Feedback Based Routing [*]

Dapeng Zhu
dapengz@cs.stanford.edu

Mark Gritter
mgritter@cs.stanford.edu

David R. Cheriton
cheriton@cs.stanford.edu

Department of Computer Science
Stanford University
Stanford, CA 94305

ABSTRACT

In this paper, we describe the problems that affect availability in BGP, such as vulnerability to attacks, slow convergence time, and lack of scalability. These problems arise from the basic assumption of BGP: every router has to cooperate to make routing work. We propose a new routing system, feedback based routing, which bifurcates structural information and dynamic information. Only structural information is propagated. Dynamic information is discovered by the routers based on feedback and probes. Routing decisions are made based on the dynamic information. We argue that this system is resilient to minority compromises in the infrastructure, provides higher availability than BGP, and can scale to the size of the Internet of the future.

1. INTRODUCTION

BGP is the current inter-domain routing system. In BGP, a router receives structural and dynamic information about the Internet from its peers, builds a model of the network from this information, then derives its routing table from this model [18]. Effective packet forwarding depends on this routing information being accurate and timely, and every router building a reasonably consistent model of the network connectivity and capacities.

This model worked great in the early days of the Internet, when the number of networks was small and the network operators trusted each other. The situation is different now. The networks connecting to the Internet are very diverse. It is not clear that a network operated by the U.S. government trusts a network operated by someone sympathetic to the terrorists. The number of networks, or in BGP terminology, prefixes that are visible in the BGP routing table is growing exponentially [11]. As a result, the amount of routing messages a router has to handle is growing exponentially [13]. Finally, the distributed nature of the BGP algorithm makes it extremely difficult to predict its behavior. For example, researchers have discovered that some routing policy combinations can cause BGP to diverge [23]. Many surprising discoveries like this([9], [13], [22], [21]) show the unpredictability and instability of BGP. These changes made three problems of BGP more prominent.

First, BGP is vulnerable to attacks from a single router. Such a concern is demonstrated to be valid by a failure [5] in which a misconfigured BGP router advertised to its peers that it had a direct route to every address in the Internet. This false information disrupted a large portion of the Internet for several hours. Recently, CERT warned of increasing focus on compromising routers by the hacker community [10], where there are active discussions on attacking routing protocols [6]. Going beyond the “prank” level of threat, there are concerns arising from recent terrorist events of the feasibility and effect of a serious attack on the Internet infrastructure.

There is a great deal of concern about the scalability of BGP, as the number of edge networks grow exponentially. This exponential growth caused some researchers to question whether BGP routers’ CPUs can handle the amount of BGP messages that will be generated by tomorrow’s Internet [14]. The correct functioning of BGP requires the cooperation of every router involved, and if some of the routers become overwhelmed with BGP updates, the whole routing system will take longer to converge.

BGP also suffers from slow convergence time, as a result of design decisions, the distributed decision making process, and the exponential growth of the Internet. Internet measurements have shown that BGP takes hundreds of seconds to converge after a routing failure [13]. Labovitz et al. also suggested that the convergence time will increase as the Internet grows [13]. As more mission critical communication systems, such as air traffic control and emergency communication, start to use the Internet as the underlying infrastructure, such service outages are simply unacceptable.

In this paper, we explore an alternative approach to inter-domain routing. In this system, we separate routing information into its structural and dynamic components. Structural information denotes the existence of links and is propagated to the edge of the Internet. Dynamic information denotes the quality of paths across the Internet. The routers

[*] This project is supported by the US Defense Advanced Research Projects Agency (DARPA) under contract number MDA972-99-C-0024.

at the core of the Internet only propagate structural information and forward packets. All routing decisions are done at the edge, based on structural information and end-to-end performance measurement.

This paper first describes the feedback based routing algorithm in section 2. We then analyze resistance to attacks, convergence time, and scalability in section 3. We discuss a mechanism to defend against Denial of Service attack, and a mechanism to implement fast fail-over virtual links in section 4, as applications of the routing system we propose. Related work is described in section 5.

