BGP SPECIFICATIONS AND BASIC IMPLEMENTATION

Tawfiq Khan
TCOM 610
George Mason University

Routing Protocol Basics

- Routing Overview
- Distance Vector vs Link State
- BGP Overview
- BGP Specification (RFC4271)
- BGP Implementations
  - Cisco and Juniper syntax

Routing Overview

- Data network: control plane and data plane
- Routing protocol: control plane -- data are used to control routers to deliver data packet to its destination
- Routers speak routing protocols between them
- In-band control: data plane and control plane are mixed (SS7: separate control network)
- Forwarding: select an output port based on destination address and transmit the packet

Routing Algorithm

- Static vs dynamic
- Flat vs Hierarchical
- Intra-domain vs inter-domain
- Link state vs distance vector
- Host intelligence vs router intelligence

- BGP: modified distance vector-based dynamic inter-domain routing protocol (path vector); used to exchange network reachability info across different autonomous systems (AS)

Internet and BGP

- Internet consist of many networks inter-connected together
- Each network runs its own interior gateway protocols (IGP) such as OSPF, ISIS, RIP, which controls the routing and exchange reachability info WITHIN the network
- The reachability info among different networks are exchanged over BGP4, usually in an aggregated way
- Autonomous System (AS): identified by unique ASN, typically means an administrative domain with the consistent routing policy and single IGP domain
  - Similar to a country code in Postal System
  - Uniquely identified a network

IGP - Interior Gateway Protocol

- Interior Gateway Protocol
- Used to exchange reachability info within an autonomous system (under single network administration)
- Two flavors: distance vector or link state
- Distance vector protocol: periodic update of whole routing table between neighbors, potential routing looping, slower convergence, improved by composite metrics and triggered updates, “routed by rumor”
- Link State protocol: periodic flooding of directly connected networks to every node; reliably flood link state to every node, better convergence and metrics; hierarchical deployment, use Dijkstra algorithm to get shortest path; “routed by map”
- Both are distributed: each node calculates its own routing table: Dijkstra algorithm
- Popular IGP: OSPF, ISIS, RIP, EIGRP

Comparison

Distance-Vector Routing

- Each router sends routing table to its neighbors
- The information sent is an estimate of its path cost to all networks
- Information is sent on a regular periodic basis
- A router determines next-hop information by using the distributed Bellman-Ford algorithm on the received estimated path costs

Link-State Routing

- Each router sends directly connected routes to all other routers
- The information sent is the exact value of its link cost to adjacent networks
- Information is sent when changes occur
- A router first builds up a description of the topology of the network and then may use any routing algorithm to determine next–hop information

Internet Routing Types

- Static Route: statically point a network to an interface
- Default Route: 0.0.0.0/0, also referred to as last resort, match everything except those specific in the routing table (typically IGP or static route)
- Dynamic Route: routes learnt by routing protocols such as BGP4
- Stub AS: AS connected to only one upstream ISP
- Multi-homed Non-transit AS: AS connected to multiple ISPs
- Multi-homed Transit AS: not only connected to multiple ISPs, also announced full Internet route to its customers (traffic's src/dest can be outside of the network)

BGP - Path Vector

- Each route has entire path attached
  - Distance vector attaches only “distance”
- No count to infinity problems
  - Loops detected by walking the path vector
- Next hop selection
  - Path length can be *one* of criteria used to select next hop
  - Path length usually not first criteria used, instead used as a tie-breaker when other parameters (and there may be many) are the same
- BGP uses path vector algorithm for loop detection
- For destination-based routing only

BGP Basic

- Used to exchange routing information, including networks and attributes between participating routers
- Routers running BGP are known as BGP speakers
- Uses TCP as transport protocol
  - Reduces complexity of BGP itself
  - Uses port 179

BGP Property

- BGP is used to construct graph of autonomous systems (ASN)
- Path information associated with destination networks ASN for loop free routing
- BGP speakers with sessions are known as peers or neighbors
- Open messages are used to establish peering relationship
- Mechanisms to end connections (admin down, tcp session tear-down)
- Application level periodic keepalive update to maintain neighbor liveness
- Table version is used to monitor the change frequency of the routing table (instability)

