Some Foundational Problems in Interdomain Routing

Nick Feamster and Hari Balakrishnan
MIT Computer Science & Artificial Intelligence Lab
{feamster,hari}@csail.mit.edu

Jennifer Rexford
AT&T Labs–Research
jrex@research.att.com

ABSTRACT

The substantial complexity of interdomain routing in the Internet comes from the need to support flexible policies while scaling to a large number of Autonomous Systems. Despite impressive progress in *characterizing* the various ills of the Border Gateway Protocol (BGP), many problems remain unsolved, and the behavior of the routing system is still poorly understood. This paper argues that we must understand interdomain routing in terms of: (1) intrinsic properties and design tradeoffs of policy-based routing, independent of the specific routing protocol and (2) properties that relate to artifacts in today's protocol. We pose open questions for the research community that, if answered, should help us understand why BGP's many problems are so difficult to fix. Understanding the fundamental properties of interdomain routing will help us decide how to make progress, be it making backward-compatible modifications to BGP or designing a radically different protocol.

1. INTRODUCTION

The current interdomain routing protocol, Border Gateway Protocol (BGP4) [26], has evolved over the past decade and now constitutes a critical part of the Internet infrastructure. Two requirements for interdomain routing are at the root of BGP's bewildering complexity:

1. *Policy.* ASes have business relationships with one another but must cooperate to achieve global reachability. Operators use routing policies to control the flow of outbound (and, to some degree, inbound) traffic and specify which routes are advertised to neighboring networks under what conditions. Even the most basic properties of the protocol depend on how the policies are configured.
2. *Scalability.* Routing protocols must scale with increasing network size. The main mechanism to achieve scalability is aggressive aggregation of routing information, including destination prefixes. In large ASes, routing information is restricted to improve scalability: each router in the AS receives a summary of routing information from its *route reflector*, rather than relying on a "full mesh" of routers communicating with each other.

In turn, the complexity of the routing system has caused some "bursting at the seams". As BGP has evolved, network operators and researchers have encountered, and often analyzed, many problems that cause persistent oscillation [16, 18, 29], forwarding loops [18], slow convergence [21], and network partitions [7]. Previous studies have given us some collective understanding of many of BGP's problems, and in many cases "point fixes" have alleviated the problem at hand, but often at the cost of increasing complexity even further. We contend that the array of incremental fixes has led overall to a *poorer*, rather than better, *foundational* understanding of the right way to design a policy-rich, scalable interdomain routing protocol. Indeed, despite much effort and even some successes, hardly anyone would claim that Internet routing is a "solved problem".

Rather than attempt to fix specific problems in BGP or delve further into esoteric, BGP-specific arcana, we believe that it is now time to ask: *Which problems and properties of interdomain routing stem from the particular vagaries of BGP, and which represent fundamental limitations and challenges of scalable policy-based routing?* Tackling this question is a necessary prerequisite to improving interdomain routing; as a community, we may choose to develop a radically new routing protocol, or we may decide to make incremental modifications to BGP, but in both cases we will benefit from understanding routing problems at a foundational level.

Although some of BGP's problems result from the specific details of how BGP operates today, we believe that many problems reflect intrinsic properties and tradeoffs of scalable policy-based routing and are independent of the protocol that is used to disseminate routes (*e.g.*, path vector, link state, central server, etc.). These properties and tradeoffs may cause problems that cannot be solved for any scalable policy-based interdomain routing protocol. We pose questions that, if answered, will help understand the intrinsic problems with scalable policy-based interdomain routing and allow us to better understand the design tradeoffs of path-vector, policy-based routing. Our questions are not comprehensive, but they should help us understand why "fixing" BGP has proved so challenging.

Let's consider an example of a problem caused by policy requirements. Previous work found that BGP is vulnerable to persistent oscillations, such as the "bad gadget" scenario [16]. In this situation, three (or more) ASes continually oscillate between their available routing choices because each AS prefers to route indirectly via another AS rather than directly to the destination. There are two possible reactions to such a discovery. One could assert that "BGP is broken" and condemn the protocol designers. On the other hand, one could recognize that the oscillation is a fundamental problem that results from the inability to satisfy group preferences. To some extent, both reactions are correct. In Section 3.1.1, we will pose questions that address the tension between local policy and protocol convergence.

