An Introduction to the Border Gateway Protocol (BGP) for Protocol Developers

Jeffrey Haas (jhaas@juniper.net)

John Scudder (jgs@juniper.net)

What this talk is not

- How to configure routing on your favorite vendor's implementation.
- Configuration best practices.
- A tutorial on BGP policy.
- An introduction to internet routing.
- A discussion of security issues around BGP deployment.

What this talk is about

- The high-level protocol elements and procedures in BGP and its extensions.
- discussion about how the extension mechanism has permitted the protocol to grow over time.
- some of the things that have gone well and some that haven’t.

RFC 4271 Basics

BGP State

What is the border gateway protocol (BGP-4)?

- Path (distance) vector protocol.
- Specified in RFC 4271.
- The main routing protocol for the internet.
- It permits “autonomous systems” (ASes) to exchange their routes.
- Over time, it grew from its original IPv4 use case to support IPv6 and carry virtual private networks (VPNs).
- It now supports other use cases such as carrying link-state (BGP-LS, LSVR) and is continuing to evolve to carry other state.
- If you want a history of the protocol, consider reviewing https://www.youtube.com/watch?v=iPUBwXk4iEk

What's an Autonomous System? [jgs]

- Usually called an "AS" (pronounced ay ess)
- Identified by an "AS Number" or "ASN" (used in the AS_PATH and elsewhere)
- Contiguous set of BGP routers under the same administrative control
  - The basic unit of abstraction used to scale BGP globally
- Within an AS, Internal BGP ("iBGP") is used
  - Generally, policy is not applied on IBGP sessions, the goal is for the whole AS to make consistent routing decisions
- ASes talk to one another using External BGP ("eBGP")
  - Policy (filtering and manipulation of routes) is the norm
- We will return to all these points…

Routing state

- BGP routes pair sets of destinations, called “Network Layer Reachability Information (NLRI)” with a set of properties called “Path Attributes” shared by those destinations.
- BGP is a “stateful” protocol. Once you’re told a route, you’re expected to hold on to it, if you want it.
  - If (due to a bug) you lose something you were supposed to have, or keep something you were supposed to remove, the protocol is not good at helping you recover.
- At a high level, BGP is a “key-value” protocol, which is a property that makes it attractive for many use cases.

Carrying Routing State in Update Messages

- BGP routes are carried in “Update” messages (Protocol data units, or PDUs).
- Updates carry positive state (new/changed routes) or negative state (withdrawn routes).
  - Changed state is called an “implicit withdrawal” of the previous state.
- Updates permit more than one NLRI to be advertised or withdrawn in an update for packing efficiency. This reduces CPU work and network traffic.
  - A significant amount of work on extensions is to try to maintain packing.

Where are Routes Stored

- RFC 4271 talks about storing BGP routes in the “Routing Information Base(s)” - the RIBs.
- The RIBs are keyed on the NLRI.
- There are three logical views
  (there is no requirement for the implementation to do it exactly this way!):
  - Adj-RIBs-In: Routes the speaker receives from another BGP speaker.
  - Loc-RIB: The best route from the routes it has received from its neighbor and calculated as part of the “decision process”. (Route selection)
  - Adj-RIBs-Out: The set of routes from the Loc-RIB advertised to its neighbors.

