS1 10.1/16, LOCAL-PREF=100 from R1 over I-BGP
  - R3 selects AS3 AS1 10.1/16, LOCAL-PREF=100 and installs it into local-RIB
  - R3 announces onlyAS4, AS3 AS1 10.1/16 to AS 5

40

LOCAL-PREF Question

- Q: Compare MED to LOCAL-PREF
  A:
  - MED is used between ASs (i.e. over E-BGP); LOCAL-PREF is used inside one AS (over I-BGP)
  - MED is used to tell one provider AS which *entry link* to prefer; LOCAL-PREF is used to tell the rest of the world which *AS path* we want to use, by not announcing the other ones.

41

WEIGHT

- Associate a weight with a neighbor
- For a local choice at a BGP router

```
neighbor IP-address weight weight-value
```

- The route passing via the neighbor of the largest weight will be chosen
- Never advertised

42

Choice of the best route

- Done by **decision process**; route installed in Loc-RIB
- At most one best route to exactly the same prefix is chosen
  - Only one route to 2.2/16 can be chosen
  - But there can be different routes to 2.2.2/24 and 2.2/16
- Route validation: check if NEXT_HOP is accessible
- Decreasing priority (configurable, skip some steps)
  1. max WEIGHT
  2. max LOCAL_PREF
  3. shortest AS_PATH
  4. ORIGIN attribute IGP > EGP > INCOMPLETE
  5. min MULTI_EXIT_DISC
  6. shortest IGP distance to NEXT_HOP
  7. source of the route: E-BGP > I-BGP
  8. route advertised by router having the smallest IP address

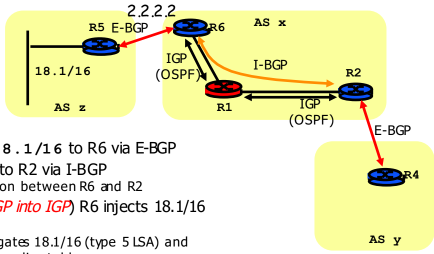
43

Interaction BGP—IGP—Packet Forwarding

- How BGP routers inform all the routers in their AS about prefixes they learn?
- There are main two interactions between BGP and internal routing that you have to know
- *Redistribution*: routes learnt by BGP are passed to IGP (ex: OSPF)
  - Called "redistribution of BGP into OSPF"
  - OSPF propagates the routes using type 5 LSAs to all routers in OSPF cloud
- *Injection*: routes learnt by BGP are written into the forwarding table of this router
  - Routes do not propagate; this helps only this router

44

Redistribution Example

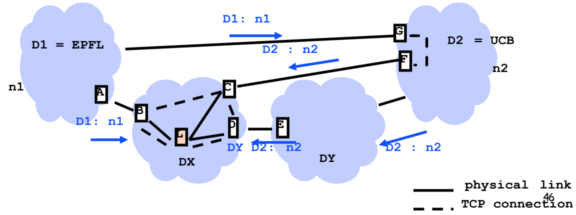


- R5 advertises 18.1/16 to R6 via E-BGP
- R6 transmits it to R2 via I-BGP
  - TCP connection between R6 and R2
- (*redistribute BGP into IGP*) R6 injects 18.1/16 into IGP (OSPF)
  - OSPF propagates 18.1/16 (type 5 LSA) and updates forwarding tables
  - After OSPF converges, R1, R2 now have a route to 18.1/6
- R2 advertises route to R4 via E-BGP
  - (*synchronize with IGP*) R2 must wait for the OSPF entry to 18.1/16 before advertising via E-BGP
- Packet to 18.1/16 from AS y finds forwarding table entries in R2, R1 and R6

45

Example with Re-Distribution

- by I-BGP, F learns from G the route to D2-D1-n1
- C redistributes the external route D2:n2 into OSPF;
- by I-BGP, D learns the route D2:n2; by E-BGP D learns the route DYD2:n2; D selects D2:n2 and does not redistribute it to OSPF
- by I-BGP, B learns the route D2:n2 from C
- by E-BGP, A learns the route DX:D2:n2
- by OSPF, L learns the route to n2 via C



