CS557: Inter-domain Routing

Christos Papadopoulos
Inter-domain Routing

BGP
Sources

- John Stewart III: “BGP4 - Inter-domain routing in the Internet”
- RFC1771: main BGP RFC
- RFC1772-3-4: application, experiences, and analysis of BGP
- RFC1965: AS confederations for BGP
- Christian Huitema: “Routing in the Internet”, chapters 8 and 9
- Cisco tutorial on line
Autonomous Systems

• What is an AS?
  – a set of routers under a single technical administration
  – uses an *interior gateway protocol (IGP)* and common metrics to route packets within the AS
  – uses an *exterior gateway protocol (EGP)* to route packets to other AS’s

• AS may use multiple IGPs and metrics, but appears as single AS to other AS’s
Example
History

• Mid-80s: EGP
  – reachability protocol (no shortest path)
  – did not accommodate cycles (tree topology)
  – evolved when all networks connected to ARPANET

• Limited size network topology

• Result: BGP introduced as routing protocol
Choices

• Link state or distance vector?
  – no universal metric - policy decisions

• Problems with distance-vector:
  – Bellman-Ford algorithm slow to converge (counting to infinity problem)

• Problems with link state:
  – metric used by routers not the same - loops
  – LS database too large - entire Internet
  – may expose policies to other AS’s
Solution: Path Vectors

- Each routing update carries the entire path
- Loops are detected as follows:
  - when AS gets route check if AS already in path
    - if yes, reject route
    - if no, add self and (possibly) advertise route further
- Advantage:
  - metrics are local - AS chooses path, protocol ensures no loops
Interconnecting BGP Peers

• BGP uses TCP to connect peers (port 179)
• Advantages:
  – BGP much simpler
  – no need for periodic refresh - routes are valid until withdrawn, or the connection is lost
  – incremental updates
• Disadvantages
  – congestion control on a routing protocol?
Hop-by-hop Model

• BGP advertises to neighbors only those routes that it uses
  – consistent with the hop-by-hop Internet paradigm
  – e.g., AS1 cannot tell AS2 to route to other AS’s in a manner different than what AS2 has chosen (need source routing for that)
AS Categories

– **Stub**: an AS that has only a single connection to one other AS - carries only local traffic
– **Multi-homed**: an AS that has connections to more than one AS, but does not carry transit traffic
– **Transit**: an AS that has connections to more than one AS, and carries both transit and local traffic (under certain policy restrictions)
Policy With BGP

- BGP provides capability for enforcing various policies
- Policies are **not** part of BGP: they are provided to BGP as configuration information
- BGP enforces policies by choosing paths from multiple alternatives and controlling advertisement to other AS’s
Examples of BGP Policies

• A multi-homed AS refuses to act as transit
  – limit path advertisement
• A multi-homed AS can become transit for some AS’s
  – only advertise paths to some AS’s
• An AS can favor or disfavor certain AS’s for traffic transit from itself
  – Pick appropriate routes by examining path vectors
BGP Is NOT Needed If:

- Single homed network (stub)
- AS does not provide downstream routing
- AS uses a default route
BGP-4

- Latest version of BGP
- BGP-4 supports CIDR
Routing Information Bases (RIB)

- Routes are stored in RIBs
- **Adj-RIBs-In**: routing info that has been learned from other routers (unprocessed routing info)
- **Loc-RIB**: local routing information selected from Adj-RIBs-In (routes selected locally)
- **Adj-RIBs-Out**: info to be advertised to peers (routes to be advertised)
BGP Common Header

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Marker (security and message delineation) 16 bytes</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Length (2 bytes)</td>
<td>Type (1 byte)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Types: OPEN, UPDATE, NOTIFICATION, KEEPALIVE
**BGP OPEN Message**

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Marker (security and message delineation)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Length</th>
<th>Type: open</th>
<th>version</th>
</tr>
</thead>
<tbody>
<tr>
<td>My autonomous system</td>
<td>Hold time</td>
<td></td>
</tr>
</tbody>
</table>

**BGP identifier**

<table>
<thead>
<tr>
<th>Parameter length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optional parameters &lt;type, length, value&gt;</td>
</tr>
</tbody>
</table>

- My AS: id assigned to that AS
- Hold timer: max interval between KEEPALIVE or UPDATE messages
- BGP ID: address of one (typically virtual) interface and is same for all messages
**BGP UPDATE Message**

![Diagram of BGP UPDATE Message]