RFC 4271 Basics

BGP Sessions and Messaging

BGP Routing Protocol Core Requirements

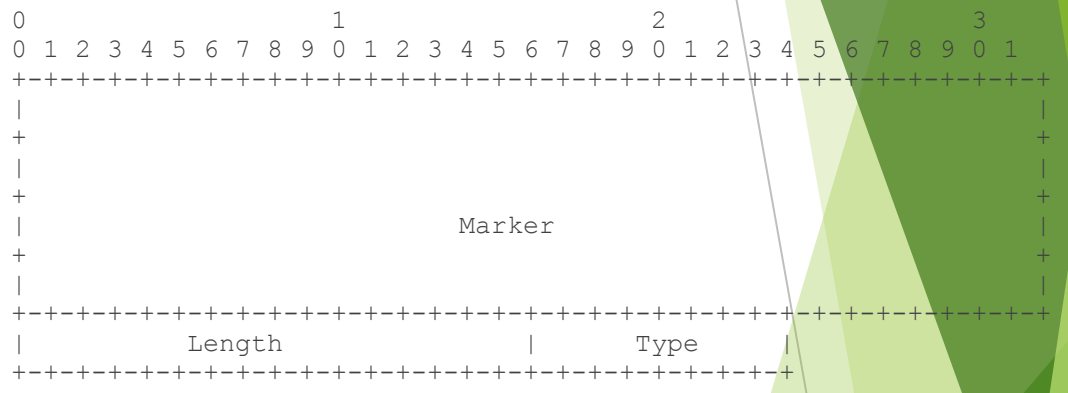
- BGP is connection oriented.
- BGP is configured to know what its connection endpoints are.
- BGP uses TCP as its transport protocol.
- BGP exchanges messages (PDUs) over TCP to establish a new session, verify the liveness of the connection, and exchange routes.

Carrying Messages Between Two BGP Speakers

- BGP is carried in TCP on port 179.
- good things about TCP :
  - TCP provides stream-oriented, reliable, in-order delivery.
  - BGP messages may be “large” - up to 4k bytes, or even larger with modern extensions!
  - TCP means that BGP doesn’t have to work on sequencing, reliability and retransmission, or packet size limitations - it delegates that work.
- bad things about TCP :
  - Slowness, issues related to packet drops, window congestion, and TCP security impact BGP.
  - TCP doesn’t provide framing services for BGP. BGP is responsible for doing that work on its own.
  - Every BGP developer eventually must develop some TCP expertise.

BGP Messages

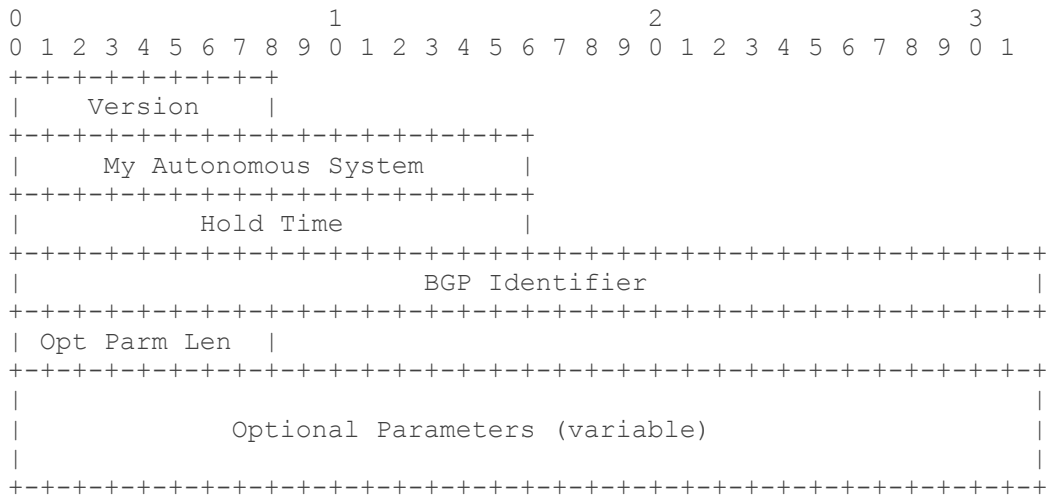
- BGP messages all have a common message header format.
- The marker is vestigial! it was originally meant for security, but now only carries sixteen octets of 0xff.
- Over the life of BGP, there's currently only 5 message types!



Initial Protocol Handshake

- When a pair of BGP speakers establish a connection over TCP port 179, they exchange parameters for the session using “Open” messages.
- The minimum information sent by each side in an open message is:
  - A version number that has been 4 for most of BGP’s life!
  - The autonomous system number (ASes).
  - A hold time as a bid for how long to keep the session established if you stop receiving messages from the other speaker.
  - A “BGP Identifier” (router-id).
  - “Optional Parameters” where most of the interesting stuff has gone over the years.