2. ALGORITHM

2.1 Overview and Terminology

In our system, there are two types of inter-domain routers: transit and access. Figure 1 illustrates the location of these routers. Transit routers are usually located at the border of autonomous systems, while access routers are usually located at the border of an edge network. An edge network is the network of an organization. It is represented by one or more prefixes.

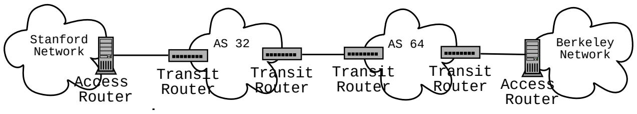

Figure 1: Access Routers and Transit Routers

Transit routers do not compute routing tables. They perform the following three functions: first, they forward packets according to routing information, which we call Internet Relay Tokens (IRT), in the packets. Second, they filter packets according to local access control list. Finally they propagate structural information. Access routers are responsible for inserting Internet Relay Tokens into IP packets. An IRT contains the list of autonomous systems a packet is going to travel. It is source routing at the granularity of autonomous systems. The packet format is specified in [7], part of the Wide-area Relay and Addressing Protocol (WRAP) specification.

In essence, our routing algorithm is composed of structural information propagation, route quality monitoring, and measurement based routing preference. An access router computes two routes to every advertised network prefix, based on the structural information it receives. The two routes are chosen to be as independent as possible. The access router monitors the quality of these two routes, and uses the better one to carry its traffic.

For the purposes of our algorithm, the Internet is modeled as a directed graph consisting of a vertex for each autonomous system and each edge network. The edges between vertices represent peering relationships; each edge may be either a single network link or many separate physical links. Since most Internet peering arrangements are bidirectional, the peering between autonomous systems A and B is represented by two directed edges $A \rightarrow B$ and $B \rightarrow A$. The peering between an autonomous system A and an edge network B is represented by one directed edge $A \rightarrow B$, since most edge networks do not carry traffic destined for other networks.

2.2 Structural Information Propagation

We associate a time stamp and an expiration time with every edge. An edge is removed from the structural graph after the expiration time is reached, and no new announcement is heard. The expiration timer should not be less than an hour. Routers announce the existence of edges periodically, but loss of edges are not explicitly advertised. Thus there are no link withdraw messages in our system.

For each edge $A \rightarrow B$ in the graph, there are three sets of advisory information associated with it, which determines whether a packet can be forwarded from A to B. The information is expressed in a packet matching rule. Therefore we also call the information rule sets. Here is an example: A packet with source address `136.152.197.176` and destination address `171.64.79.1` will match the rule `dst = 171.64.0.0/16` and the rule `src = 136.0.0.0/8`. In the expected case, the rules are simple matches on source and/or destination addresses. But they can be arbitrarily complex.

The first rule set is the positive rule set. If a packet does not match any rule in this set, then the edge does not exist with respect to this packet. The second rule set is called the negative rule set. If a packet matches any rule in this set, the edge does not exist with respect to this packet. Traffic engineering rule set is the third kind. Access routers that respect this rule set will not send packets that do not match any rule in this set. A transit router might have to enforce some rules (i.e. drop the packets that do not obey the rules), as a result of a contractual agreement. But they are free to ignore any rule if such an agreement does not exist, since it costs resource to filter packets. If we view BGP as a reachability maintenance protocol, then our protocol does not only maintain reachability, but also unreachability.

Neighboring vertices exchange information about edges they know about. Routers inside a vertex exchange connectivity information about their neighbors using an interior routing protocol.

Digital signatures can be used to increase the difficulty of attacks against the system. In a system without digital signatures, suppose there is a link from AS 32 to `128.12.0.0/16`, a malicious router might modify or generate a routing advertisement for this link which has a negative rule set of `0.0.0.0/0` — i.e., no packet is allowed to flow through the link. This modified rule can replace the valid link advertisement, causing a loss of connectivity to `128.12.0.0/16`. If both end points of an edge have to sign an edge, such an attack will not happen. However, digital signatures do not prevent other types of attacks. For example, a compromised router can announce an edge it has with a peer, but refuses to forward packets through this edge. If digital signature is used, an access router will not use an edge in its route computation unless its signature has been verified. The additional verification does not decrease the scalability of the protocol because the computation is off the critical path. In other words, even if an access router can not keep up with edge verification, it can still function correctly.