<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Marker (security and message delineation)</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Length</strong></td>
<td><strong>Type: update</strong></td>
<td><strong>Withdrawn..</strong></td>
</tr>
<tr>
<td></td>
<td>.routes len</td>
<td>Withdrawn routes (variable)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Path attribute len</td>
<td>Path attributes (variable) - origin, path, metrics, etc.</td>
<td></td>
</tr>
</tbody>
</table>

- UPDATE message may report multiple withdrawn routes.
- Many prefixes may be included in UPDATE, but must share same attributes.
NLRI

- Network Level Reachability Information
  - list of IP prefixes encoded as follows:

<table>
<thead>
<tr>
<th>Length (1 byte)</th>
<th>Prefix (variable)</th>
</tr>
</thead>
</table>

Path Attributes

Type-Length-Value encoding

Attribute type (2 bytes) Attribute length (1-2 bytes)

Attribute Value (variable)

Attribute type field

Attribute flags (1 byte) Attribute type code (1 byte)

Flags: optional v.s. well-known
transitive v.s. non-transitive (passed on)
partial (someone in path did not understand this attribute)
extended length (2 bytes instead of 1)

Attribute types: Origin, AS_PATH, Next_Hop (more later..)
BGP NOTIFICATION Message

Used for error notification (update error, expired timer, FSM, cease) TCP connection is closed *immediately* after notification.
BGP KEEPALIVE Message

Sent periodically (but before hold timer expires) to peers to ensure connectivity. Sent in place of an UPDATE message.

Note: hold_time = zero means no keepalives will be sent
Path Selection Criteria

- Information based on path attributes
- Attributes + external (policy) information
- Examples:
  - hop count
  - policy considerations
  - presence or absence of certain AS
  - path origin (EGP, IGP)
  - link dynamics (flapping, stable)
Path Attributes

• Categories (recall flags):
  – well-known mandatory (passed on)
  – well-known discretionary (passed on)
  – optional transitive (passed on)
  – optional non-transitive (if unrecognized, not passed on)

• Optional attributes allow for BGP extensions
Path Attribute Message Format (Repeated)

Attribute flags

- **O**: optional or well-known
- **T**: transitive or local
- **P**: partially evaluated
- **E**: length in 1 or 2 bytes

Attribute type code

<table>
<thead>
<tr>
<th>O</th>
<th>T</th>
<th>P</th>
<th>E</th>
<th>0</th>
</tr>
</thead>
</table>

Origin
AS_path
Next hop
etc.
ORIGIN Path Attribute

• Well-known, mandatory attribute
• Describes how a prefix was generated at the origin AS. Possible values:
  – **IGP**: prefix learned from IGP
  – **EGP**: prefix learned through EGP
  – **INCOMPLETE**: none of the above (often seen for static routes)
AS_PATH Attribute

- Well-known, mandatory attribute
- Important components:
  - list of traversed AS’s
- If forwarding to internal peer:
  - do not modify AS_PATH attribute
- If forwarding to external peer:
  - prepend self into the path
AS_PATH Attribute

AS-Path

AS 200
AS 100
AS 300
AS 400
AS 500

170.10.0.0/16
170.10.0.0/16
150.10.0.0/16
150.10.0.0/16
180.10.0.0/16
180.10.0.0/16
300 200 100
300 200 100
300 200
300 400
CIDR and BGP

What should T announce to Z?
Options

• Advertise all paths:
  – Path 1: through T can reach 197.8.0.0/23
  – Path 2: through T can reach 197.8.2.0/24
  – Path 3: through T can reach 197.8.3.0/24

• But this does not reduce routing tables! We would like to advertise:
  – Path 1: through T can reach 197.8.0.0/22
Sets and Sequences

- Problem: what do we list in the route?
  - list T: omitting information not acceptable, may lead to loops
  - list T, X, Y: misleading, appears as 3-hop path
- Solution: restructure AS Path attribute as:
  - path: (Sequence (T), Set (X, Y))
  - if Z wants to advertise path:
    - Path: (Sequence (Z, T), Set (X, Y))
  - in practice used only if paths in set have same attributes
Next Hop Path Attribute

• Well-known, mandatory attribute
• NEXT_HOP: IP address of border router to be used as next hop
• Usually, next hop is the router sending the UPDATE message
• Useful when some routers do not speak BGP
Example of NEXT_HOP

A (BGP) ➔ UPDATE MSG through BGP ➔ B (BGP)

Traffic to 138.39.0.0/16

C (no BGP)

138.39.0.0/16
LOCAL PREF Path Attribute

• Well-known, discretionary
• Provided by a BGP router to all other internal BGP routers
  – denotes degree of preference for each destination
LOCAL-PREF

Local Preference

AS 100
160.10.0.0/16

AS 200

AS 400

AS 300

160.10.0.0/16  500
> 160.10.0.0/16  200

Higher preference wins
You are AS1 with two links A & B to AS2. How to force all traffic to AS2’s prefix 12/24 through link A?