Now let's consider a design tradeoff introduced by scalability requirements. To scale to a large number of destinations, routers maintain routes based on IP prefixes (addresses that share the first n bits). Combining multiple contiguous address blocks into a single prefix reduces routing table size, at the expense of hiding information about different downstream paths and failure modes for the smaller subnets. More specific prefixes give an AS a finer granularity of control over inbound traffic; however, the extra prefixes result in more BGP messages and larger routing tables throughout the Internet. We discuss this tradeoff further in Section 4.1.2.

This paper highlights problems of practical significance that we have encountered in our work on BGP verification, modeling, convergence, and traffic engineering. Section 2 defines a model of interdomain routing. Sections 3 and 4 explore problems related to BGP's policy and scalability requirements, respectively.

2. INTERDOMAIN ROUTING MODEL

We now define a model of interdomain routing that scopes our discussion. ASes exchange routing information via exterior routers at one or more locations. Each AS has interior routers that obtain information about external routes from the exterior routers. Given any set of available routes to a destination d, S_d, each router selects a best route, $r_d = \lambda(S_d)$. Every router must have a preference relation for all $a, b \in S_d$: either $a \prec b$, or $b \prec a$. Each router applies an export policy to determine the neighboring routers to which it should readvertise its current best route in S_d.

This model captures many features of BGP. The preference function, λ, incorporates the BGP decision process and the effects of routing policies on route selection. In BGP, each router propagates *only* the best route (or nothing) for a destination to a neighboring router. The notion of "exterior" and "interior" routers reflects the general property that some routers in a network will exchange routes in other administrative domains and others will not; it also allows for distinctive behavior in the two realms: eBGP for exchanging routes between ASes, and iBGP for exchanging routes between interior routers. The model also recognizes that each router in an AS may make different decisions, as in BGP. The model also reflects several limitations of the current interdomain routing policy. BGP does not permit policies that dictate which ASes *must be* and *must not be* traversed en route to a destination.

3. POLICY-INDUCED PROBLEMS

Introducing policy into interdomain routing causes two main problems: protocol oscillations and weak security.

3.1 Protocol Oscillations

Instability results from two main causes: inter-AS oscillations (caused by policy disputes) and intra-AS oscillations (caused by non-monotonic ranking functions).

3.1.1 Policy Disputes

Because BGP's path selection is based on an AS's local preferences, rather than shortest paths, a group of ASes can have preferences that cause BGP to oscillate forever [16, 29]. These "policy disputes" occur because there is no possible path assignment for which at least one AS in the system does not have a better path available; thus, that AS would switch to the better route. That act of switching creates a different path assignment that is also unstable.[1] *Even when given stable inputs, BGP might never converge!*

Griffin *et al.* showed that, in general, determining whether a set of ASes would experience a policy dispute is an NP-complete problem [16]. They also defined the concept of a "dispute wheel", which describes a circular relationship among a group of ASes where each AS prefers an indirect route via another AS in the group over a direct route to the destination. They showed that sets of policies without a dispute wheel are guaranteed not to oscillate. Checking for a dispute wheel requires a *global* view of policies, but Gao and Rexford observed that, if every AS considers each of its neighbors as either a customer, a provider, or a peer, and obeys certain local constraints on preference and export policies, then BGP is guaranteed to converge [13, 12].

It might seem that the dispute problem is "solved" because the Gao/Rexford constraints are realistic and they guarantee convergence. We disagree. First, it may

[1]For those familiar with game theory, the situation is analogous to a game where there is no *pure strategy* Nash equilibrium: for any set of pure strategies, there is one player who is better off switching strategies.

be difficult to guarantee that the constraints are satisfied (the proposed constraints on inter-AS relationships are a global property). Second, there may be legitimate reasons to deviate from the guidelines: an AS may decide to provide transit between two peer ASes (which violates the Gao/Rexford constraints) as part of a special business relationship. Fundamentally, the decision to export routes is *contractual*—it reflects a willingness to carry traffic—and each AS should retain autonomy over whether it exports routes for a prefix to a particular neighbor. In other words, an AS's decision to advertise a route should be based on whether its neighbor is paying to see that route, *not* dictated by whether the routing protocol will converge. To understand whether this is even possible, we must first answer a more fundamental question: *Is it possible to design a policy-based protocol that always converges?* We pose two open questions:

