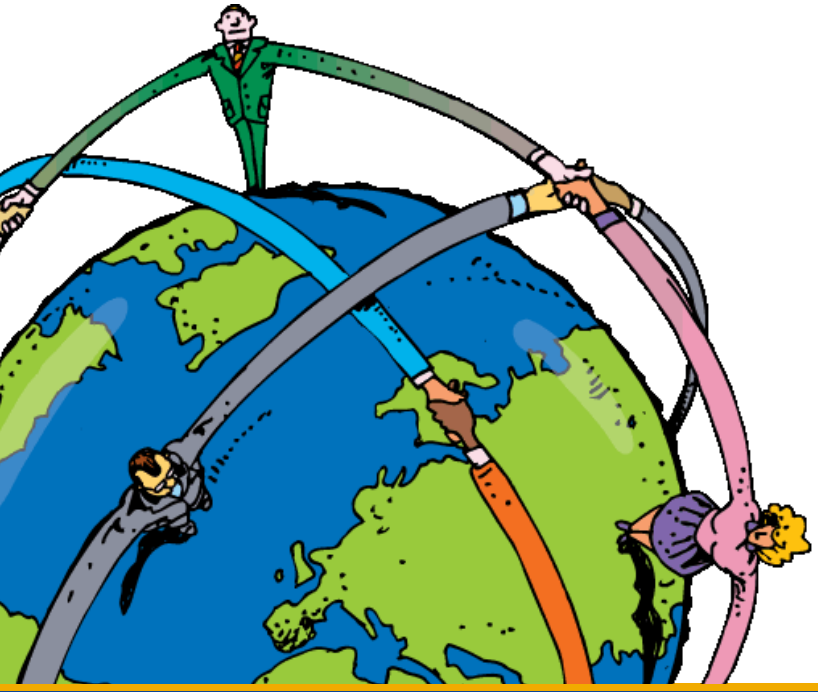


BGP at 18: Lessons in protocol design

Yakov Rekhter
yakov@juniper.net

Design by pragmatism

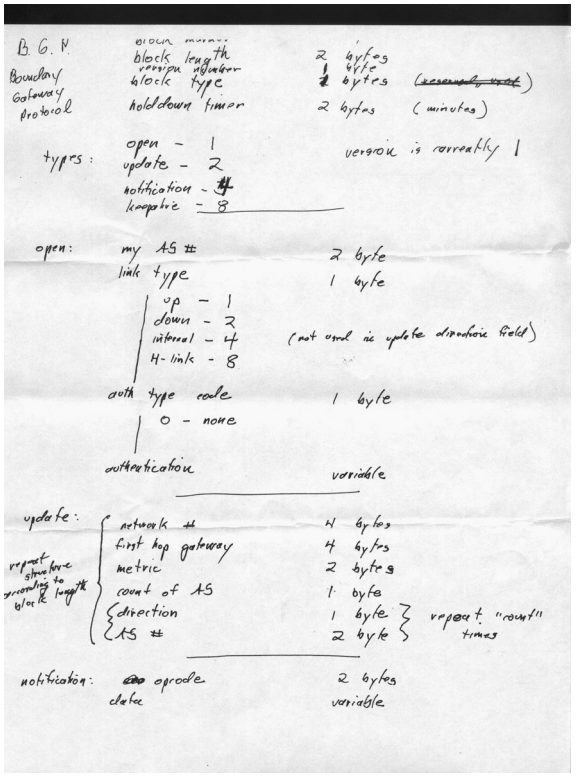
In the beginning…

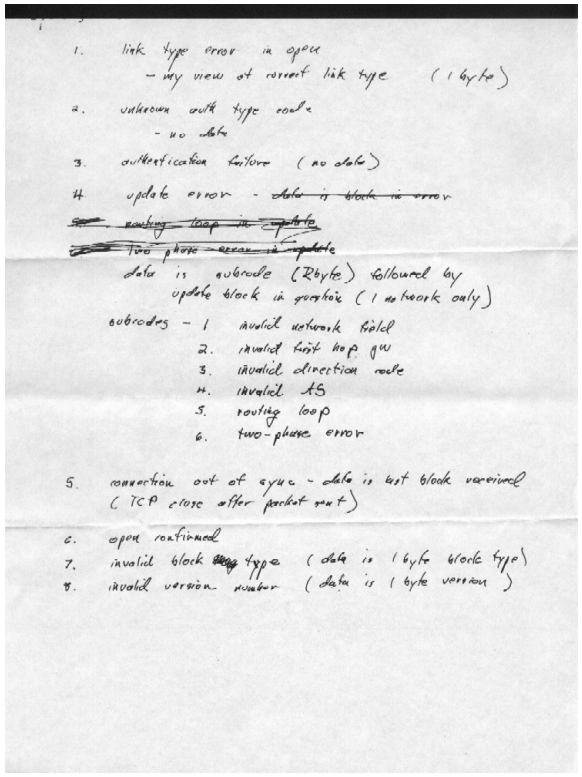
- **In 1987 NSF established a cooperative agreement with IBM/MCI/MERIT to build and operate NSFNET Backbone Phase II**
  - interconnect several regional networks and supercomputer centers
- **EGP-2 – inter-domain "routing" protocol**
  - constrains topology (at the autonomous system level) to a spanning tree
  - based on periodic refresh of complete reachability information
  - runs directly over IP