At a:
LP=10 for 12/24:b
At c:
LP=5 for 12/24:d
LOCAL-PREF Example 2

You are AS1 with two links A & B to AS2. How to load-share AS2-bound traffic between links A & B?

a: 12.0/25:b w/LP 10
a: 12.128/25:b w/LP 5

You are AS1 with two links A & B to AS2. How to load-share AS2-bound traffic between links A & B?

a: 12.0/25:b w/LP 10
a: 12.128/25:b w/LP 5
c: 12.0/25:d w/LP 5
c: 12.128/25:d w/LP 10
Multi-exit Discriminator (MED) 
Path Attribute

• Optional, non-transitive attribute
• Used when two AS’s connect to each other in more than one place
• Carries a metric expressing degree of preference
AS2 includes MEDs with prefixes sent to AS1 over Links A and B. AS1 uses these to select appropriate link when sending to PFX.
MED (Multi-Exit Discriminator)

Lower cost wins
AS2 can use MED to instruct AS1 to use link A for traffic to AS3, and link B for traffic to AS4. How is this done?
MED Example

You are AS1 with two links A & B to AS2. How can you make AS2 send north traffic to link A and south traffic to link B?

11.0/16:a (MED: 10)
11.0/16:c (MED: 20)
11.1/16:a (MED: 20)
11.1/16:c (MED: 10)
MED (Continued)

MED is typically used in provider/subscriber scenarios. It can lead to unfairness if used between ISPs because it may force one ISP to carry more traffic:

- ISP1 ignores MED from ISP2
- ISP2 obeys MED from ISP1
- ISP2 ends up carrying traffic most of the way
Why Is MED Non-transitive?

- MEDs relative to ISPs
- AS3 and AS4 only have one link to AS1
- AS4 learns two ways to reach PFX, one from AS2 and another from AS3; but does not make sense to compare - not learned from same AS
External v.s. Internal BGP

E-BGP v.s. I-BGP
Internal v.s. External BGP

• BGP can be used by R3 and R4 to learn routes.
• How do R1 and R2 learn routes?
• Option 1: Inject routes in IGP
  • only works for small routing tables
• Option 2: Use I-BGP
I-BGP

Upstream Provider A
AS100
eBGP

iBGP
AS 1

Upstream Provider B
AS200
eBGP

iBGP
AS 2
Internal BGP (I-BGP)

- Same messages as E-BGP
- Different rules about re-advertising prefixes:
  - prefix learned from E-BGP can be advertised to I-BGP neighbor and vice-versa, but
  - prefix learned from one I-BGP neighbor cannot be advertised to another I-BGP neighbor
  - reason: no AS PATH within the same AS and thus danger of looping
Internal BGP (I-BGP)

- R3 can tell R1 and R2 prefixes from R4
- R3 can tell R4 prefixes from R1 and R2
- R3 cannot tell R2 prefixes from R1

R2 can only find these prefixes through a direct connection to R1
Result: I-BGP routers must be fully connected (via TCP)!
  - contrast with E-BGP sessions that map to physical links
I-BGP

Upstream Provider A
AS100

Upstream Provider B
AS200

I-BGP mesh
Route Selection

• Question: which routes should be installed in the forwarding table?
• Input: All routes that have been learned and accepted by local BGP
  – If only one route, then select it
  – If multiple routes (with same length prefix) then we have a decision to make
UPDATE Message Handling

- Unrecognized, optional, non-transitive attributes are ignored. Unrecognized, optional, transitive attributes cause the Partial bit to be set.
- WITHDRAWN routes are processed first.
- Feasible routes are placed in Adj-RIB-In, replacing old ones, if any.
Decision Process

• Calculate degree of preference for each route in Adj-RIB-In as follows (apply following steps until one route is left):
  – select route with highest LOCAL-PREF
  – select route with shortest AS-PATH
  – apply MED (if routes learned from same neighbor)
  – select route with smallest NEXT-HOP cost
...Decision Process

- select route learned from E-BGP peer with lowest BGP ID
- select route from I-BGP neighbor with lowest BGP ID