BGP Operations

- Routes are exchanged once peering relationship established
- Initially all routes in BGP table are exchanged, then updates are incremental
  - TCP is stateful; only send routes it uses (best routes)
- Routing information includes prefix, attributes, and AS path
- If no routing changes occur, only keepalive messages are exchanged
- Session is closed upon one side terminates TCP connection or error conditions occur

BGP Messages

- Open – to open peering session
- Update – to advertise/withdraw prefixes
- Notification – to notify error conditions
- Keepalive – periodic hello for liveness (heartbeat)
- Specified in RFC4271 (obsolete RFC 1771)

BGP Message Header

- Marker (4x4 = 16 octets)
  - all ones for compatibility
  - Used for authentication before (but now obsolete)
- Total length (2 octets), including header
- Type code (1 octet)
  - 1 – OPEN
  - 2 – UPDATE
  - 3 – NOTIFICATION
  - 4 – KEEPALIVE
  - 5 – Route Refresh (RFC 2918)
- Minimum size header is 19 octets
- Always BGP message header plus different message info

BGP Open Message

- BGP header plus following fields
  - Version (1 octet) – always 4
  - ASnum (2 octets) – 32 bits for future
  - Hold time (2 octets) – Seconds
  - BGP ID (4 octets) – IP address
  - Optional Parameter length (1 octet)
  - Optional Parameters (variable)
  - Format: (ParamType, ParamLength, ParamValue)
- Minimal Message size is 10 bytes
- Minimum, 29 bytes including header (19 + 10)

Optional Parameters

- Format of TLV triplets
- Parameter Type: 1 octets
- Parameter Length: 1 octets
- Parameter Value: variable
- Usually used for capacity parameters in RFC3392
  - Type Code 1 for authentication is expired
- TLV (type, length, value) design makes BGP easy to extend, but harder to implement

Update Message

- Withdrawn Routes Length (2 octets)
- Withdrawn Routes (variable)
- Total Path Attribute Length (2 octets)
- Path Attributes (variable)
- Network Layer Reachability Information (variable)
- Could be used to advertise multiple prefixes with same path attributes and withdraw multiple prefixes
- Min: 23 Octets (19 + 2 + 2), Max: 4096

Withdraw routes

- Prefixes that are no longer available
- Represented as <length, prefix> pairs
  - Length is 1 octet (length in bits)
  - Prefix is minimum number of needed octet, filled up with 0s to make up to octet boundary
- Single update message can remove multiple prefixes

Path Attributes

- Each attribute is (type, length, value) – TLV again
- Attribute type (2 octets) = (attribute flags, attribute code)
- Attribute Flags (1 octet)
  - Optional
  - Transitive
  - Partial/Complete
  - Extended (one octets or two octets)
- Attribute Type Code (1 octet)
- Length (1 or 2 octets depending on flag value – extended or not)
- Value (variable)

Common Attributes

- Origin (Type Code = 1) (i, e, ?)
- AS_Path (Type Code = 2) (AS-SET, AS-SEQ)
- Next Hop (Type Code = 3)
- Multi Exit Discriminator (MED) (Type Code = 4)
- Local_Pref (Type Code = 5)
- Atomic aggregate (Type Code = 6) (flag for aggregation without AS_Set) (Length=0)
- Aggregator (Type Code = 7) (ASN and BGP identifier)

NRLI

- Network Layer Reachability Information
  - Represented as <length, prefix> pairs
  - Length in bits
  - Length zero used for 0.0.0.0/0
  - Multiple prefixes with the same path attributes for a single announcement – to improve efficiency
- NLRI announcements used to direct traffic
  - NLRI along with path information indicates that a particular AS route can be used for that network

Keepalive

- The same as open message without any message body (19 bytes)
- Protocol level hello, not transport level keepalive
- Keepalive is exchanged periodically if no update is sent
- Typical 1/3 of the hold time interval
- If holdtime = 0, then no Keepalive message is sent
- Used a timer to trigger the sending of keepalive