46

Re-Distribution Considered Harmful

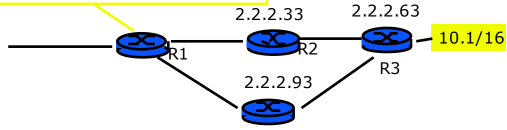
- In practice, operators avoid re-distribution of BGP into IGP
  - Large number of routing entries in IGP
  - Reconvergence time after failures is large if IGP has many routing table entries
- A classical solution is based on *recursive table lookup*
  - When IP packet is submitted to router, the forwarding table may indicate a "NEXT-HOP" which is not on-link with router
  - A second table lookup needs to be done to resolve the next-hop into an on-link neighbour
    - in practice, second lookup is done in advance – not in real time– by preprocessing the routing table

47

Example: Recursive Table Lookup

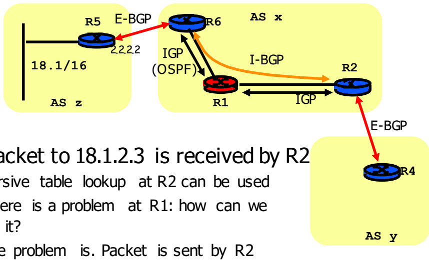
- At R1, data packet to 10.1.x.y is received
- The forwarding table at R1 is looked up
  - Q: what are the next events ?
  - A: first, the nex-hop 2.2.2.63 is found; a second lookup for 2.2.2.63 is done; the packet is sent to MAC address x09:F1:6A:33:76:21

To	*NEXT-HOP*	*layer-2 addr*
10.1/16	2.2.2.63	N/A
2.2.2.63	2.2.2.33	x09:F1:6A:33:76:21



48

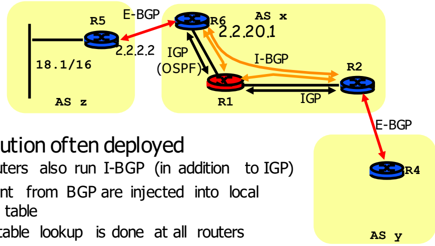
Avoid Redistribution: Combine Recursive Lookup and NEXT-HOP



- Data packet to 18.1.2.3 is received by R2
  - Recursive table lookup at R2 can be used
  - Q: there is a problem at R1: how can we solve it?
  - A: the problem is. Packet is sent by R2 towards R1, which is the neighbour towards 2.2.2.2. At R1, there is no forwarding entry to 18.1/16 – blackhole. See next slide...

49

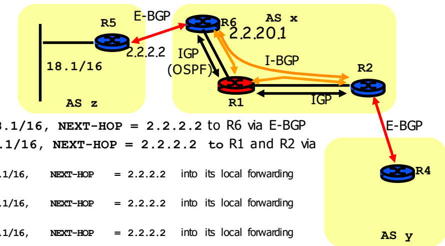
Practical Solution: run BGP everywhere



- Practical solution often deployed
  - All core routers also run I-BGP (in addition to IGP)
  - Routes learnt from BGP are injected into local forwarding table
  - Recursive table lookup is done at all routers
  - Q: repeat the sequence of previous slide with this new assumption
- Potential problem: I-BGP mesh -> use reflectors
  - see next

50

Practical Solution: run BGP everywhere



- R5 advertises `18.1/16, NEXT-HOP = 2.2.2.2` to R6 via E-BGP
- R6 transmits `18.1/16, NEXT-HOP = 2.2.2.2 to` R1 and R2 via I-BGP
  - R6 *injects* `18.1/16, NEXT-HOP = 2.2.2.2` into its local forwarding table
  - R1 *injects* `18.1/16, NEXT-HOP = 2.2.2.2` into its local forwarding table
  - R2 *injects* `18.1/16, NEXT-HOP = 2.2.2.2` into its local forwarding table
- Independently, IGP finds that at R2 packets to 2.2.10.1 should be sent to R1 (route to R6 goes through R1)
- Data packet to 18.1.2.3 is received by R2
  - At R2, recursive table lookup determines that packet should be forwarded to R1
  - At R1, recursive table lookup determines that packet should be forwarded to R6 (2.2.10.1)
  - At R6, table lookup determines that packet should be forwarded to 2.2.2.2