- *Policy restriction.* Given no restrictions on topology or export policies, how must preferences be restricted (*i.e.*, what are the restrictions on the preference function, λ) to guarantee convergence?
- *Protocol changes.* Can a routing protocol that incorporates additional information or features (*e.g.*, negotiation, pricing, and randomization) permit more flexible export policies and preferences while still guaranteeing convergence?

3.1.2 Non-monotonic Ranking

When advertising routes for some destination to a neighboring AS at multiple network locations, an AS can attach a route attribute called the multiple-exit discriminator (MED) to express its preferences regarding which route the neighbor should use. For example, a network advertising a route in both San Francisco and New York may place a large MED value on the route advertised in San Francisco to indicate to the neighboring AS that it would prefer traffic for that destination to enter in New York. MED provides useful semantics because it allows one AS to express preferences to its neighbor over where traffic enters its network for some destination.

Because the actual MED values are set by the AS that advertises the route, the AS that receives the route *cannot* compare the MED values across routes received from two different ASes for the same destination. As a result, routers may not have monotonic preferences between pairs of routes. That is, when combined, pairwise preferences do not form a total ordering (*i.e.*, it is possible for (1) $a \prec b$, (2) $b \prec c$, and (3) $c \prec a$). Because the routers in an AS cannot express a monotonic ranking, *BGP can also oscillate within a single AS* [18].

Because MED provides useful semantics, recent work has explored how to retain MED while preventing the oscillation problems it causes. Basu *et al.* suggested a modification to iBGP whereby each router readvertises *multiple* candidate routes from S_d [1]. This proposal prevents MED-induced oscillation, but it requires modifications to all deployed BGP-speaking routers.

AS-specific semantics are still poorly understood:

- *Exit point semantics and oscillation.* Is it possible to preserve exit-preference semantics *and* ensure that each router has monotonic preferences?
- *Total ordering and oscillation.* Are there sets of preferences for which pairwise ranking is impossible, but for which routing will always converge?

We believe the problems caused by MED can be eliminated by making the MED attribute comparable across routes learned from different ASes and remapping MED values in a way that preserves the exit semantics.

3.2 Weak Security

Interdomain routing involves thousands of competing ASes, each of which sets policy about the routes it is willing to accept and propagate. Routing security has been studied in some detail [25], but *interdomain* routing security is particularly difficult because interdomain routing must support complex policy. Today's infrastructure provides scant support for either preventing or detecting invalid routes. BGP does not allow an AS to verify that a route it learns is valid and provides no guarantees about where packets will actually go.

3.2.1 Control-Plane Security

BGP does not provide any support for controlling route announcements. Specifically, *BGP does not prevent an AS from advertising arbitrary prefixes*. One of the most fundamental problems in interdomain routing is determining whether an AS is authorized to announce a certain prefix. S-BGP proposes using certificates to bind IP address space to the AS that owns the space, but this solution requires a public key infrastructure, expensive cryptographic operations, and relatively high message overhead [20]. Other work has tried to work around BGP, rather than modifying it, by proposing a secure registry that ASes can query out-of-band to determine prefix ownership [14]. Since PKIs are cumbersome and registries are difficult to maintain, this problem is largely unsolved, and the following questions are still relevant:

- *Decentralized verification.* Is it possible to verify prefix ownership without requiring a centralized, trusted verification or lookup infrastructure?
- *In-band verification.* Is it possible to design an in-band verification scheme, or is there some intrinsic property related to path-vector or policy-based protocols that requires the ownership problem to be solved out-of-band?

S-BGP, SPV [19], and Whisper [27] all attempt to provide "path authentication", which verifies that the AS path in a route corresponds to the sequence of ASes that the route advertisement actually traversed. However, these approaches do not enable an AS to verify that the route it receives is one that it *should be receiving*.