2.3 Algorithm for Access Routers

Upon receiving structural information, access routers build a graph representation of the Internet. This representation incorporates the advisory information for each link. Then for each destination, the access router tries to find two routes. The two routes should differ as much as possible. In the ideal case, when both the source and destination are multihomed, the two routes should be vertex disjoint. One route will be used as the primary route, and the other serves as the backup route.

When the access router first boots, it chooses the route with the shortest AS hop count as the primary route. In subsequent operations, the cost of a route is the round trip time to the destination. The access router chooses the route with the lowest cost as the primary route.

An access router can sample the round trip time (RTT) to a destination D by measuring the time between a TCP SYN packet and the corresponding SYN ACK packet. (Here, we focus on TCP traffic, the dominant type of Internet traffic, especially in the wide area.) It keeps a running average of the round trip time, similar to TCP [17]. An access router is the ideal place to monitor the feedback because all traffic between the organization and Internet flows through it. We assume that there is one access router per organization, since here is no reason for more than one. Standard techniques can be applied to increase the availability of the access router.

Occasionally, an access router generates probes to test the quality of a route, such as that of a newly calculated backup route. The SYN-SYN-ACK sampling of TCP traffic above allows the router to record a promising address in the destination address range to probe with. For example, it may observe a TCP SYN to port 80, which suggests that a web server is present. In this case, it can send a TCP SYN to the same specific address and destination port over this route and time the round trip time to the SYN ACK. In the absence of an identified server, the router uses an ICMP echo to attempt to measure the round trip time. As another alternative, the router can try to use the backup route to forward packets and measure the actual delay in the same way as for the primary route.

Figure 2 illustrates the operation of an access router at Stanford. Through feedback and probes, it learns that RTT to Berkeley through the upper route is 20ms, while the RTT through the lower route is 100ms. Therefore it uses the upper route to forward packets to Berkeley.

An access router periodically computes a primary route and a backup route based on its current view of the Internet. The computation also takes current routes into consideration. If both routes to a destination have an infinite RTT in the routing table, the edges in the two routes are excluded when computing the two new routes (except the edges that connect the source and destination to the Internet. We either know or have to assume that they are working.) The results from the computation are used as the primary and backup routes. The computation can fail with very small probability. In that case, we randomly exclude some old edge from the two routes and redo the computation.

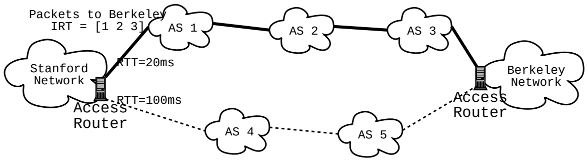

Figure 2: Primary and Backup Routes

If only the backup route has an infinite RTT, The vertices of the primary route, and the edges of the backup routes are excluded from the computation. The result is used as the backup route.

If neither route has an infinite RTT, only the vertices of the primary route are excluded from the computation to find a new route. If the result is different from the current backup route, it is probed. If the RTT is less than the RTT of the current backup route, it is used as the backup route.

3. ANALYSIS

3.1 Attack Resistance

One of the attacks against routing protocols is fooling other routers into using a link, while the link either does not exist, or does not function well. If digital signatures are used, then all the edges an access router knows are real. But one end point of the edge can drop all the packets that are supposed to go through the edge, thus creating a black hole.

Our routing system is robust with respect to such attacks. An access router constantly monitor the quality of a route it is currently using. If it can not reach the destination, it would try a different (at least edge disjoint) route, according to the algorithm described in section 2.3.