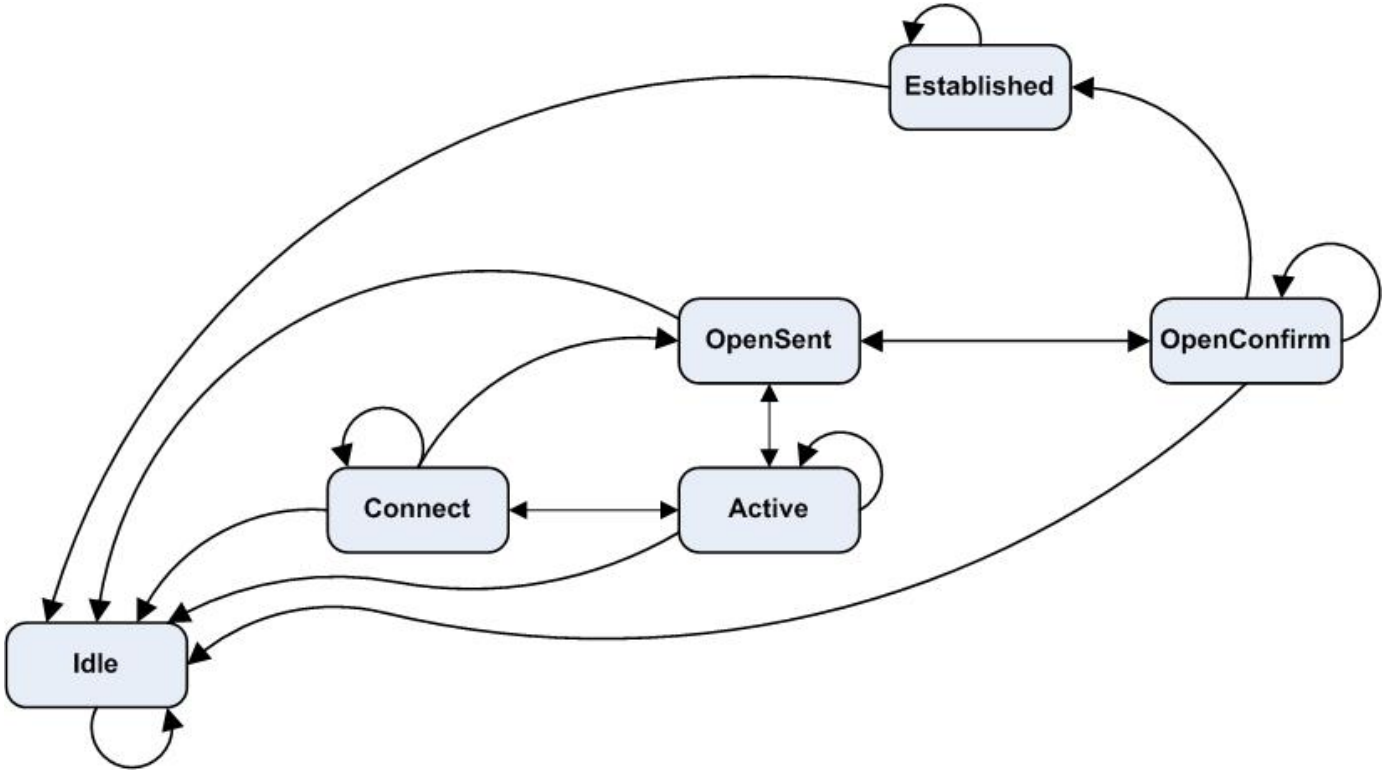
BGP Notification

- Format: (Error code, error subcode, error data)
- Error code (1 octet)
  - 1 header error
  - 2 open error
  - 3 update error
  - 4 holdtime expired
  - 5 FSM error
  - 6 cease
- Error subcode (1 octet)
  - Connection not sync, bad message length, bad message type
- Data – variable octets, depending error code

BGP States

- Idle
- Connect
- Active
- OpenSent
- OpenConfirm
- Established
- Active vs Passive connection

BGP FSM



Idle State

- State where you are doing nothing and refuse incoming connections, no resource allocated
- Manual or Auto start event gets you out of this state into Active state by listening for connections
- Initiate TCP connections and listen for remote connections and change state to Connect
- Most error bring back to Idle state