For example, an AS typically wants to filter routes that have "valleys"—those that use a small customer AS to transit between two larger ASes [11]. In order to detect these routes, an AS must know the relationships between other ASes, but inferring this information assumes that ASes advertise routes correctly in the first place. Verifying that routes are actually ones that an AS would want to use might be possible with appropriate filtering techniques, but several open questions remain:

- *Detecting routes that violate policy.* Given the limited information that any given AS learns, what types of bogus routes can an AS detect (*e.g.*, routes that are not valley-free)? What additional information could be added to the routing protocol to make this easier? Would adding this information reveal too many details about business relationships?

3.2.2 Data-Plane Security

Even if an AS could verify that the routes it receives were authentic and policy-compliant, it still *cannot* verify that packets actually traverse the same ASes as those in the route's AS path [23]. Since policy should ultimately dictate the path that *data* packets take, we ask:

- *Verifying the forwarding path.* How can an AS verify that a route's AS path matches the actual forwarding path?

A router should reject packets destined for hosts with no corresponding advertised route. More generally, a router should reject packets from sources that should not have a valid route through this router to the destination. Deploying packet filters only at routers in large ASes in the Internet "core" could eliminate most of these packets [24], but an AS must be able to construct these filters in the first place, which would require discovering the routes from the source to that AS.

- *Dynamic packet filter construction.* How can an AS determine the source and destination addresses it should see on packets arriving on a link? [2]

4. SCALABILITY-INDUCED PROBLEMS

Interdomain routing must scale to many ASes, routers, and destinations. The three main scaling techniques—(1) representing an AS as a single node (*e.g.*, in the AS path), (2) route reflection, and (3) prefix aggregation—reduce overhead by *hiding routing information*. In this section, we review how this obfuscation causes serious problems (e.g., slow convergence, forwarding loops, persistent oscillation, and network partitions) and pose open questions related to solving these problems.

[2] This question has a dual for *route* filters in the control plane.

4.1 Missing Topology Details

BGP abstracts the routing details inside each AS and aggregates information about routes to individual destinations. These techniques allow BGP to scale, but they also make it difficult to determine the cause of a routing update, which can slow convergence, prevent problem diagnosis, hide fine-grained information about the reachability of destinations, and reduce an AS's control over incoming traffic.

4.1.1 Inability to Pinpoint the Causes of Updates

Many aspects of BGP abstract an AS as a single node. An AS that receives a route from a neighbor learns only the next-hop IP address for that route and the AS path (the sequence of ASes that advertised the route). This abstraction provides scalability because an AS need not be concerned about how neighboring ASes route packets within their networks. On the other hand, it makes pinpointing the origin of a routing update difficult (if not impossible), because an AS has essentially no information about the origin of a route change or withdrawal. The inability to pinpoint the source of a routing update slows convergence and complicates diagnosis.

Labovitz *et al.* observed that, *when a BGP route is withdrawn, routers may explore $O(N!)$ alternate paths, where N is the number of ASes in the system* [21]. Much subsequent research has proposed modifications to BGP purporting to prevent path exploration, so it may appear that path exploration is a consequence of bad design. In actuality, the jury is still out. In particular, many of the proposed solutions, which attempt to skip the exploration of routes that share a failed AS edge or subpath, assume that each pair of ASes only has a single edge that can fail. Preventing path exploration appears to require additional information about individual edges between ASes, which adds a significant amount of complexity [4]. Such a modification breaks the "single node abstraction"; can it be made to scale? We must understand the following tradeoff:

- *Scalability vs. convergence speed.* What information does BGP need to prevent path exploration and speed convergence? Will such a modification cause too many routing messages or instability?

Recent work on "root cause analysis" aims to pinpoint the location and cause of a routing change by analyzing streams of BGP update messages across prefix, time, and vantage point [3, 10]. Note that if routers could successfully pinpoint the cause and location of BGP updates, then protocol convergence could be significantly faster. Unfortunately, the same single node abstraction that makes it hard to reduce convergence delay also makes root cause analysis difficult: an internal routing change may not always induce a BGP update at every one of the AS's routers [28], and the ability to see these updates depends on where BGP messages are observed. Additionally, if a routing change inside an AS affects some, *but not all*, destination prefixes reachable via that AS, these techniques may incorrectly trace the origin of the update to an AS farther downstream [28]. These results beg the question: Given the limited information in the AS path, is it fundamentally impossible to pinpoint the cause and location of a routing change?

