Towards a Next-Generation Inter-domain Routing Protocol

1999 Internet Map
Coloured by ISP
Source: Bill Cheswick, Lumeta
AS-level Topology 2003
Source: CAIDA
Inter-domain Routing

Tier-1 ISPs

Tier-2 ISPs

Tier-3 ISPs and Big Customers
Inter-domain Routing

Tier-1 ISPs
AS 1
AS 2

Tier-2 ISPs
AS 3
AS 4
AS 5
AS 6
AS 7
AS 8
AS 9
AS 10

Tier-3 ISPs and Big Customers

Net: 128.16.0.0/16
ASPath: 5,2,1,3,6

Net: 128.16.0.0/16
Inter-domain Routing

Net: 128.16.0.0/16

Route would Loop
Inter-domain Routing

Tier-1 ISPs

AS 1

AS 2

AS 3

AS 4

AS 5

AS 6

AS 7

AS 8

AS 9

AS 10

Net: 128.16.0.0/16

Tier-2 ISPs

Tier-3 ISPs and Big Customers

Prefer shortest AS path

1,3,6

2,1,3,6

AS 1

AS 2

AS 3

AS 4

AS 5

AS 6

AS 7

AS 8

AS 9

AS 10
Inter-domain Routing Policy

Tier-1 ISPs
AS 1
AS 2
AS 3
AS 4
AS 5
AS 6
AS 7
AS 8
AS 9
AS 10

Tier-2 ISPs

Net: 128.16.0.0/16

Tier-3 ISPs and Big Customers

Only accept customer routes
Inter-domain Routing Policy

Tier-1 ISPs

Tier-2 ISPs

Tier-3 ISPs and Big Customers

Net: 128.16.0.0/16

Don’t export provider routes to a provider
Inter-domain Routing Policy

 Tier-1 ISPs

 Tier-2 ISPs

 Net: 128.16.0.0/16

 Prefer customer routes

 Tier-3 ISPs and Big Customers
Inter-domain Routing

BGP4 is the only inter-domain routing protocol currently in use world-wide.

- Lack of security.
- Ease of misconfiguration.
- Policy through local filtering.
- Poorly understood interaction between local policies.
- Poor convergence.
- Lack of appropriate information hiding.
- Non-determinism.
- Poor overload behaviour.
What problem does BGP attempt to solve?

- *Global interconnectivity* between Internet providers.
- *Dynamic routing* in the presence of failure.
  - An approximation to *shortest-path* routing.
  - Subject to *local policy* constraints of each ISP.
Policy, policy, and policy

- An ISP’s routing policy is a commercial secret.
  - Don’t want to tell anyone else what the policy is.
  - BGP does policy entirely through local filtering of the set of possible alternative routes.

- Need path information to set a useful range of policies.
  - But path information inherently reveals information about routing adjacencies.
  - Can trivially infer many (most?) simple policies from looking at the routing tables.
Local Filtering

*Doing policy entirely through local filtering is the root cause of many of BGP’s problems.*

- Low-level mechanism for configuring what not to accept is prone to misconfiguration.
- No semantics in the protocol as to why a route is used make it hard to discover errors or attacks.
- No information about alternative routes means BGP must to a lengthy path exploration to figure out which alternatives are feasible.
- No information about which alternatives will work for whom means BGP can’t do effective information hiding.
  - Small changes in one part of the world are frequently globally visible.
Policy Hiding

- It’s not practical to hide most customer/provider routing relationships when using BGP.
  - Customer pays provider to advertise their route to the rest of the world.
- It is practical to hide many private peering relationships.

- Perhaps 95% of the “peerings” visible in route-views and RIPE appear to function as customer/provider links.
  - Note that the flow of money and whether a peering effectively functions as a customer/provider link are not necessarily correlated or revealed by the routing protocols.
Towards a Routing Framework

- Given that:
  - Most links function as customer/provider.
  - Customer/provider links are inherently visible to the world.
  - Additional semantics visible in the routing protocol would allow more informed route calculation, and permit better information hiding.

- Then it seems logical to design a routing protocol that uses this information explicitly.
IP Address Space

- The IP address space is a mess.
  - At best, a poor relationship between topology and address prefixes.
  - Many prefixes per AS.

- Binding between address prefixes and organizations is pretty stable.
  - Routes to a prefix change much more rapidly though due to failure or reconfiguration upstream.
Towards a Routing Framework (2)

*Separate dynamic routing from address prefix binding.*

- Use one protocol to distribute bindings between an address prefix and an origin AS.
  - Relatively static binding.
  - Can use strong crypto and offline computation to secure this binding.
- Use another protocol to dynamically calculate paths to origin ASes.
  - Dynamic calculation, needs fast reconvergence.
  - Different security mechanisms are appropriate.
Routing Hierarchy

- Autonomous Systems
- Other Peering Links
- Customer/Provider Links
Routing Hierarchy

Customer/Provider Links

Customer/Provider Hierarchy

Other Peering Links
Multiple Routing Hierarchies

- There is more information available within a routing hierarchy than there is between them.
  - Different routing algorithms may be appropriate.
Routing Protocol Styles

- Link-state:
  - Great convergence properties.
  - Scales fairly well.
  - Can’t easily hide policy information.