- Start events should not be generated immediately for a peer that was previously transitioned to Idle due to an error – use exponential backoff (initial: 60 secs)

Active State

- Trying to acquire a peer by listening for and accepting TCP connection
- If connection succeeds, an Open message is sent and the FSM transitions to the OpenSent state (clear ConnectRetry timer, and start HoldTimer)
- If Connection fails, move back to Idle state
- Connect retry timer expired causes the system to send another SYN packet

Connect State

- Waiting for transport protocol to be completed
- If succeed, clear ConnectRetry timer and send OPEN message and transition into OpenSent
- If failed, start ConnectRetry timer, and transition into Active or Idle State

OpenSent State

- Wait for an Open message from peer
- When received
  - Negotiate version
  - Check AS
  - Negotiate hold down timer (lowest value used)
  - Check for connection collision (look through open confirm table)
- If header error or collision, send NOTIFICATION and goes into IDLE
- Else send KEEPALIVE and transition into OpenConfirm

OpenConfirm State

- Waiting for keepalive that acknowledges the Open message
- If keepalive received, transition into Established
- If Keepalive timer expires, send another keepalive
- If Holdtimer expires, send NOTIFICATION, and transition into IDLE

Established State

- State in which updates and keepalives are exchanged
- Keepalive Timer to trigger sending Keepalive message to peer
- Hold Timer kept to notice absence of Keepalive or Update message coming from peer
- If any error or hold timer expired, send NOTIFICATION and transition into IDLE

Typical Scenario

- Idle=>Active=>OpenSent=>OpenConfirm=>Established
- Idle=>Connect=>OpenSent=>OpenConfirm=>Established

Attribute Classification

- Well-known mandatory
- Well-known discretionary
- Optional transitive
- Optional non-transitive

Attribute Encoding

- First bit indicates optional (1) or well-known (0)
- Second indicates transitive (1) or non-transitive (0)
- Third bit indicates partial (1) or complete (0) optional transitive attribute
- Fourth bit indicates length, 1 octet (0) or 2 octets (1)
- Lower order bits unused and always zero

Well-know Attributes

- Must be recognized by all BGP speakers
- Mandatory attributes must be included in all update messages with non-zero NLRI
- Can be updated, but must be passed along to other BGP speakers

Optional Attributes

- Do not need to be recognized by a BGP speaker
- If a speaker doesn't recognize an optional transitive attribute, it passes it along, un-modified, setting the partial bit
- Unrecognized non-transitive optional attributes can be quietly discarded
- Optional attributes attached by something other than the originator need to have the partial bit set

Origin

- Type Code: 1
- Well-known, mandatory attribute
- Indicates source of NLRI Values
  - 0 IGP (NLRI inside originating AS)
  - 1 EGP (NRLI learned via EBGP)
  - 2 Incomplete (NLRI from some other means)

AS_PATH

- Type Code: 2
- Well known, mandatory attribute
- Format: <path segment type, length, value>
- Types:
  - 1 AS_SET
  - 2 AS_SEQUENCE
- Value encoded as a list of AS numbers, each ASN 2 octets long
- IBGP speaker never modify AS_PATH
- EBGP speakers must prepend own AS

NextHop

- Type Code: 3
- Well-known mandatory attribute
- IP address of the border router that should be used as the next hop
- For EBGP speakers, this router must be on a subnet common to both speakers
- IBGP speakers should not modify Next_Hop

Multi-Exit Discriminator

- Type Code: 4
- Optional non-transitive attribute
- Also know as the MED
- Hint to external neighbors about the preferred path into an AS that has multiple entry points
- Never passed via EBGP beyond direct peers
- Lower is better

LOCAL_PREF

- Type Code: 5
- Well-known discretionary attribute
- Indicates degree of preference for a route as compared to other routes to same destination within an AS
- Included in message to IBGP speakers only
- Higher means more preferred

Aggregator/Atomic_Aggregate

- Type Code: 7
- Optional transitive attribute
- Length: 6 octets: AS number (2 octets) and IP address of router (4 octets) that has generated an aggregate
- Example: 1668:66.185.128.29