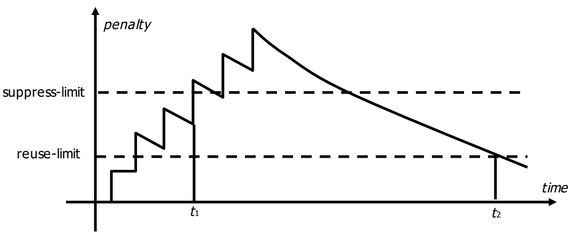
51

Route dampening

- Route modification propagates everywhere
  - successive UPDATE and WITHDRAW of a route
- Sometimes routes are *flapping*
  - successive UPDATE and WITHDRAW
  - caused for example by BGP speaker that often crashes and reboots
- Solution:
  - decision process eliminates flapping routes
- How
  - withdrawn routes are kept in Adj-RIN-in
  - if comes up again soon (ie : flap), route receives a penalty
  - penalty fades out exponentially (halved at each half-life-time)
  - used to suppress or restore routes
- Thresholds: suppress-limit, reuse-limit

52

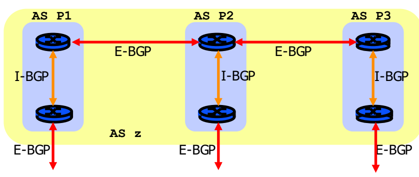
Route dampening



- Route suppressed at t_1, restored at t_2

53

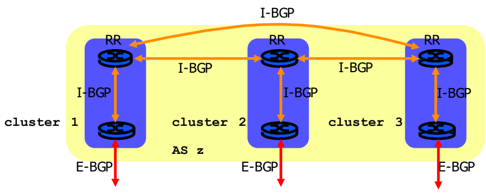
Avoid I-BGP Mesh: Confederations



- AS decomposed into sub-AS
  - private AS number
  - similar to OSPF areas
    - I-BGP inside sub-AS (full interconnection)
    - E-BGP between sub-AS

54

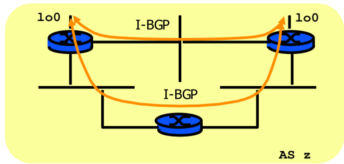
Avoid I-BGP Mesh: Route reflectors



- Cluster of routers
  - one I-BGP session between one client and RR
  - CLUSTER_ID
- Route reflector
  - re-advertises a route learnt via I-BGP
  - to avoid loops
    - ORIGINATOR_ID attribute associated with the advertisement

55

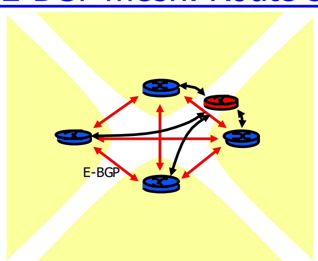
I-BGP configuration



- I-BGP configured on loopback interface (lo0)
  - interface always up
  - IP address associated with the interface
  - IGP routing guarantees packet forwarding to the interface

56

Avoid E-BGP mesh: Route server



- At interconnection point
- Instead of $n(n-1)/2$ peer to peer E-BGP connections n connections to Route Server
- To avoid loops ADVERTISER attribute indicates which router in the AS generated the route

57

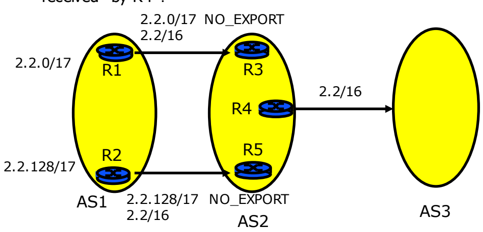
COMMUNITY

- Other attributes can be associated with routes in order to simplify rules. They are called « communities »
  - mark routes that share a common property
  - signal routes that needs to be processed in a predefined way
- Standard well-known values
  - NO_EXPORT - the route should not be advertised to peers outside a confederation
  - NO_ADVERTIZE - the route should not be advertised to any peer
- Defined by one AS
  - a label of the form AS-no:x, x - value (0-65535)
- Transitive

58

NO_EXPORT

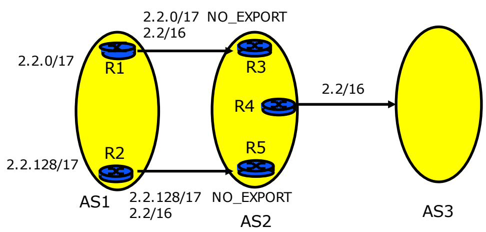
- Written on E-BGP by one AS, transmitted on I-BGP by accepting AS, not forwarded
- Example: AS2 has different routes to AS1 but AS2 sends only one aggregate route to AS3
  - simplifies the aggregation rules at AS2
  - What is the route followed by a packet sent to 2.2.48 received by R4 ?