However, our scheme fails when a compromised router generates a bogus SYN ACK packet in response to a SYN packet from a client. This can fool the access routers into forwarding packets through a compromised router. Because a misbehaving router can intercept and respond to any packets that go though it, we argue that this problem can only be addressed by user level verification and/or authentication. As an illustration of the problem, a compromised router can act as a web server. All user HTTP requests to `CNN.com` that go through this router will be intercepted and answered with bogus pages, such that only an intelligent being can detect the difference. Our system allows an access router's clients to inform it of bad routes. Our future research will address the problem of bogus complaints.

3.2 Scalability

There are three reasons why our system scales better than BGP. Route computation and propagation are removed from the critical path. The availability of the routing system does not depend on in-time computation of shortest paths any more. Although computation is still needed to incorporate updated information, it can be delayed. Because there are two routes to any destination, if one of them fails, packets can be forwarded using the other one. Since the two routes are as independent as possible, the probability of both of them failing is much lower than that of one.

The amount of routing messages is significantly lowered in our system, as a result of the separation of structural and dynamic information. Because the majority of them are not caused by structural changes on the Internet. Rather, they are the result of traffic fluctuation, routing policy changes and traffic engineering attempts [9, 22, 11]. These messages will not appear in our system.

Finally, the requirements upon transit routers are substantially reduced, since their resource requirement is no longer tied to the size of the Internet. Other than a small per-edge cost for storing structural information, transit router's resource requirement is independent of the number of distinct address prefixes and autonomous systems. In BGP, every router has to perform route computation, and possibly update its forwarding state, in response to routing updates. Our system requires this calculation only at the access routers, which can be incrementally updated, and have less demanding performance requirements. This keeps the core of the network independent from the costs of Internet growth such as more address prefixes, and from the requirement to provide aggregatable addressing by matching IP address structure to topology.

In order to understand the amount of information that an access router has to deal with, we processed the April 1st, 2002 BGP routing table dumps from the Route View Project [15]. We found that there are 13041 autonomous systems, 124330 prefixes, and 184557 links. Among the links, 57702 are inter-autonomous links. While these numbers might not accurately describe the topology of the Internet, they do give us a sense of the amount of computation and storage we are dealing with. Suppose we need 200 bytes to store information about a link, and 100 bytes for each autonomous system and prefixes, the total memory requirement is less than 50 megabytes.

The routing information we put into each packet is also minimal. Our analysis of the BGP routing tables showed that there are less than 5 autonomous systems between most prefixes. Therefore less than 24 bytes are needed for most packets. In comparison, IPv6 needs 32 additional bytes in the IP header.

4. APPLICATIONS

4.1 Defend Against Denial of Service Attacks

Denial of service attacks are becoming a serious threat to the availability of Internet connected networks and services. Although the attacks might not specifically target the routing infrastructure, we believe that the inter-domain routing system can be leveraged to help edge networks defend against DoS attacks.

The current defense against Denial of Service attacks involves manually contacting the upstream providers to install filters in their border routers to stop the flow of attack packets. This process is slow because sometimes several ISPs have to be contacted, and subject to social engineering (hackers impersonating edge network administrators). In this section, we propose an automated mechanism to install filters.

Once a denial of service attack has been detected, it is often feasible to recognize a pattern that differentiates the attack from normal traffic. The victim can then originate an update of the routing information for the link between his network and the upstream provider. The negative rules of this link would include the pattern that matches these DoS packets. A responsible upstream provider would then incorporate the rule into the access control list of its border routers, thus stopping the DoS packets from reaching the victim. Information about the link can be propagated further, and other autonomous systems on the way from the victim to the attacker can incorporate the rules to stop the attack traffic even earlier on.

4.2 Virtual Links with Zero Failover Time

Many Internet based applications require a reliable link between two remote locations. For example, if a surgery is performed at San Francisco, while the experts are located in New York City, then it is unacceptable if there is an connection outage. Although the current inter-domain routing system is designed to handle link failures, it can not deliver convergence time in the sub-second time scale [11]. Labovitz et al. [13] has shown that even with multi-homing, BGP's convergence time after a link failure is 3 minutes on average. And this number is increasing as the Internet grows bigger. Such a long period of no connectivity is unacceptable for mission critical real time applications. Therefore, the traditional solution to providing a link with fast fail-over property is by using leased lines. However, the cost of doing so is much higher than using a pure IP based approach.