BGP Open Message



Initial Protocol Handshake (2)

- Each BGP speaker looks at the other's open message to decide if they find the parameters "acceptable". If so, they set the "hold timer" as the minimum bid from both systems and the session becomes "established".
- A large amount of finite state machine detail has been glossed over! Much of of the finite state machine (FSM) in RFC 4271 deals with all sorts of corner cases, including dealing with more than one connection between the same speakers - a collision!
- Getting this stuff right is hard, but not the interesting part of the protocol.

Determining if the Other BGP Speaker is Still Alive

- TCP's timers are often very long. For a routing protocol that is concerned about the "liveness" of reachability, you need to know if the other side of the connection is still there on a timely basis.
- A BGP speaker expects to get a message from the other speaker at least once every hold timer interval.
  - Update messages are good enough!
  - If there's nothing else interesting to send, a "Keepalive" message is sent. this is just an empty message header with the correct message type value.

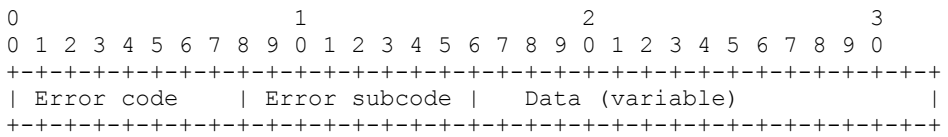
Terminating the Connection

- When ending the BGP connection, whether during initial handshake, or because of error conditions, a “Notification” message is sent.
- The notification contains an ”error code”, an “error subcode”, and data that is specific to that combination of values.
- No matter what the content is, the TCP connection is terminated after transmitting the notification!

Notification Message

While most of the information sent in notification messages is diagnostic information, some of the code/subcode pairs will trigger behaviors in the protocol.

For example, version negotiation, graceful restart, etc.



Update Messages

- Most of the interesting stuff is carried in update messages. As said in previous slides, these carry BGP routes.
- Originally, RFC 4271 and its predecessors only carried IPv4.

RFC 4271 Basics

Exchanging Routes with Updates

Update Message

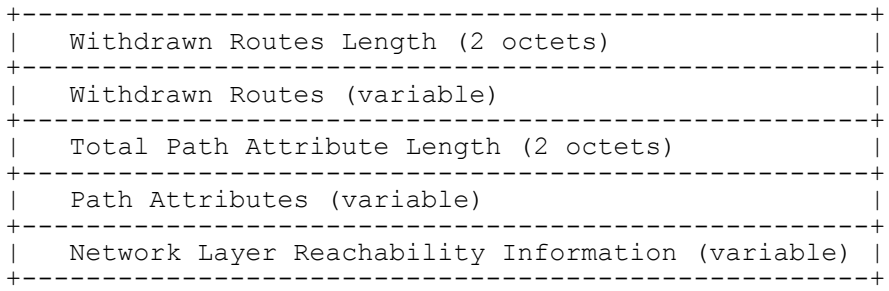
Withdrawn IPv4 routes (WD_NLRI) and new routes (NLRI) are carried in a common prefix format.

Path Attributes only apply to the new routes (NLRI); most of the interesting stuff is here!

Withdrawn routes and new routes can exist in the same message.

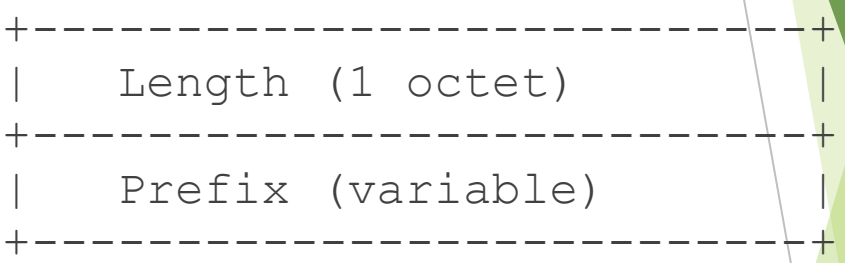
NLRI length is implied from message length in the common header - WD_NLRI length - Total Path attribute length!!!

> irk: Inconsistent encoding encourages implementation errors.