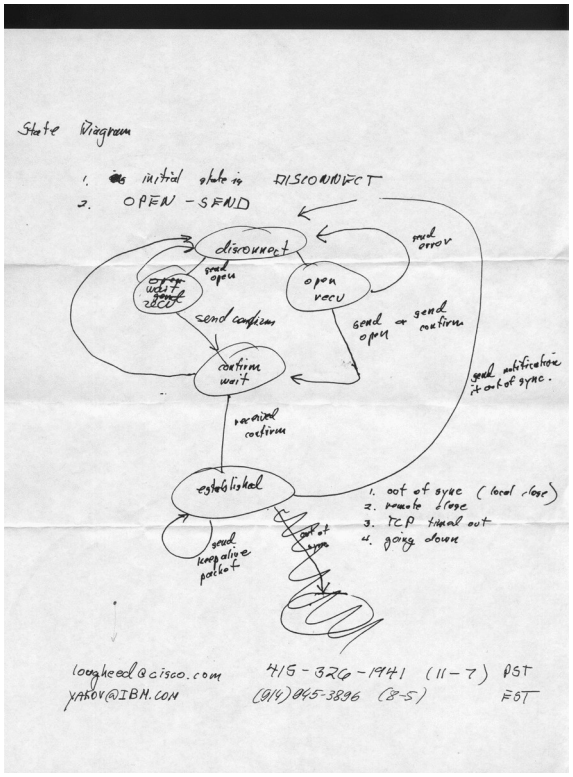
In the beginning…

- **January 1989, 12th IETF – TNP (“three napkins protocol”)**
  - Produced over lunch by K. Lougheed (Cisco) and Y. Rekhter (at that time with IBM Research), with the help of Len Bosack (at that time with Cisco)

In the beginning…







In the beginning…

- **Spring 1989 – two interoperable implementations**
  - Cisco (K. Lougheed)
  - NSFNET/IBM (Y. Rekhter)
- **June 1989 – RFC1105 "A Border Gateway Protocol (BGP)"**

In the beginning - design goals

- **Overcome limitations of EGP-2:**
  - eliminate restriction on inter-AS topology to be spanning tree (with ARPANET as a root)
  - eliminate problems caused by IP fragmentation of EGP-2 updates
    - loss of IP fragment would cause the loss of the whole update, several consecutive losses of EGP-2 updates would cause loss of reachability information (due to timeout)
- **Support few thousand classful IPv4 routes**
- **Replace EGP-2 in the NSFNET Backbone**
- **Was positioned as a short-term solution, to be (eventually) replaced by a long-term solution**

In the beginning - key ideas

- **Carry the information about the path traversed by the routing information (AS_PATH); use this information to suppress routing information looping**
  - Suppressing routing information looping provides (steady state) loop free packet forwarding
- **Use incremental updates (instead of periodic refresh)**
  - Requires reliable exchange of (incremental) updates
- **Use TCP as a reliable transport – avoid reinventing the wheel**
  - TCP was good enough

In the beginning – controversy

- **Did not support arbitrary routing policies**
  - supporting arbitrary routing policies had been left to the long-term solution
- **Using TCP for a routing protocol had been claims to violate certain design/architectural principles**
  - favor pragmatic/engineering considerations over violation of design/architectural principles

From BGP-1 to BGP-2 to BGP-3 (1989-1991)

- **BGP-2 (1990):**
  - Change encoding to facilitate support for future extensions by:
    - carry all the routing information (except for the reachability information) as attributes
    - define different types of attributes: mandatory vs optional attributes; transitive vs non-transitive attributes
    - encode each attribute as <type, length, value>
  - Eliminate useless features: link direction (up/down/horizontal)
  - Add new feature (marker) for authentication
    - Later on turned out to be fairly useless (use MD5 TCP authentication instead)

From BGP-1 to BGP-2 to BGP-3 (1989-1991)

- **BGP-3 (1991):**
  - Optimizes and simplifies the exchange of the information about previously reachable routes
  - Added mechanism to restrict a pair of BGP speakers to a single BGP session
    - "connection collision avoidance"
    - recently it turned out that in some cases having more than one BGP session between a pair of BGP speakers may be desirable
      - work is in progress to facilitate multiple BGP sessions between a pair of BGP speakers