59

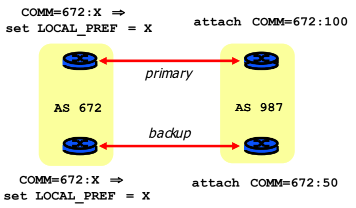
NO_EXPORT

- Q: What is the route followed by a packet sent to 2.2.48 received by R4 ?
- A: the packet is sent via R3 and R1



60

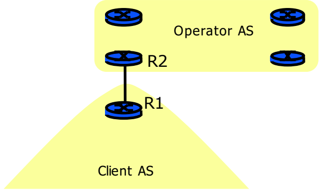
COMMUNITY



- Set LOCAL_PREF according to community values

61

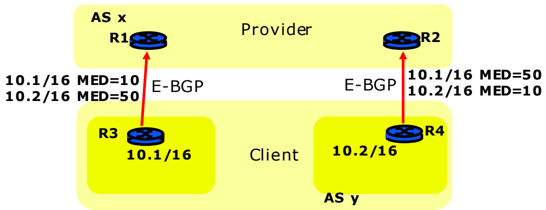
Ex1: Stub AS



- BGP not needed between Client and Operator
- No AS number for client
- R2 learns all prefixes in Client by static configuration or IGP on link R1—R2
- what if R1 fails ?

62

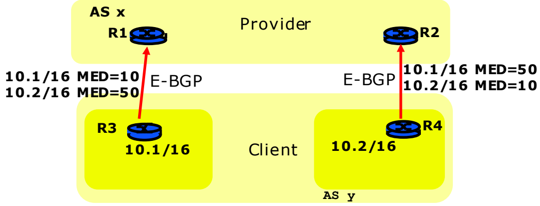
Ex2: Dual Homing to Single Provider



- With numbered Client AS
  - Use MED to share traffic from ISP to Client on two links
  - Use Client IGP configuration to share traffic from Client on two links
  - Q1: is it possible to avoid distributing BGP routes into Client IGP ?
  - Q2: is it possible to avoid assigning an AS number to Client ?
  - Q3: is it possible to avoid BGP between Client and Provider ?

63

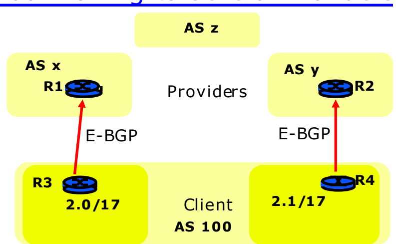
Ex2: Dual Homing to Single Provider



- Q1: is it possible to avoid distributing BGP routes into Client IGP ?
- A: yes, for example: configure R3 and R4 as default routers in Client AS; traffic from Client AS is forwarded to nearest of R3 and R4. If R3 or R4 fails, to the remaining one
- Q2: is it possible to avoid assigning an AS number to Client ?
- A: Yes, it is sufficient to assign to Client a private AS number: Provider translates this number to its own.
- Q3: is it possible to avoid BGP between Client and Provider ?
- A: Yes, by running a protocol like RIP between Client and Provider and redistributing Client routes into Provider IGP. Thus Provider pretends to the rest of the world that the prefixes of Client are its own.

64

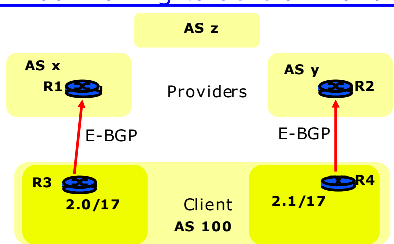
Ex3: Dual Homing to Several Providers



- Client has its own address space and AS number
- Q: how can routes be announced between AS 100 and AS x? AS x and AS z?
- Q: assume Client wants most traffic to favor AS y How can that be done?