IPv4 prefixes

- The prefixes are encoded as a prefix length, followed by enough octets of prefix to cover the bits for that destination.
- They're variable length!
- You're supposed to ignore "trailing bits".
  - Since NLRI are the "keys", improper handling of trailing bits is a source of bugs.



A “Stream” of Prefixes

- NLRI and WD_NLRI, as seen in prior slides, are contained in the update message in an “envelope” that says how long the set of contained WD_NLRI or NLRI are.
- Each destination is variable length and you need to ignore “trailing bits”.
  - If you have two NLRI that are identical when ignoring their trailing bits, but not the same byte patterns, you have an ambiguous duplicate NLRI case!
- This means the only way to determine if the set of NLRI are properly encoded is to decode the stream member-by-member.
  - If the NLRI stream doesn’t terminate at the expected boundary, the packet is ”syntactically invalid” and your only choice is to terminate the connection!
- irk: NLRI counts would have gone a long way toward assisting developers detect NLRI encoding errors.

RFC 4271 basics

Path Attributes

Path Attributes - Origin

- RFC 4271 starts with only 7 path attributes:
- Origin - where did this destination enter BGP from?
  - From your IGP?
  - From something like static routes (Incomplete - we don't know!).
  - From the RFC 904 EGP protocol (very unlikely in 2024!)
- These days, Origin is a vestigial (but mandatory!) attribute, only useful as a kind of coarse metric. [jgs]

Path Attributes - AS_PATH

- A feature BGP is strongly known for is the AS_PATH.
- BGP uses this for loop detection in the protocol.
- Operators use its contents for policy!
- A vector of AS numbers read right for the originating AS to left as the most recent (usually, neighbor) as in the path.
  - BGP speakers at AS boundaries (external BGP /eBGP) prepend their ASN when sending routes to neighbors.
  - AS_SETS complicate the picture! (they are also … mostly … deprecated.)
- The length of the AS_PATH is used for picking the best BGP route (shorter is better):
  - Operators can prepend their AS multiple times to make the route less preferable.
  - Path length is a default metric, but there are many other factors (“decision process) and operators can configure policy.

Path Attributes - Next_Hop

- This is the address that should be used for forwarding toward destinations contained in the NLRI.
- The NEXT_HOP may be “first party” when it’s the address of the BGP speaker you learned this from.
- “Third party” next hops are permitted.
  - This could be a next hop on the same broadcast network as the BGP speaker you learned the route from.
  - More often, this is because the route came from an internal BGP (iBGP) session and the next hop isn’t changed. BGP is expected to “resolve” the immediate forwarding next hop from the received next hop using the “routing table”.

Path Attributes - MED and LOCAL_PREF

- The MULTI_EXIT_DISC(riminator) is a 32-bit metric exchanged between eBGP speakers.
  - It permits the remote AS to choose among multiple exits to the same neighbor AS.
  - Lower values are better!
  - There are gotchas with using MEDs, that we won't have time to cover today. (hint: MEDs break total ordering.)
- The LOCAL_PREF(erence) is set by an iBGP speaker when sending routes to another iBGP speaker to pick the preference of this route within the AS.
  - Higher values are better!

Path Attributes - aggregation

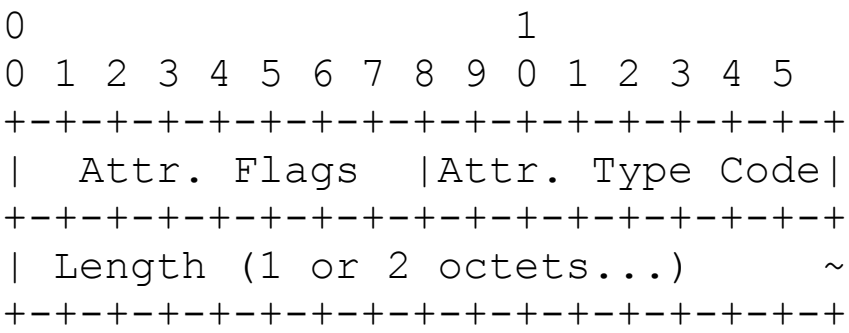
- RFC 4271 describes a procedure to create less specific “aggregate” routes from one or more specific contributing routes.
- When this is done, the AS_PATHs are aggregated and may create an “AS_SET”.
  - These things have become a problem for modern BGP security!
- The “Aggregator” attribute is attached to the route as information about what router (by BGP identifier) in what AS did the aggregation.
- The “atomic aggregate” attribute is mostly vestigial!
  - These days it indicates when the aggregate has discarded some information when building its aggregated AS_PATH.
- The advanced aggregation procedures described in the appendices of RFC 4271 are largely "aspirational" and not followed in practice.