BGP-4 (1992)

- **Support for classless Inter-Domain Routing (CIDR)**
  - significant scalability improvement by supporting reachability information aggregation/abstraction
- **Reachability information is encoded as a set of variable length prefixes**
  - replacing fixed length encoding in BGP-3
- **Wide deployment in the Internet around 1993**
- **Published as an RFC in 1995 (RFC1771)**
  - by the time of publishing RFC1771, BGP-4 (and CIDR) has been widely deployed in the Internet

BGP in mid-90s

- I-BGP mesh replacement
- Improved route filtering

Mid-90s: I-BGP mesh replacement

- **Bit of history (prior to mid-90s):**
  - to distribute information within an autonomous system all the BGP speakers within the autonomous system have to have a BGP session with each other
    - known as “full I-BGP (Internal BGP) mesh”
  - simple to implement
  - full I-BGP mesh was not a pressing practical problem in the beginning
    - but recognized as a potential problem early on
  - full I-BGP mesh became a practical problem in mid-90s
    - due to a (much more) widespread deployment of BGP, and growth of the Internet

Mid-90s: I-BGP mesh replacements

- **BGP Confederations**
  - “divide and conquer” – partition AS into several sub-ASs, and require full I-BGP mesh only within each sub-AS
- **BGP Route Reflection**
  - Replace full I-BGP mesh with hub-and-spoke
- **Did we need both at that time ?**
- **Do we still need both now ?**
  - yes, as both are deployed

Mid-90s: improving route filtering

- **Ability to constrain distribution of routing information (by route filtering) is one of the key requirements for BGP (as for any inter-domain routing)**
- **BGP Communities**
  - Provides a compact way of marking routes for the purpose of route filtering
  - Fairly general mechanism:
    - Partition community space into communities with global semantics (well-known communities), and communities with local semantics
    - Each Autonomous System could assign communities with local semantics on its own, yet such communities are guaranteed to be globally unique

Late-90s improvements

- Authentication
- Multi-protocol extensions
- Capability Advertisement

Late-90s: authentication

- **Bit of history (prior to late-90s):**
  - authentication by using the Marker field (introduced in BGP-2) turned out to be fairly useless for the purpose of authentication
    - under-specified (to say the least), and (therefore) unimplemented
    - trying to authenticate BGP without authenticating the underlying TCP is of (fairly) limited value
      - e.g., authenticating BGP by using the Marker field does not prevent Denial of Service attack caused by sending TCP RST
  - need to authenticate TCP connection

Late-90s: TCP MD5 authentication

- **Authenticate TCP connection by computing MD5 digest of each TCP segment, and carrying this digest as a (new) TCP option**
  - as a side effect, provides BGP session authentication as well
- **Implemented by many vendors**
- **Deployment is still (fairly) limited**
  - why ???

Late-90s: Multiprotocol extensions

- **Bit of history (prior to late-90s):**
  - BGP reachability information restricted to IPv4 address prefixes
  - What to do about inter-domain routing for IPv6?
  - How to support non-congruent unicast and multicast inter-domain topologies for IPv4 ?
  - One of the options was IDRP (ISO10747)
    - multiprotocol capable superset of BGP-4
  - Another option was to extend BGP-4

Late-90s: Multiprotocol BGP

- **Extend BGP, rather than implement (and deploy) IDRP**
  - Not as many features as IDRP, but (much) less complex to implement
    - whether all of the (additional) features of IDRP would ever be needed was not clear at that time
  - Solves what has to be solved… but in a fairly general fashion
    - carry reachability information as part of a (new) attribute; encode reachability information as <type, length, value>
    - use <Address Family Identifier, Subsequent Address Family Identifier> to identify the type of the reachability information (e.g., IPv4, IPv6, NSAP, etc…)
    - made BGP suitable for a wide variety of applications (see later)

Late-90s: Capability Advertisements

- **Bit of history (prior to late-90s):**

  Q: How can a router find out the set of BGP features supported by a peer ?

  A: Use version number

  - "textbook" approach
  - used from BGP-1 through BGP-4
  - poor fit for handling independent features: N independent features would require 2^N different version numbers

Late-90s: Capability Advertisement

- **BGP Capability Advertisement:**
  - instead of mapping a set of supported features to a particular version number, advertise support for each such feature at the BGP session establishment
- **BGP Capability Advertisement provides a (much) more flexible (and direct) way of introducing new features**
- **BGP Multiprotocol extensions was the first application of BGP Capability Advertisement, but not the last one**
  - more applications to follow
- **Thanks to BGP Capability Advertisement, today we still have BGP version 4 (BGP-4)**