65

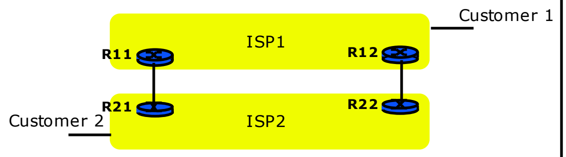
Ex3: Dual Homing to Several Providers



- Client has its own address space and AS number
- Q: how can routes be announced between AS 100 and AS x? AS x and AS z?
  A: R3 announces 2.0/17 and 2.0/16; traffic from AS x to 2.0/17 will flow via AS x; if R3 fails, it will use the longer prefix and flow via AS y.
  AS x announces 2.0/17 and 2.0/16 to AS z
- Q: assume Client wants most traffic to prefer AS y. How can that be done?
  A: R3 announces an artificially inflated path: 100 100 100 100 : 2.0/17. AS z will favour the path via AS y which has a shorter AS path length

66

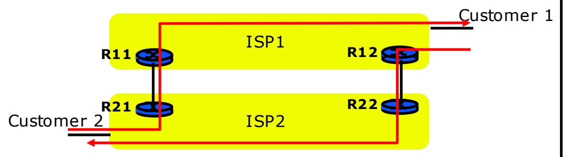
Ex4: Hot Potato Routing



- Packets from Customer 2 to Customer 1
  - Both R21 and R22 have a route to Customer 1
  - Shortest path routing favors R21
  - Q1: by which mechanism is that done?
- Q2: what is the path followed in the reverse direction?

67

Ex4: Hot Potato Routing



- Packets from Customer 2 to Customer 1
  - Both R21 and R22 have a route to Customer 1
  - Shortest path routing favors R21
  - Q1: by which mechanism is that done?
  - A: « Choice of the best route » (criterion 7), assuming all routers in ISP2 run BGP
- Q2: what is the path followed in the reverse direction?
  - A: see picture. Note the asymmetric routing

68

Route filtering

- Associate an access list with a neighbor

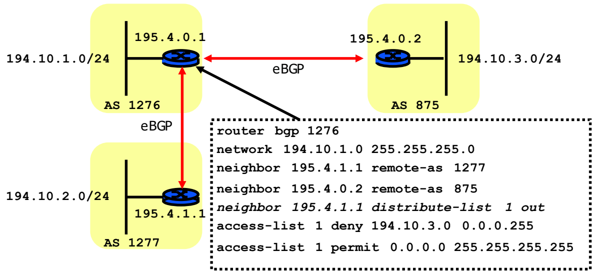
neighbor *ID* **distribute-list** *no-of-the-list* **[in/out]**

- Define an access list
  - non-significant-bits (inverse of the netmask)
  - if no action specified at the end of the list, apply "deny everything else"

access-list *No-of-the-list* **[deny/permit]**
IP-address non-significant-bits

69

Route filtering



- AS 1276 does not want to forward traffic to `194.10.3.0/24` of AS 875 - it does not re-advertise this prefix

70

Path filtering

- Associate a filter list with a neighbor

neighbor *ID* **filter-list** *no-of-the-list* **[in/out]**

- Define a filter list

ip as-path access-list *no-of-the-list* **[deny/permit]**
regular-expression

- Regular expressions

```
^  beginning of the path
$  end of the path
.  any character
?  one character
_  matches ^ $ ( ) 'space'
*  any number of characters (zero included)
+  any number of characters (at least one)
```

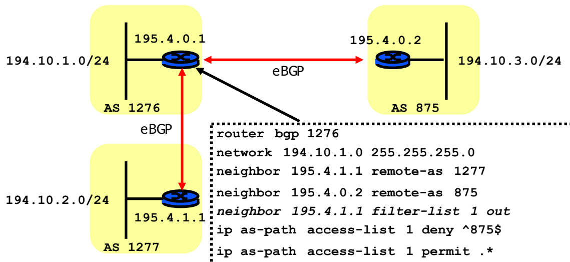
71

Path filtering

- Examples
  - `^$` - local routes only (empty AS_PATH)
  - `.*` - all routes (all paths AS_PATH)
  - `^300$` - AS_PATH = 300
  - `^300_` - all routes coming from 300 (e.g. AS_PATH = 300 200 100)
  - `_300$` - all routes originated at 300 (e.g. AS_PATH = 100 200 300)
  - `_300_` - all routes passing via 300 (e.g. AS_PATH = 200 300 100)