Encoding Path Attributes - A Recipe for the Future

The path attributes are encoded as a vector of TLVs.

The flags are a nybble of flags marking the attribute as:

- Optional or mandatory.
- Transitive or non-transitive.
- “Partial”.
- Regular (1 octet) or extended (2 octet) length field.



Path Attribute Flags - Length

- The path attribute length field may be 1 or 2 octets in length based on this flag.
- In general, attributes should set the length field to what can minimally encode that length. However, it's not required that you do this.
- irk: A variable length length field is a place where errors occur in implementations.

Path Attribute Flags - Transitivity

- “Transitive” attributes are expected to be propagated by BGP speakers that receive this attribute, even if they don’t understand the attribute! (Think “tunneling”)
- “Non-transitive” attributes must be discarded by BGP speakers that don’t understand them.
- Non-transitivity provides scoping for features that must only be used in a domain of routers that all understand that attribute.
- Transitivity relates to *routers*, not AS boundaries!!!
- Most new path attributes end up being transitive.
  - That can lead to problems: https://datatracker.ietf.org/doc/draft-haas-idr-BGP-attribute-escape/

Path Attribute Flags - Optional

- Very few path attributes are “well-known” (the “optional” bit is not set).
  - These are all defined in RFC 4271.
  - They must all be transitive.
- BGP error handling procedures insist on attributes having specific flag bits!
- irk: This bit has mostly been a reason for bugs.

Extending BGP

Carrying New Attributes

Carrying New Path Attributes

- BGP has been successful because new features can be added to the protocol without having to do flag-day upgrades for all BGP speakers.
- As long as the path attribute is transitive, it can be carried via ignorant BGP speakers.
  - This implies that the feature needs to be deployed in situations where ignorance is okay!
  - An early example of this was BGP communities (RFC 1997)

Hidden Issues with New Features - Transitivity

- If a BGP speaker is ignorant of a Path Attribute and carries a malformed transitive attribute, it may be noticed several hops away from the speaker that originated the bad attribute.
  - The "blast radius" may be large because the route is likely to be propagated to many downstream BGP speakers.
- In original RFC 4271 procedures, when you detected such malformed attributes, the BGP connection would be reset. This penalized the adjacent router even if it wasn't the problematic party!
- These issues were responsible for RFC 7606 where correctable issues could be detected and the reachability withdrawn on error rather than dropping the connection.

Hidden Issues with New Features - the Decision Process

- For iBGP, all BGP speakers within an AS are expected to perform the same algorithm to pick BGP routes.
  - Details of the algorithm are interesting and important but won't be covered here.
- When there is inconsistent route selection, forwarding loops can happen!
- If new features try to change the decision process, they need to be designed to safely be deployed so that inconsistent route selection doesn’t happen.
  - This may involve a flag-day for an AS.
  - Scoping can be designed into such features to try to make deployment safer.