- *(Im)possibility of problem diagnosis.* Do BGP messages provide enough information to pinpoint cause and location of a routing change? What additional information would facilitate diagnosis?

4.1.2 Coarse Information about Destinations

To control routing table size, routers maintain routes for IP *prefixes* using a process called aggregation. Aggregation dramatically reduces routing table size because Internet addressing is typically hierarchical. Unfortunately, the side effects of aggregation are not well understood. Aggregation can hide important information, such as the fact that groups of destinations within a single prefix may not share fate (*e.g.*, the destinations may be geographically distributed). Aggregation can cause mismatches between the AS-level forwarding path and the BGP AS path [23] and can foil an AS's attempt to implement a backup path [30] or control inbound traffic.

- *Scalability vs. flexibility.* In what situations does aggregation limit the ability to perform inbound traffic engineering or implement backup paths?

Even if a router makes the same routing decision for two contiguous prefixes, it typically cannot aggregate those routes because it does not know if other routers that learn BGP routes from this router might select different routes for the separate prefixes. As a result, ASes must either aggregate routes at the risk of interfering with the traffic engineering goals of other ASes or maintain larger routing tables without receiving compensation for the incurring the extra overhead.

- *Incentives for more specific routes.* Should the routing protocol incorporate incentives to encourage an AS to maintain more specific prefixes?

4.2 Missing Routes

A route reflector selects a single best route for each destination and advertises this route to its clients [2], obviating the need for each pair of routers in an AS to exchange routes. Route reflectors reduce the number of iBGP sessions in an AS and the total number of BGP routes that each router must learn but can cause forwarding loops, protocol oscillations, and partitions.

4.2.1 Deflections, Loops, and Oscillations

Because route reflectors perform route selection on behalf of their clients, *route reflection does not assign the same routes as a full-mesh iBGP configuration*.

In a network with route reflectors, a router may select a BGP route with a different exit point than some of the routers along the forwarding path to that exit point, causing the routers to "deflect" data packets toward another exit point [18]. Previous work has underscored the severity of deflections by showing that certain iBGP topologies can cause persistent forwarding loops [5]. Deflections can also cause packets to traverse a different sequence of ASes than the ones in the AS path [23]. In general, deciding if an iBGP topology induces forwarding loops is an NP-complete problem [6]; Griffin *et al.* outline sufficient conditions for iBGP "forwarding correctness" (*i.e.*, freedom from loops and deflections) [18].

These sufficient conditions might seem to close the book on iBGP correctness, but we disagree. First, these sufficient conditions are very strong: they imply that *every edge* that is on a shortest path to an exit point must have a corresponding iBGP session. Second, the conditions require that redundant route reflectors must be located close to the primary to have a similar view of the best routes, introducing undesirable fate sharing. Finally, we have recently discovered IGP ("interior gateway protocol"; *e.g.*, OSPF) topologies for which this constraint is not satisfiable. In light of this, we ask:

- *Forwarding correctness.* Are there weaker sufficient conditions for forwarding correctness?
- *Redundancy and route reflection.* Is it (im)possible to achieve both redundancy and deflection-free routing with route reflectors?
- *Computing correct iBGP topologies.* Given an IGP topology, what is an efficient algorithm to compute a forwarding-correct iBGP topology?

Route reflectors can cause a router's preference over two exit routers to depend on the presence or absence of routes from other routers, which can cause oscillations [7, 18]. Thus, the question about ordering and oscillation from Section 3.1.2 also applies in this context.

4.2.2 Network Partitions

An iBGP topology can create partitions, *even if the underlying IP-level topology is connected* [7]. For example, a missing iBGP session might keep one router from receiving any route for a destination prefix, leading to a blackhole that discards data packets headed toward that destination. We proved that a connected iBGP topology will not create a partition if and only if the routers at the top of the hierarchy all have iBGP sessions with each other [7], but other questions remain:

- *Guaranteeing path visibility.* What are sufficient conditions to guarantee visibility (*i.e.*, that every router will learn at least one route to a destination if an IP-level path to that destination exists), both in the common case and in the case of partitions in the underlying IP network?