72

Path filtering




```
router bgp 1276
network 194.10.1.0 255.255.255.0
neighbor 195.4.1.1 remote-as 1277
neighbor 195.4.0.2 remote-as 875
neighbor 195.4.1.1 filter-list 1 out
ip as-path access-list 1 deny ^875$
ip as-path access-list 1 permit .*
```

- AS 1276 does not want to forward traffic for all internal routes of AS 875

73

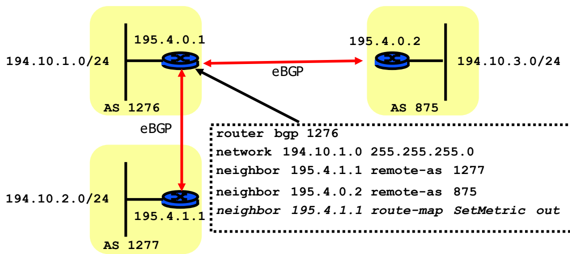
Route maps

```
route-map map-tag [permit|deny] instance-no
first-instance-conditions: set match
next-instance-conditions: set match
...
route-map SetMetric permit 10
match ip address 1
set metric 200

route-map SetMetric permit 20
set metric 300

access-list 1 permit 194.10.3.0 0.0.0.255
```

- Set metric 200 (MED) on route `194.10.3/24`

74

Route maps




```
router bgp 1276
network 194.10.1.0 255.255.255.0
neighbor 195.4.1.1 remote-as 1277
neighbor 195.4.0.2 remote-as 875
neighbor 195.4.1.1 route-map SetMetric out
```

- Set metric 200 on route `194.10.3/24`, 300 otherwise

75

Route maps

```
neighbor 192.68.5.2 route-map SetLocal in

route-map SetLocal permit 10
set local-preference 300
```

- Set LOCAL_PREF to 300

```
neighbor 172.16.2.2 route-map AddASnum out

route-map AddASnum permit 10
set as-path prepend 801 801
```

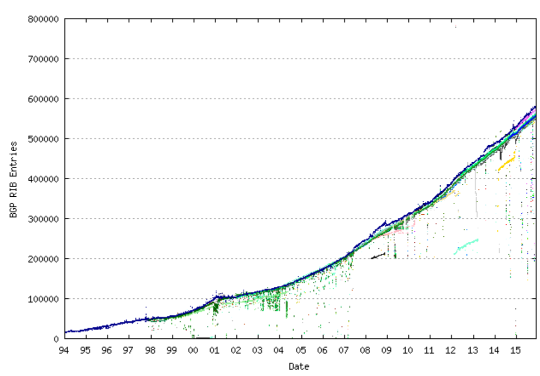
- Prepend AS 801 801 to AS_PATH (makes it longer)

76

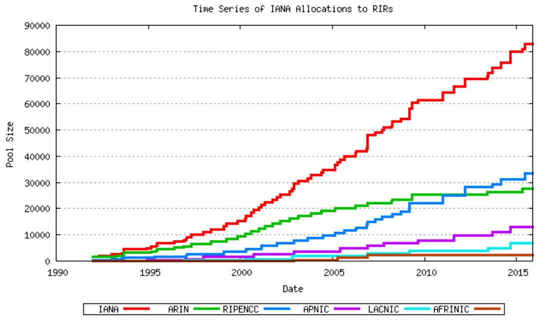
Some statistics

- Number of routes
  - 1988-1994: exponential increase
  - 1994-1995: CIDR
  - 1995-1998: linear increase (10000/year)
  - 1999-2000: return to exponential increase (42% per year)
  - since 2001: return to linear increase, ~120,000
- Number of ASs
  - 51% per year for 4 last years
  - 14000 AS effectively used
- Number of IP addresses
  - 162,128,493 (Jul 2002)
  - 7% per year