Other Attributes

- Community (8)
  - Group of destinations sharing a common property; provides flexible grouping
- Originator ID (9)
  - Identifies routes originator in local AS, and used for loop avoidance in IBGP
- Cluster List (10)
  - Sequence of Cluster Ids that the route has passed; used in RouteReflector setting; also used for loop avoidance for IBGP
- http://www.iana.org/assignments/bgp-parameters for all defined attributes

BGP Extension

- BGP capability negotiation: Route-refresh, MP-BGP, ORB, graceful-restart
- Multiple Protocol BGP extension: use BGP to carry BGP-VPN, MPLS label and IP v6 routing info
- BGP security: soBGP, sBGP and MD5 Signature
- Covered later in this course

Basic Cisco IOS Configuration

```
interface POS6/0
  description pop6/0: customer XYZ (66.185.133.224/30)
  ip address 66.185.133.225 255.255.255.252
  crc 32
  pos scramble-atm
router bgp 1000
  neighbor 66.185.133.226 remote-as 100
  neighbor 66.185.133.226 send-community
  neighbor 66.185.133.226 soft-reconfiguration inbound
  neighbor 66.185.133.226 prefix-list FILTER-OUT out
  neighbor 66.185.133.226 route-map TO-XYZ out
  neighbor 66.185.133.226 route-map FR-XYZ in
  neighbor 66.185.133.226 description XYZ
  neighbor 66.185.133.226 password 7 047A3F2221737E7C
```

Show commands

Show ip bgp sum
Show ip bgp neighbor x.x.x.x
Show ip bgp neighbor x.x.x.x received-routes
Show ip bgp neighbor x.x.x.x advertised-routes
Show ip bgp 125.64.20.7
Show ip bgp

```
pop1-alb#sh ip bgp sum
Neighbor          V    AS MsgRcvd MsgSent
66.185.133.226 4  100   332942  5485879
TblVer    InQ OutQ Up/Down  State/PfxRcd
35034657    0    0   16w3d         217
```

EBGP

- EBGP: two BGP speakers, each in different AS domain
- Typically require direct IP connectivity to establish session (except for multi-hop eBGP)
- Peer IP is typically the interface IP address of P2P link – the session's stability is relied on the stability of physical interface
- Append its own AS in the AS_PATH
- BGP next-hop is typically the other side of the directly connected interface

IBGP

- IBGP: required to pass externally learnt routes to other routers in the same domain
- Also required IP connectivity – but need not to be directly connected, IGP can provide intra-domain IP connectivity
- IBGP speaker don’t re-advertise IBGP learnt routes to other IBGP peers (except for RR) to prevent route loop (no AS_PATH checking within same ASN)
- This requires full IBGP mesh among all BGP speakers in the same AS
- Next-hop: IGP should provide remote next-hop reachability

BGP Continuity inside AS

- BGP: reply on AS_PATH for loop detection
- IBGP: not loop detection mechanism; by default, router CAN'T re-advertise IBGP learnt route to other routers
- How can you pass eBGP learnt routes to other routers inside an AS?
  - Full IBGP meshing inside an AS or
  - IBGP meshing among border routers and redistribute BGP routes into IGP, and let IGP propagate BGP routes inside an AS
  - In this case, we will have synchronization problem

BGP Synchronization

- BGP must be synchronized to IGP before advertising transit routes to external AS
  - wait until IGP has propagated routes across AS before advertising them to other external ASs
  - might otherwise receive traffic it cannot route yet
- Internal reachability verified before EBGP advertisement for routes received via IBGP
  - checks for existence of route in the IGP
  - Check for reachability of next-hop
  - if not, no EBGP advertisement

BGP Synchronization

- Injecting BGP routes into IGP is costly
  - redistributing leads to heavy load on internal routers
  - particularly troublesome if internal routers are less well equipped
- External route injection not really necessary
  - can use default (last resort) route
- Synchronization can be disabled
  - allows advertisement of routes learned via IBGP regardless of existence of IGP routes
  - acceptable when all transit routers inside AS are running fully meshed IBGP
  - acceptable when AS is not a transit AS