In this section, we propose a mechanism to implement highly available virtual links with zero fail-over time based on the routing system we proposed. Obviously, if a site is connected to the Internet through only one provider, then it will lose connectivity with all other sites when that provider goes down. Therefore, a site wishing to have a reliable virtual link is best served with multiple upstream providers.

Suppose networks A and B want to establish such a virtual link, each of them should have a reliable gateway. In most cases, this would be the access router. We further assume that both A and B are multi-homed. Our research show that for most prefixes that are multi-homed, there exist two vertex disjoint routes between them, given the current Internet topology. Therefore, it can be assumed that the primary route and backup route are vertex disjoint. We further assume that the failure of these two routes are independent.

The gateways at A and B duplicates every packet that is sent to each other. Each packet is assigned a unique sequence number for duplicate detection. Or if there is an underlying IP level security protocol such as IPsec that can detect and eliminate duplicate packets, such a sequence number is not needed. The gateways are responsible for eliminating the duplicate packets.

If one of the routes fails, the virtual link continues to function. Since we assume independent failures of the two routes, the probability of a virtual link failure between A and B is π^2, where π is the probability of one route failing. After one route fails, the access router should perform a route computation, and select another route as the backup route. Therefore the period in which there is only one usable route from A to B is minimal.

This seems to be a waste of bandwidth. But bandwidth is virtually free and the network should be engineered such that if one link goes down, the surviving one can still fulfill the communication needs between A and B.

The assumption of failure independence of vertex disjoint routes can be invalid since there might be hidden dependencies among autonomous systems. For example, two autonomous systems might be operated by the same company. If the company files Chapter 7 bankruptcy, both AS's will go down. It is also possible that disjoint routes go through the same physical wire, or the same wiring closet. Our future research will try to address this problem.

5. RELATED WORK

Previous work on "byzantine robustness" [16] addressed many of the same security problems we discuss in this paper. There are three major differences between Perlman's proposal and ours. First we realized that the transit routers do not need to know about the network topology. This optimization makes the transit routers' resource requirement almost independent of the growth of the Internet. Second, we made the observation that most network traffic is TCP. Instead of inventing a separate network layer ACK, as proposed by Perlman, we used TCP SYN and SYN ACK packets as our end-to-end performance measurement. Finally, we are concerned about the scalability the routing system. Therefore all expensive computation are moved out of the critical path of the algorithm. In fact most computation can be done in parallel to packet forwarding.

Secure BGP [12] tries to address the routing security issues by adding signatures to route advertisements, so that reachability information can not be forged. However, BGP already suffers from slow response to network changes. Burdening it with further validation of peer-provided routing information threatens to slow it down further, if meaningful validation is in fact possible. Even if it can be done in a scalable manner, the existence of an authenticated route does not ensure that the route is actually operational. We believe the security problem in routing is a robustness problem, and can not be solved by authentication alone.

The Resilient Overlay Network [4] is a project that creates an overlay network that can get around routing failures. We believe overlay networks are not the final solution for reliable packet forwarding. The reason is simple. Overlay networks only increases the probability that the communication does not fail when there are only isolated routing failures in the network. No overlay network is going to function when the underlying routing infrastructure complete fails, for example, as the result of a black hole attack.

There are several companies, such as RouteScience [2], SockEye Networks [3] and netVmg [1], which are building products that can be used in an edge network to dynamically choose better paths, based on the end-to-end performance measurement. These products provide possible performance enhancement, and provide resilience to the failure of upstream providers. However, because they are edge-only solutions, they do not shield the customers from a large scale failure in the network. Also because they do not control the paths a packet travel after the packet enters the upstream ISP's network, they can not provide true redundancy.

6. CONCLUDING REMARKS

There are three key components of the routing system we propose. First, our system separates determination of dynamic performance information from structural information propagation. The former is determined locally with feedback based routing instead of receiving both from supposedly trusted peers, as in BGP.