77

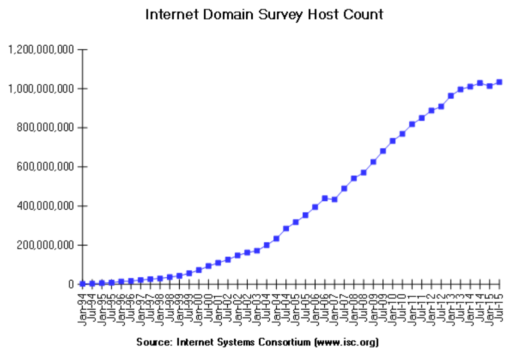


78



79

Number of hosts



80

BGP statistics

BGP routing table entries examined:	117013
Total ASes present in the Internet Routing Table:	14042
Origin-only ASes present in the Internet Routing Table:	12159
Transit ASes present in the Internet Routing Table:	1883
Transit-only ASes present in the Internet Routing Table:	63
Average AS path length visible in the Internet Routing Table:	5.3
Max AS path length visible:	23
Number of addresses announced to Internet:	1182831464
Equivalent to 70 /8s, 128 /16s and 147 /24s	
Percentage of available address space announced:	31.9
Percentage of allocated address space announced:	58.5

81

Prefix length distribution

/1:0 /2:0 /3:0 /4:0 /5:0 /6:0
/7:0 /8:17 /9:5 /10:8 /11:12 /12:46
/13:90 /14:239 /15:430 /16:7308 /17:1529 /18:2726
/19:7895 /20:7524 /21:5361 /22:8216 /23:9925 /24:64838
/25:185 /26:221 /27:126 /28:105 /29:85 /30:93
/31:0 /32:29

82

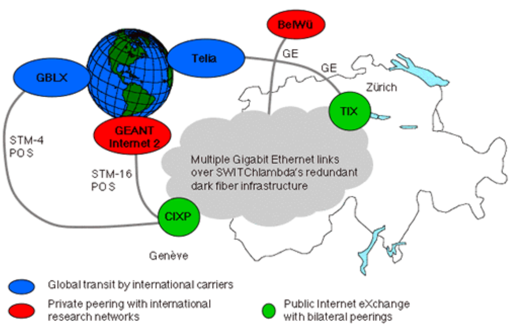
AS 559 - SWITCH

AS559 SWITCH-AS SWITCH Teleinformatics Services
Adjacency: 3 Upstream: 2 Downstream: 1
Upstream Adjacent AS list
AS1299 TCN-AS Telia Corporate Network
AS3549 GBLX Global Crossing
Downstream Adjacent AS list
AS4128 RG-SPARE RGnet, Inc.

Prefix	(AS Path)
128.178.0.0/15	1 3549 559
129.129.0.0/16	1 3549 559
129.132.0.0/16	1 3549 559

83

Switch



84

AS 1942 - CICG-GRENOBLE

AS1942 AS1942 FR-CICG-GRENOBLE
Adjacency: 1 Upstream: 1 Downstream: 0
Upstream Adjacent AS list
AS2200 AS2200 RENATER 2

Prefix	(AS Path)
129.88.0.0/16	1239 5511 2200 1942
130.190.0.0/16	1239 5511 2200 1942
147.171.0.0/16	1239 5511 2200 1942
147.173.0.0/16	1239 5511 2200 1942

2200 - Renater-2, 5511 - OpenTransit (FT), 1239 - Sprint

85

Looking glass at genbb1.opentransit.net

```
sh ip bgp 129.88.38.241
BGP routing table entry for 129.88.0.0/16, version 34110212
 2200 1942
  193.51.185.30 (metric 16) from 193.251.128.5 (193.251.128.1)
   Origin IGP, localpref 100, valid, internal
   Community: 2200:1001 2200:2200 5511:211 5511:500 5511:503 5511:999
   Originator: 193.251.128.1, Cluster list: 0.0.0.10
 2200 1942
  193.51.185.30 (metric 16) from 193.251.128.3 (193.251.128.1)
   Origin IGP, localpref 100, valid, internal
   Community: 2200:1001 2200:2200 5511:211 5511:500 5511:503 5511:999
   Originator: 193.251.128.1, Cluster list: 0.0.0.10
 2200 1942
  193.51.185.30 (metric 16) from 193.251.128.1 (193.251.128.1)
   Origin IGP, localpref 100, valid, internal, best
   Community: 2200:1001 2200:2200 5511:211 5511:500 5511:503 5511:999
```