Route Origination

- Without route origination, there will be no route advertisement even if BGP session is established
- Dynamic vs Static origination
- Dynamic Redistribution into BGP
  - Easy and fully dynamic – adopt to the change of network
  - Potential scalability and instability issues
  - Avoid mutual redistribution
  - Better to use route filter to control redistribution
- Static Redistribution into BGP
  - Use static route
  - Only inject specific aggregate into BGP
  - Always there – may cause problem

Dynamic Route Injection

- Types
  - purely dynamic (redistribute command)
  - Semi-dynamic (network command)
- Purely dynamic
  - redistribute commands injects all IGP routes into BGP
  - May control by using prefix list or additional commands in redistribute command
- Advantage
  - ease of configuration
- Disadvantage
  - can inject unwanted or faulty information
  - IGP instability affects BGP updates: route flapping

Semi-Dynamic Route Injection

- Semi-dynamic
  - network specifies subset of networks to be injected into BGP
- Route verification
  - verify the networks exist by checking in the IP routing table
  - prevents injection of misleading routes
  - Verify the existing of an exact matched route in routing table
- Advantage
  - fine grained control
- Disadvantages
  - administrative complexity
  - router implementations may put a limit on the number of prefixes that can be listed

Route Redistribution Problem

- Redistribution provides opportunity for injection of problem routes
- Typical problems
  - private addresses (RFC1918)
  - illegal (not registered) addresses
  - routes not complying with policy (prefix too long)
- Filtering can be used to avoid problems
  - Deny unwanted routes
  - Prevent mutual redistribution: information injected into IGP from EBGP then sent back into EBGP at another point
  - Use route tagging or OSPF type filter to control redistribution

Controlling Route Instability

- Through redistribution, IGP instability will induce BGP update/instability
- Factors
  - unstable access links
  - faulty hardware
- Aggregation
  - fluctuation of single route does not cause fluctuation of aggregate
  - could be performed at customer or provider boundary
- Decouple route advertisement from route existence
  - static injection of routes

Static Route Injection

- Manually define IGP or aggregate routes
  - always in IP routing table, always advertised
- Improves stability
- If route is advertised at only one point, static routes have no drawbacks
- If route advertised at multiple points, traffic may black-hole in case of failure
  - problems inside the AS might prevent a particular border router from reaching destination that might otherwise be reachable

BGP Origin

- Networks advertised with network or via aggregation have ORIGIN as internal to the AS (origin code=i)
- Networks advertised with redistribute from IGP (OSPF, RIP, ISIS et) have ORIGIN set to ?
- EGP routes: origin code=e – only for routes learnt from EBGP protocol
- Used in BGP decision process to favor one route over another (i< e < ?)
- Better approach: network or redistribute using static with route-map

Route Injection Example

- OSPF into BGP
  - Redistribute ospf 1 match internal external [1|2] nssa-external [1|2]
- Network Statement
  - router bgp 1
  - network 192.213.0.0 mask 255.255.0.0
  - ip route 192.213.0.0 255.255.0.0 null 0
- Static Route
  - router bgp 200
  - neighbor 1.1.1.1 remote-as 300
  - redistribute static
  - ...
  - ip route 175.220.0.0 255.255.255.0 null0

BGP backdoor

- Different IGPs and EGPs interacting
- Networks might be learned via multiple protocols
- Choice affects traffic patterns
  - RIP versus OSPF versus BGP make wildly different decisions
- Backdoor links and routes
  - Backdoor links provide alternate IGP path that can be used instead of an EBGP path
  - Backdoor routes are IGP routes over the backdoor links
- Need a mechanism for choosing between routing protocols in forming IP routing table

Protocol Admin Distance

- Directly connected 0
- Static 1
- EBGP 20
- EIGRP (internal) 90
- IGRP 100
- OSPF 110
- ISIS 115
- RIP 120
- EGP 140
- EIGRP (external) 170
- IBGP 200
- BGP local 200
- Unknown 255
- Lower is Better
- Juniper: protocol preference and more fine-granular

Protocol Interaction

- Can force IGP route precedence over BGP routes
- Tag BGP routes as backdoor routes
  - classifies routes as “BGP local” with large distance
  - Cisco command: **network** *address* **backdoor** (under router bgp configuration)