- Path-vector:
  - Poor convergence properties.
  - Scales well.
  - Can hide policy information and implement today’s routing policies.
Hybrid Link-State/Path Vector (HLP)
Hybrid Link-State/Path Vector (HLP)

*Within Customer-Provider link-state tree:*
- Good convergence.
- More information.
  - Eg. alternative route pre-computation.
  - Explicit representation of backup link for multihoming.
- Default policy is simple (reduces misconfiguration errors) and robust.
- Improved default security.
  - Need to be a tier-1 to do much damage.
Hybrid Link-State/Path Vector (HLP)

Between Customer-Provider trees:

- Use fragmented path-vector (FPV), rather than full path-vector used by BGP.
  - Number of links routed using FPV decreased drastically.
  - Reduces path-exploration space.

- Degrade gracefully from link-state towards path-vector if ISPs need to use more non-default policies.
  - Worst case looks pretty much like BGP.
Routing Messages

- LSA(C,E) cost=1
- FPV(A,E) cost=2
- FPV(B,A,E) cost=3
- FPV(B,A,E) cost=4
Route Change

- LSA(C,E) cost: $\infty$
- FPV(A,E) cost=3
- FPV(B,A,E) cost=4
- FPV(B,A,E) cost=5
Hybrid Link-State/Path Vector (HLP)

*Isolation and Information Hiding.*
- Lots of information within a Customer-Provider tree.
- Don’t need to convey all changes into FPV.
  - Local changes that aren’t too critical can be hidden from the wider world because it’s easy to see that similar metric alternatives exist within the Customer-Provider tree.
  - Only large-scale changes need to be pushed via FPV.
- Significantly reduce global routing table churn.
Exceptions

- Not all policies conform strictly to the hierarchy
  - Export-policy exception.
  - Prefer-customer exception.
- Dealt with in HLP by using FPV rather than Link-state.
- Fortunately this is rare. Frequency of export-policy exceptions:

<table>
<thead>
<tr>
<th>Type</th>
<th>Oct ‘03</th>
<th>Jun ‘03</th>
<th>Jan ‘03</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prov-Prov</td>
<td>0.8%</td>
<td>0.1%</td>
<td>0.3%</td>
</tr>
<tr>
<td>Prov-Peer</td>
<td>0.5%</td>
<td>0.5%</td>
<td>0.4%</td>
</tr>
<tr>
<td>Peer-Prov</td>
<td>0.1%</td>
<td>0.1%</td>
<td>0.1%</td>
</tr>
</tbody>
</table>
Performance: Routing Table Churn

Cumulative percentage

Churn

BGP
HLP
Performance: Fault Isolation
Fault Isolation and Multihoming

Cumulative percentage

ratio of BGP to HLP isolation
Convergence

- BGP: Worst case is fully connected \( n \)-node graph:
  - Convergence time is \( O((n-1)!) \)

- HLP: In the absence of exceptions, worst case is:
  - Convergence time is \( O(n^{k(D)}) \)
  - \( k(D) \) is number of peering links on path to \( D \)

In the current Internet:

- \( k \leq 1 \) for 90% of Internet routes
- \( k \leq 2 \) for 99% of Internet routes
- \( k \leq 4 \) for all Internet routes
HLP Advantages

- **Scalability**: route churn is the issue.
  - Information hiding.
  - Separation of prefix distribution from routing.

- **Convergence**:
  - Link-State converges fast.
  - FPV converges faster than Path-Vector because there are fewer infeasible alternates.

- **Security**:
  - Structure adds security.
  - Secure prefix distribution separately from dynamic routing.

- **Robustness**:
  - Harder to misconfigure, easier to figure out what the intent behind a route is.
HLP: Summary

- Understanding policy is critical to understanding how to change routing.
  - Need broad industry participation to get this right.
- Most policy is simple, some is very complex, some is inherently public, some must be kept private.
  - BGP doesn’t distinguish.
  - HLP tries to take advantage of the common case, and the inherent limitations on what can be kept private.
- Transitioning away from BGP will be really hard.
  - Can’t happen with strong incentive, and good consensus on where we want to get to.
Criteria for Successful BGP Replacement

- Interoperate with BGP without any serious degradation in capability during transition.
- Provide incremental improvement when customers and their providers both switch outside-in deployment.
- Concepts must be familiar to ISPs.
Opportunity for Replacement?

- BGP must be seen to be failing.
  - Security problems being actively exploited?
  - Convergence problems too slow for high-value traffic (VoIP, IP-TV)?
  - Growth of multi-homing causes routing table growth/churn that is unsupportable?