Extending BGP

Carrying IPv6 and Other Reachability

Carrying IPv6

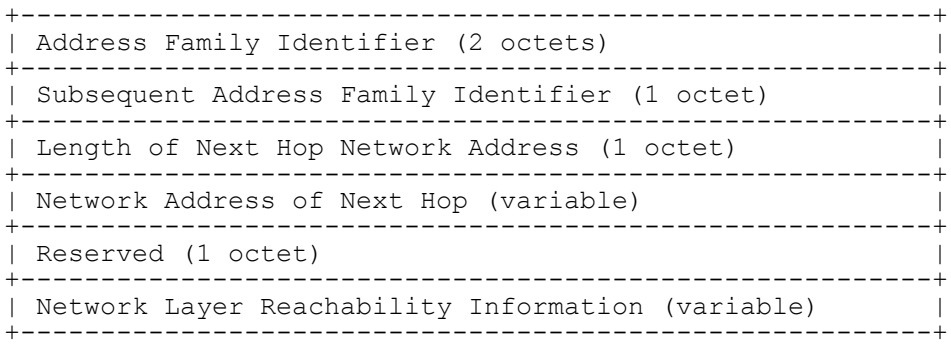
- RFC 4271 only can encode IPv4 prefixes. There's no room in the procedure for carrying IPv6!
- BGP could have been updated to a new version that carried multiple address families. However, that would have made for interoperability issues with older BGP-4 speakers.
- To keep backward compatibility, the new reachability is encoded in a path attribute.

Multi-Protocol BGP (RFC 4760)

The address family and a new “subsequent” address family are carried in the PDU.

- SAFI started originally to permit alternate topologies such as multicast. It quickly evolved to supporting VPNs.

The address family specific next hop goes in here (and the NEXT_HOP attribute goes away).



Multi-Protocol BGP (RFC 4760) (2)

- The new attribute is non-transitive. Thus, it doesn't propagate through ignorant speakers.
- How do a pair of BGP speakers decide what address families they're going to use? This is done using BGP "capabilities".
- irk: IPv4 unicast routes can now be contained in more than one place in the packet. This has led to bugs.

Extending BGP

Capabilities

Advertising Extensions

- Part of the work for doing IPv6, or any other address family or feature, is figuring out if a pair of BGP speakers will support that feature.
- To generally extend BGP procedures without a new version number for each feature to be added, BGP speakers must advertise support for a given “capability”.
  - Most of the time, both speakers must support a given capability for it to be used.

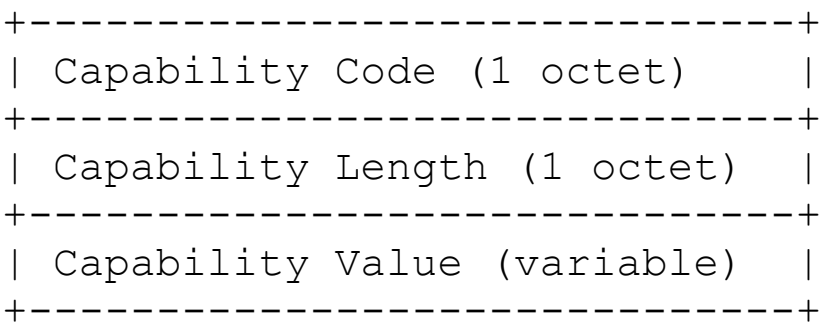
BGP capability advertisement (RFC 3992)

Capabilities are encoded in the BGP open message's "optional parameters" field.

They're TLVs.

irk: A given TLV code may occur more than once. this can create ambiguity in processing the capabilities.

[jgs] Remember Optional Parameters in the BGP OPEN? Well, all capabilities are encoded within just *one* optional parameter (called "Capabilities"). New Optional Parameters are never used. The reason for this won't be covered here.

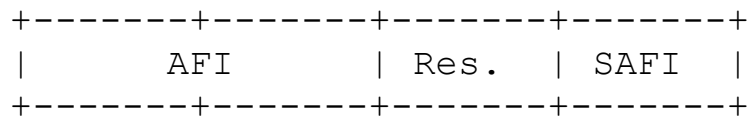


Capabilities for Multi-Protocol BGP (RFC 4760)

Vectors of multi-protocol AFI/SAFI sets are carried in a multi-protocol capability.

It's required that a given AFI/SAFI is exchanged bidirectionally for the session to support that AFI/SAFI.

This signals that a pair of speakers can understand a new NLRI format!



Scaling BGP

Confederations and Route Reflection

iBGP is Usually a Full ~~mess~~ Mesh

- The original expectation within an AS was that each BGP router peer with every other BGP router; i.e., a full mesh. This has several consequences:
  - When you add a new BGP speaker, you have to add it to every other BGP speaker. This is a provisioning overhead.
  - BGP connection scaling potentially becomes a problem in an implementation.
  - BGP route scaling, because of the additional connections, can become a problem.
- There was a desire to get out of the need for full mesh in deployments.