- Install selected route in Loc-RIB
- Disseminate routes to peers, update Adj-RIB-Out
- Done
BGP Operations
BGP Example

R1 advertises routes within AS1 to R2
R2 advertises routes within AS2 and AS3 to R1
R2 learns AS3 routes from I-BGP with R4
R4 learns AS3 routes from E-BGP with R6
R4 advertises routes within AS2 and AS1 to R6
Link Failures

- Two types of link failures:
  - failure on an E-BGP link
  - failure on an I-BGP Link
- These failures are treated completely different in BGP
- Why?
Failure on an E-BGP Link

If the link R1-R2 goes down, then the TCP connection breaks and so does the E-BGP connection; BGP routes are removed. This is the *desired* behavior.
Failure on an I-BGP Link

If link R1-R2 goes down, R1 and R2 should still be able to exchange traffic! The indirect path through R3 must be used.

Thus, E-BGP and I-BGP must use different conventions w.r.t. TCP endpoints.
Virtual Interfaces (VIFs, a.k.a. Loop-back Interfaces)

A VIF is not associated with a physical link or hardware interface. How do routers learn of VIF addresses? Using IGP.
Multi-homing

• With multi-homing, a single network has more than one connections to the Internet
• Improves reliability and performance:
  – can accommodate link failure
  – bandwidth is sum of links to Internet
• Multiple connections provide *load sharing* but not load balancing
  – BGP cannot do load balancing
Issues With Multi-homing

• Symmetric routing
  – while conventional wisdom prefers symmetric paths, many are asymmetric

• Packet re-ordering
  – may trigger TCP’s fast retransmit algorithm

• Other concerns:
  – addressing, DNS, aggregation
Static routing may send traffic from ISPs 2-n to customer over one link and traffic from ISP1 over the other link. Lacks flexibility.
Multi-homing to a Single Provider: Case 1

- **Easy solution:**
  - use IMUX or Multi-link PPP

- **Harder solution:**
  - use BGP
  - makes assumptions about traffic (same amount of prefixes can be reached from both links)
Multi-homing to a Single Provider: Case 2

- If multiple prefixes, may use MED
  - good if traffic load from prefixes is equal
- If single prefix, load may be unequal
  - break-down prefix and advertise different prefixes over different links
Multi-homing to a Single Provider: Case 3

- For ISP-> customer traffic, same as before:
  - use MED
  - good if traffic load to prefixes is equal
- For customer -> ISP traffic:
  - R3 alternates links (re-ordering?)
  - Customer learns full BGP routes and load-shares
Multi-homing to a Single Provider: Case 4

- Most reliable approach
  - no equipment sharing
- Customer -> ISP:
  - same as case 2
- ISP -> customer:
  - same as case 3
Multi-homing to Multiple Providers

- Major issues:
  - addressing
  - aggregation
- Customer address space:
  - delegated by ISP1
  - delegated by ISP2
  - delegated by ISP1 and ISP2
  - obtained independently
- Advantages and disadvantages?
Case 1: Customer Uses Address Space From One ISP (1 or 2)

- Customer uses address space from ISP1
- ISP1 advertises /16 aggregate
- Customer advertises /24 route to ISP2
- ISP2 relays route to ISP1 and ISP3
- ISP2-3 use /24 route
- ISP1 routes directly
- Problems with traffic load?
Pitfalls

- ISP1 aggregates to a /19 at border router to reduce internal tables.
- ISP1 still announces /16.
- ISP1 hears /24 from ISP2.
- ISP1 routes packets for customer to ISP2!
- Workaround: ISP1 *must* inject /24 into I-BGP.
Case 2: Customer Uses Address Space From Both ISPs

- ISP1 and ISP2 continue to announce aggregates
- Load sharing depends on traffic to two prefixes
- Lack of reliability: if ISP1 link goes down, part of customer becomes inaccessible
- Customer may announce prefixes to both ISPs, but still problems with longest match as in case 1
Case 3: Customer Uses Its Own Address Space

- Offers the most control, but at the cost of aggregation
- Still need to control paths:
  - suppose ISP1 large, ISP2-3 small
  - customer advertises long path to ISP1, but local-pref attribute used to override
  - ISP3 learns shorter path from ISP2
Scaling the I-BGP Mesh

• Two methods:
  – BGP confederations
    • Scale by divide and conquer (sub-AS)
  – Route reflectors
    • Scale by adding hierarchical IBGP route forwarding
AS Confederation