We are exploring the design of a robust, scalable intra-AS routing architecture that guarantees visibility [8].

5. DISCUSSION

This paper poses questions that, if answered, should give us a better understanding of BGP's pitfalls. At a fundamental level, many of the problems concern three basic properties: *route validity* (the protocol does not propagate bogus routes), *path visibility* (the protocol propagates at least one route to a router if an IP-layer path exists), and *safety* (the protocol converges to a stable route assignment). Our work on creating a routing logic for analyzing these properties [6], detecting violations of these properties in practice [7], and enforcing these properties when assigning BGP routes to routers [8] may serve as a useful foundation for designing and evaluating possible solutions to these problems.

BGP's problems tend to fall into three main categories: those caused by protocol specifics, those that arise because BGP is a path-vector protocol, and those that are intrinsic to scalable policy-based routing. Thinking about BGP's problems in terms of these categories should help in evaluating assumptions in design, analysis, and measurement studies. Assumptions that violate BGP specifics may be reasonable: these specifics may ultimately be eliminated (especially if they cause problems), so the results may still be applicable. On the other hand, studies that rely on assumptions that run counter to intrinsic properties are less valuable.

Moving forward, we see two possible approaches to fixing BGP's problems. The first is to build fixes into the existing infrastructure by determining sufficient conditions (or what must be added to the infrastructure, separate from the protocol) for some property to be satisfied (*e.g.*, [7, 9, 14, 15, 16, 22]). This approach can help us understand BGP's limitations and also provide necessary machinery for improving the state of the art in the near term. The second is to design for intrinsic robustness by redesigning the protocol to explicitly prevent the problem (*e.g.*, [1, 8, 17]), which may provide more effective solutions in the long term.

Acknowledgments. This work was supported in part by Cisco Systems.

6. REFERENCES

[1] BASU, A., ONG, C.-H. L., RASALA, A., SHEPHERD, F. B., AND WILFONG, G. Route oscillations in IBGP with route reflection. In *Proc. ACM SIGCOMM* (August 2002).

[2] BATES, T., CHANDRA, R., AND CHEN, E. BGP Route Reflection - An Alternative to Full Mesh IBGP. RFC 2796, April 2000.

[3] CAESAR, M., SUBRAMANIAN, L., AND KATZ, R. H. Towards localizing root causes of BGP dynamics. Tech. Rep. CSD-03-1292, UC Berkeley, November 2003.

[4] CHANDRASHEKAR, J., DUAN, Z., ZHANG, Z.-L., AND KRASKY, J. Limiting Path Exploration in Path Vector Protocols. Tech. rep., University of Minnesota, 2004.

[5] DUBE, R. A comparison of scaling techniques for BGP. *ACM Computer Communications Review 29*, 3 (July 1999), 44–46.

[6] FEAMSTER, N., AND BALAKRISHNAN, H. Towards a logic for wide-area Internet routing. In *ACM SIGCOMM Workshop on Future Directions in Network Architecture* (August 2003).

[7] FEAMSTER, N., AND BALAKRISHNAN, H. Verifying the correctness of wide-area Internet routing. Tech. Rep. MIT-LCS-TR-948, Massachusetts Inst. of Tech., May 2004.

[8] FEAMSTER, N., BALAKRISHNAN, H., REXFORD, J., SHAIKH, A., AND VAN DER MERWE, J. The case for separating routing from routers. In *ACM SIGCOMM Workshop on Future Directions in Network Architecture* (Portland, OR, August 2004).

[9] FEAMSTER, N., WINICK, J., AND REXFORD, J. A model of BGP routing for network engineering. In *Proc. ACM SIGMETRICS* (June 2004).

[10] FELDMANN, A., MAENNEL, O., MAO, Z. M., BERGER, A., AND MAGGS, B. Locating Internet routing instabilities. In *Proc. ACM SIGCOMM* (Portland, OR, September 2004).