Late 90s – beginning 2000s : new applications of BGP

- **Extending BGP to support services other than the Internet:**
  - BGP/MPLS VPNs (aka 2547 VPNs) - 1998
  - BGP for VPN auto-discovery - 2000
  - BGP-based Virtual Private LAN Service (VPLS) – 2002
- **Made possible by a combination of multi-protocol extensions and capability advertisement**
  - as well as all previous enhancements to BGP (e.g., route reflectors, route dampening, etc…)
- **Extends reachability information carried by BGP well beyond IPv4 or IPv6 address prefixes**
- **Generated LARGE amount of controversy**

New applications of BGP: controversy

- Claim: BGP should not be used by such applications as BGP/MPLS VPNs (2547 VPNs), VPN autodiscovery, VPLS, etc… because BGP was not designed to support these applications
- Dubious argument, as from a practical point of view what matters is not what BGP was designed for, but what BGP can do in a cost effective manner
- Remember that BGP was NOT designed to handle CIDR, was NOT designed to handle 200,000+ Internet routes, was NOT designed to support IPv6, etc…

New applications of BGP: controversy (cont.)

- **Claim: BGP should not be used by applications that require BGP to carry non-routing information, because BGP was designed to carry only routing information**
  - e.g., BGP should not be used for VPN auto-discovery and/or VPLS because these applications require BGP to carry non-routing information
- **Dubious argument:**
  - e.g., why BGP should be restricted to carry only "routing" information ?
- **From a practical point of view what matters is not whether the the (new) information is "routing", but whether the requirements for the distribution of the (new) information are similar to what is already provided by BGP**

New applications of BGP: controversy (cont.)

- **Claim: BGP gets more and more complex to implement**
- **If there is a (market) demand for the new functionality, this functionality will be implemented no matter what**
- **The real question is whether the new functionality should be implemented by extending BGP, or by inventing new protocols**
- **From a system-wide perspective extending BGP makes the overall system less complex than inventing new protocols**
  - as long as the new functionality matches what is provided by BGP
  - even if it adds complexity to a particular component of the system – BGP

New applications of BGP: controversy (cont.)

- **Claim: extending BGP to support new applications adversely impacts overall router software reliability ?**
  - by creating the situation where software bugs in the extensions needed to support one service affect the rest of the protocol (and thus other services)
- **Dubious argument:**
  - more protocols means more lines of code; number of bugs tends to be proportional to number of lines of code
  - competent system design and software engineering eliminates the situation where software bugs in the extensions needed to support one service affect the rest of the protocol (and thus other services)
    - e.g., use a distinct control plane instance for each service

Securing BGP

What problem(s) are we trying to solve ?

- authenticate peers,
  - implemented a while ago (TCP MD5), but still fairly limited deployment
- or authenticate originator of the routing information,
  - Secure Origin BGP (So-BGP) – no deployment
- or authenticate originator and the path traversed by the routing information, or…
  - Secure BGP (S-BGP) – no deployment

Securing BGP – things to consider

- **Cost/benefit consideration is the main factor that influences the deployment**
- **Who is going to bear the cost ? Who is going to benefit ?**
  - could those who bear the cost reap the benefit ?
  - could those who do not bear the cost reap the benefit ?
- **Could it be deployed in an incremental fashion ?**
- **Could benefits be obtained even in a partial deployment ?**
  - if yes, then how much ?
- **Impact of competition on cooperation among (competing) service providers should not be ignored**

Recent enhancements

- **Graceful Restart**
- **Route Target Constrain**
- **Carrying multicast routing information in BGP**
  - In support of multicast in 2547 VPNs
- **Etc…**

Recent enhancements – Route Target Constrain

- **Use BGP to distribute information that constrains BGP information distribution**
  - Use BGP to distribute route filtering information used by BGP
    - E.g., distribution of Route Target Communities by BGP is used to constrain distribution of VPN information tagged with the Route Target Communities
- **Recursive use of BGP**

In conclusion

- **Evolution by trial and error**
  - do not be afraid to introduce new features
  - use operational experience to test usefulness of new feature
    - rather than have endless arguments in the absence of any empirical evidences
  - do not be afraid to get rid of features turned out to be useless
  - design protocol to facilitate addition/deletion of features

In conclusion (cont.)

- **Just-in-time development**
  - focus on solving practical problems in real time – emphasis on engineering
  - do not spend too much time on solving anticipated/future/potential problems
- **Maximize re-use of the existing mechanisms**
  - by making new mechanisms sufficiently general
  - by being satisfied with “good enough” match between what is required and what is provided by the (existing) mechanisms

In conclusion (cont.)

- Short-term solutions tend to stay for a long time; long-term solutions tend to never happen
- “Good Enough” solutions are sufficient; “perfect” solutions are not necessary
- Meet market needs and accommodate technical progress by focusing on flexibility and extendibility, not by depending on the “crystal ball”
- Do not be afraid to question and violate, if needed, “architectural” principles (or any other dogmas)

Design by pragmatism