Second, our system reduces routing in the backbone to purely structural information propagation. By removing route computation from the transit routers, and by removing route computation from the critical path of access routers, our system scales better than BGP.

Third, based on the structural information, an access router maintains more than one route to every destination. These redundant routes decrease the response time after a router or link failure.

The Internet has changed a lot from its early days. The trust among network operators no longer exists; the size of the Internet is growing exponentially; and many people have realized BGP is neither predictable nor stable. We believe a routing algorithm that depends on everyone behaving to function will no longer work well in such an environment. We also believe the solution to the scalability problem lies in giving the control of route selection to the edge networks and freeing the core routers from maintaining a complete model of the Internet. Our results to date suggest that feedback-based routing is a promising direction to consider. We plan to explore ways to incorporate this mechanism into the Internet as it evolves.

7. ACKNOWLEDGMENTS

The paper greatly benefited from the comments of the anonymous reviewers. Shelley Zhuang of UC Berkeley reviewed an early draft of the paper and provided much valuable feedback.

8. REFERENCES

[1] netvmg. http://www.netvmg.com.

[2] Routescience. http://www.routescience.com.

[3] Sockeye networks. http://www.sockeye.com.

[4] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris. Resilient overlay networks. In *Proceedings of ACM SOSP*, October 2001.

[5] R. Barrett, S. Haar, and R. Whitestone. Routing snafu causes internet outage. *Interactive Week*, April 25 1997.

[6] batz. Security issues affecting internet transit points and backbone providers. Blackhat Briefings, 1999.

[7] D. R. Cheriton and M. Gritter. Triad: A new next-generation internet architecture. http://www.dsg.stanford.edu/triad/triad.ps.gz, July, 2000.

[8] A. Collins. The detour framework for packet rerouting. http://www.cs.washington.edu/research/networking/detour/, November 1998.

[9] J. Cowie, A. Ogielski, B. Premore, and Y. Yuan. Global routing instabilities during code red ii and nimda worm propagation. http://www.renesys.com/projects/bgp_instability, September 2001.

[10] K. Houle and G. Weaver. Trends in denial of service attack technology. http://www.cert.org/archive/pdf/DoS_trends.pdf, October 2001.

[11] G. Huston. Commentary on inter-domain routing in the internet. RFC 3221, December 2001.

[12] S. Kent, C. Lynn, and K. Seo. Secure border gateway protocol, April 2000.

[13] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian. Delayed internet routing convergence. In *Proceedings of ACM SIGCOMM*, pages 175–187, 2000.

[14] C. D. Marsan. Faster 'net growth rate raises fears about routers. *Network World Fusion*, April 2 2001.

[15] D. Meyer. University of oregon route views project. http://antc.uoregon.edu/route-views/.

[16] R. Perlman. Network layer protocols with byzantine robustness. Technical Report 429, MIT Laboratory for Computer Science, 1988.

[17] J. Postel. Transmission control protocol, September 1981.

[18] Y. Rekhter and T. Li. A border gateway protocol (bgp-4), March 1995.

[19] S. Savage, T. Anderson, A. Aggarwal, D. Becker, N. Cardwell, A. Collins, E. Hoffman, J. Snell, A. Vahdat, G. Voelker, and J. Zahorjan. Detour: a case for informed internet routing and transport. *IEEE Micro*, 19(1):50–59, January 1999.

[20] S. Savage, N. Cardwell, and T. Anderson. The case for informed transport protocols. In *Proceedings of HotOS*, March 1999.

[21] S. Savage, A. Collins, E. Hoffman, J. Snell, and T. Anderson. The end-to-end effects of internet path selection. In *Proceedings of SIGCOMM*, pages 289–299, September 1999.

[22] A. Shaikh, L. Kalampoukas, R. Dube, and A. Varma. Routing stability in congested networks: Experimentation and analysis. In *Proceedings of SIGCOMM*, pages 163–174, 2000.

[23] K. Varadhan, R. Govindan, and D. Estrin. Persistent route oscillations in inter-domain routing. *Computer Networks*, 32(1):1–16, 2000.