[11] GAO, L. On inferring autonomous system relationships in the Internet. *IEEE/ACM Trans. Networking 9*, 6 (December 2001).

[12] GAO, L., GRIFFIN, T. G., AND REXFORD, J. Inherently safe backup routing with BGP. In *Proc. IEEE INFOCOM* (Anchorage, AK, April 2001).

[13] GAO, L., AND REXFORD, J. Stable Internet routing without global coordination. *IEEE/ACM Trans. Networking 9*, 6 (December 2001), 681–692.

[14] GOODELL, G., AIELLO, W., GRIFFIN, T., IOANNIDIS, J., MCDANIEL, P., AND RUBIN, A. Working around BGP: An incremental approach to improving security and accuracy of interdomain routing. In *Proc. Network and Distributed Systems Security, Internet Society* (February 2003).

[15] GOVINDAN, R., ALAETTINOGLU, C., VARADHAN, K., AND ESTRIN, D. Route servers for inter-domain routing. *Computer Networks and ISDN Systems 30* (1998), 1157–1174.

[16] GRIFFIN, T., SHEPHERD, F. B., AND WILFONG, G. The stable paths problem and interdomain routing. *IEEE/ACM Trans. Networking 10,* 1 (2002), 232–243.

[17] GRIFFIN, T., AND WILFONG, G. A safe path vector protocol. In *Proc. IEEE INFOCOM* (March 2000).

[18] GRIFFIN, T. G., AND WILFONG, G. On the correctness of IBGP configuration. In *Proc. ACM SIGCOMM* (August 2002).

[19] HU, Y.-C., PERRIG, A., AND SIRBU, M. SPV: Secure path vector routing for securing BGP. In *Proc. ACM SIGCOMM* (Portland, OR, September 2004).

[20] KENT, S., LYNN, C., MIKKELSON, J., AND SEO, K. Secure border gateway protocol (S-BGP) - real world performance and deployment issues. In *Proc. Network and Distributed Systems Security, Internet Society* (2000).

[21] LABOVITZ, C., AHUJA, A., BOSE, A., AND JAHANIAN, F. Delayed Internet routing convergence. *IEEE/ACM Trans. Networking 9*, 3 (June 2001), 293–306.

[22] MAHAJAN, R., WETHERALL, D., AND ANDERSON, T. Interdomain routing with negotiation. Tech. Rep. CSE-04-06-02, University of Washington, May 2004.

[23] MAO, Z. M., REXFORD, J., WANG, J., AND KATZ, R. H. Towards an accurate AS-level traceroute tool. In *Proc. ACM SIGCOMM* (August 2003).

[24] PARK, K., AND LEE, H. On the effectiveness of route-based packet filtering for distributed DoS attack prevention in power-law Internets. In *Proc. ACM SIGCOMM* (San Diego, CA, August 2001).

[25] PERLMAN, R. *Network Layer Protocols with Byzantine Robustness*. PhD thesis, Massachusetts Institute of Technology, October 1988. MIT-LCS-TR-429.

[26] REKHTER, Y., LI, T., AND HARES, S. A Border Gateway Protocol 4 (BGP-4). Internet Draft draft-ietf-idr-bgp4-25.txt, September 2004.

[27] SUBRAMANIAN, L., ROTH, V., STOICA, I., SHENKER, S., AND KATZ, R. Listen and whisper: Security mechanisms for BGP. In *Proc. USENIX/ACM Symposium on Networked Systems Design and Implementation* (San Francisco, CA, March 2004).

[28] TEIXEIRA, R., AND REXFORD, J. A measurement framework for pinpointing routing changes. In *ACM SIGCOMM Workshop on Network Troubleshooting* (Portland, OR, August 2004).

[29] VARADHAN, K., GOVINDAN, R., AND ESTRIN, D. Persistent route oscillations in inter-domain routing. *Computer Networks 32*, 1 (2000), 1–16.

[30] WHITE, R., AND FEAMSTER, N. *Considerations in Validating the Path in Routing Protocols*. IETF, April 2004. Internet Draft. Expires October 2004.