Required Properties to Get Rid of Full Mesh

- Per RFC 4271, routes learned via iBGP do not propagate to other iBGP routers. This avoided the need for an additional route loop prevention mechanism than the AS_PATH.
  - We want these routes to propagate within an AS. This means we need an additional loop detection mechanism!
- If we're changing how routes propagate, what changes are needed in BGP route selection?

BGP confederations (RFC 5065)

- The core idea is permitting an AS to be subdivided into internal “confederation ASes” that aren’t visible to the outside networks. They continue to see only the main AS.
- Confederation member ASes are established by configuration.
- BGP mostly looks like normal iBGP/eBGP, just with either outside ASes, or internal confederation member ASes.
- The AS_PATH gets a new “confederation” type for its sequence and sets that is used only within the confederation. These types are stripped at the confederation boundary.
- Route selection behaves as if everything within the confederation is normal iBGP!
- irk: Consistent configuration of the confederation member ASes and confederation as number is needed to deploy this feature.
- Confederations aren't widely used today.
  - This is a pity - many strange AS_PATH hacking can go away if they are used.

BGP Route Reflection (RFC 4456)

- Generally, more popular than confederations!
- Unless they are route reflectors, iBGP routers do not require additional configuration!
- The reflector will propagate routes from their clients back to clients - or to other reflectors, allowing for hierarchy.
- Because route reflectors perform route selection, they do information reduction ("path hiding")
  - This is good for scaling :-)
  - This is bad for robustness :-(
- Loop prevention is enabled via a new “cluster list” feature. The first reflector also inserts the received route’s BGP identifier for that client as the “ORIGINATOR_ID”.
- Route selection requires changes to use the ORIGINATOR_ID instead of BGP identifier for route selection. Cluster list length is also considered.

Extending BGP: Transparency through the Reflector

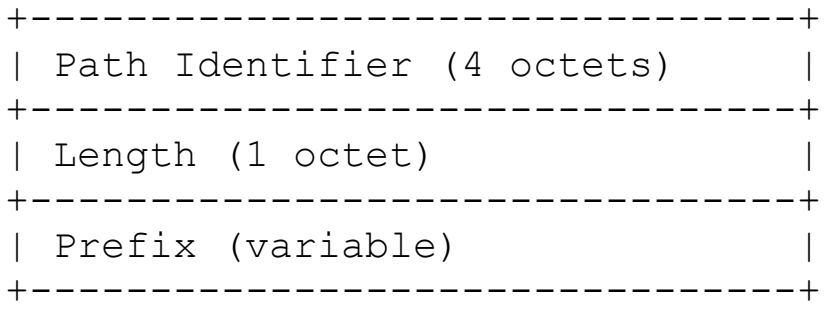
- A significant reason why path attributes even for iBGP-only features are transitive is that there's a desire to not need to update the route reflectors to support the new feature.
- Unfortunately, this also means that "attribute escape" can be a problem when internal-only features pass outside of the network that uses them and are later received by another network that may (improperly!) use them.
- Adding attributes is "easy". Adding new NLRI AFI/SAFI is... "not easy" (and won't be covered in detail).

“Unhiding” Paths: BGP add-paths (RFC 7911)

Sometimes the “path hiding” behavior of a route reflector isn’t desired.

In such circumstances negotiating the “add-paths” feature can permit a pair of BGP speakers to pass more than the best route between them.

This is done by adding an opaque path identifier to the NLRI.



Summary

- BGP started by just carrying IPv4.
- Starting with an extensibility mechanism permitted it to grow organically without requiring a new version.
- Some scoping was built into the extensions. This has done well but is tricky to always use safely.
- The flexibility to carry new types of reachability (keys) paired with new values (path attributes) means protocol designers are constantly looking to use BGP to carry new things, especially across cooperating networks.
- However, not every operator wants to receive every network's feature of the day.
- Protocol irks can lead to buggy code. This is bad enough when the issue is localized. However, the “blast radius” for bugs can be severe.