86

From genbb1.opentransit.net

```
Tracing the route to horus.imag.fr (129.88.38.1)

 1 P8-0-0.GENAR1.Geneva.opentransit.net (193.251.242.130) 0 msec 0 msec 0 msec
 2 P6-0-0.GENAR2.Geneva.opentransit.net (193.251.150.30) 0 msec 4 msec 0 msec
 3 P4-3.BAGBB1.Bagnolet.opentransit.net (193.251.154.97) 8 msec 8 msec 8 msec
 4 193.51.185.30 [AS 2200] 16 msec 16 msec 16 msec
 5 grenoble-pos1-0.cssi.renater.fr (193.51.179.238) [AS 2200] 16 msec 20 msec 16 ms
 6 tigre-grenoble.cssi.renater.fr (195.220.98.58) [AS 2200] 20 msec 20 msec 20 msec
 7 r-campus.grenet.fr (193.54.184.45) [AS 1942] 20 msec 16 msec 16 msec
 8 r-imag.grenet.fr (193.54.185.123) [AS 1942] 20 msec 20 msec 20 msec
 9 horus.imag.fr (129.88.38.1) [AS 1942] 16 msec 20 msec 20 msec
```

87

Looking glass at genbb1.opentransit.net

```
sh ip bgp 128.178.50.92
BGP routing table entry for 128.178.0.0/15, version 30024182
 1299 559
  193.251.252.22 (metric 13) from 193.251.128.5 (193.251.128.4)
   Origin IGP, metric 100, localpref 85, valid, internal
   Community: 5511:666 5511:710
   Originator: 193.251.128.4, Cluster list: 0.0.0.10
 1299 559
  193.251.252.22 (metric 13) from 193.251.128.3 (193.251.128.4)
   Origin IGP, metric 100, localpref 85, valid, internal
   Community: 5511:666 5511:710
   Originator: 193.251.128.4, Cluster list: 0.0.0.10
 1299 559
  193.251.252.22 (metric 13) from 193.251.128.1 (193.251.128.4)
   Origin IGP, metric 100, localpref 85, valid, internal, best
   Community: 5511:666 5511:710
   Originator: 193.251.128.4, Cluster list: 0.0.0.10
```

88

From genbb1.opentransit.net

```
Tracing the route to empc19.epfl.ch (128.178.50.92)
 1 P5-1.PASBB1.Pastourelle.opentransit.net (193.251.150.25) 8 msec
   P4-1.PASBB1.Pastourelle.opentransit.net (193.251.242.134) 8 msec
   P5-1.PASBB1.Pastourelle.opentransit.net (193.251.150.25) 8 msec
 2 P8-0.PASBB2.Pastourelle.opentransit.net (193.251.240.102) 8 msec 8 msec 8 msec
 3 Telia.GW.opentransit.net (193.251.252.22) 8 msec 12 msec 8 msec
 4 prs-bb1-pos0-3-0.telia.net (213.248.70.1) [AS 1299] 8 msec 8 msec 8 msec
 5 ffm-bb1-pos2-1-0.telia.net (213.248.64.190) [AS 1299] 16 msec 16 msec 16 msec
 6 zch-b1-pos6-1.telia.net (213.248.65.42) [AS 1299] 48 msec 32 msec 48 msec
 7 dante-01287-zch-b1.c.telia.net (213.248.79.190) [AS 1299] 44 msec 36 msec 44
   msec
 8 swiEZ2-G3-2.switch.ch (130.59.36.249) [AS 559] 36 msec 44 msec 36 msec
 9 swiLS2-G2-3.switch.ch (130.59.36.33) [AS 559] 36 msec 36 msec 36 msec
10 * * *
```

89

Conclusion

- BGP
  - essential to the current structure of the Internet
  - influence the choice of the IGP routing - OSPF recommended
  - AS numbers exhaustion - extended to 32 bits
  - complex - policy management, filtering
  - bad configuration - route suppression

90