- Subdivide a single AS into multiple, internal sub-AS’s to reduce I-BGP mesh size
- Still advertises a single AS to external peers
R2 does not see sub-AS 10-14, but sees AS1
Confederations

- BGP sessions between sub-AS’s are like regular E-BGP but with some changes:
  - `local-pref` attribute remains meaningful within confederation (E-BGP ignores it)
  - `next-hop` attribute traverses sub-AS boundaries (assumes single IGP running - everyone has same route to `next hop`)
  - AS-PATH now includes AS-CONFED-SET and AS-CONFED-SEQUENCE to avoid loops
BGP Confederation

BGP Confederation

AS10

AS20

AS30

AS300
Route Reflectors

• **Route Reflector (RR):** router whose BGP implementation allows re-advertisement of routes between I-BGP neighbors

• **Route Reflector Client (RRC):** router that depends on RR to re-advertise its routes to entire AS. It also depends on RR to learn routes from the rest of the network
  – RRC runs normal I-BGP
RR Example

With RR there are 7 I-BGP sessions instead of 21
Rules for Route Reflectors

- Reflectors advertise routes learned from clients into the I-BGP mesh
  - RR1 advertises 138.39.0.0/16 learned from RR-C2 into I-BGP
- Reflectors do not re-advertise routes between non-clients
  - RR1 will not re-advertise 128.4.0.0/16 learned from RR3 to RR2
Route Flap Dampening

• Problem: route flap when a flaky link constantly goes up and down:
  – BGP sessions disappear and reappear
  – routes are withdrawn and re-advertised
  – global effects

• BGP was extended to dampen route flaps
Route Flap Dampening (Cont.)

• Associate a penalty with each route
  – increase when route flaps
  – exponentially decay penalty with time

• When penalty reaches threshold, suppress route
  – must never forget routes
Route Flap Dampening (Cont.)

Route Flap Dampening

Penalty

Time

Suppress-Limit

Reuse-Limit

Intro to BGP
How can BGP express the following policies:
2 will not act as transit to 3
2 will not accept packets sourced in 1
1 will use the green path for packets destined to 4 and the red for packets destined to 5
Tricky Issues

• “Synchronizing” intra and inter-domain routing
• Getting packets to the right exit router without introducing too much flux into intra-domain routing
• Multi-homing
  – interaction with aggregation
• How much policy *should* we actually be able to support???
BGP Limitations: Policy
Delayed Internet Routing
Convergence

Labovitz00
BGP Problems: Delayed Convergence

• Question:
  – How long does it take for a route to fail-over?

• How to answer this question:
  – Experimental methodology
  – Explanation of observation using simple model
Key Idea

• Study and understand BGP convergence time
  – simulation
  – measurement

• Suggests bounds of $O(n!)$ and $O((n-3) \times 30s)$
Why Is Convergence Important?

• Robustness
  – PSTN (telephone) fail-over times are in milliseconds
  – Internet fail-over times are in 10s of seconds
Methodology

• Two years of traces
• Introduce artificial faults across Internet
  – but for only their AS, of course!
  – failures, repairs, fail-over
• Measure:
  – Tup: new route, Tdown: old route goes away,
    Tshort & Tlong: a route shrinks or grows
Methodology

Internet-scale experimentation. What errors can arise? How do you deal with these errors?
Observed Convergence Latency

- Less than half of Tdown events converge within two minutes
- Long tailed distribution (up to 15 minutes)
Impact on Traffic

([Labovitz00a] figure 4a)

Why does loss go up?

⇒ Because there are route loops in the net causing packet drops.
How To Tell What’s Going On?

- Simulate BGP
  - model one router per AS
  - assume full routing mesh
  - ignore latency
  - synchronous processing via global queue

⇒ simple model that captures key details
Why does this happen?

- In RIP’s counting-to-infinity, routes of increasing metric values explored
- In BGP, the theoretically worst case occurs when all possible alternate paths are explored
  - $O(n!)$ such paths
  - Explains pathological convergence time
Impact of Timers

• BGP has minimum route advertisement timers
  – MinRouteAdver impacts the Tup/Tshort times
  – MinRouteAdver can also reduce the number of paths explored
    • At most one path for each path length
    • But it introduces consequent hold down delays
Does this explain measurements?

- Tup/Tshort converge quickly because they shorten path length and therefore are quickly accepted
- Tdown/Tlong converge slowly because BGP tries hard to find all alternatives
Other Observations

• Could do loop detection at *sender* side and not just receiver side