

Challenges In Using Simulation to Explain Global Routing Instabilities

David M. Nicol
Dartmouth College

Dept. of Computer Science, and Institute for Security Technology Studies

Keywords : BGP, routing, high performance, instability, worm.

INTRODUCTION

The global Internet is comprised of a large number of autonomous networks that agree to cooperate. Each such network---called an Autonomous System, or AS---is entirely responsible for management and routing of all internet traffic within its own domain. The global Internet works though because ASes enter into business agreements to route traffic from one to another if traffic sources and destinations are in distinct ASes. At the level of AS, so-called border routers forward Internet traffic from one AS to another. agree to route traffic between themselves. When a border router processes a received packet, it looks at the addressing information on the packet, infers which AS the packet is ultimately destined for, looks up in a table the outgoing port it is presently using to redirect traffic for that AS, and sends the packet out that way. This action is called *forwarding*, and the data structures used to direct the forwarding are called forwarding tables. A packet may be forwarded through multiple routers on the way from the source to the destination AS, the sequence of routers visited is called its *path*. Two routers that are directly connected (at least logically) so that one may forward traffic directly to another are called *peers*. A distributed algorithm called the Border Gateway Protocol (BGP) is used among routers to construct the forwarding tables.

Forwarding tables can change dynamically, in response to changes in the local environment, e.g. if one of a router's peers stops responding. Most importantly, BGP allows a router to change its forwarding tables in response to changes *announced* by peers. In typical operation, a BGP router will learn of multiple routes to various ASes. The BGP protocol allows a router considerable latitude in deciding which route to advertise, among the potentially many routes it knows of. The decision policies used are driven by business relationships and decisions. For instance, ASes A and B may peer with each other through a contract that says that they will accept from each other only traffic that has A or B as the destination AS---neither is

willing to serve as a transit router. This means that no route A learns of from B (say to get to AS C) will ever be advertised by A, because A will never route traffic destined to C through B.

Intuitively, if a router announces a new way it has discovered of reaching a particular AS to its peers, one or more of those peers may determine that their new "best" way of reaching that AS is through the peer---and its new route---that just made the announcement. Therefore, a result of a route announcement made by one router may be route announcements by its peers. Likewise, a router may announce to its peers that it can no longer reach a given AS, this is called a route withdrawal. A router may withdraw a route to an AS if all of the routes it knows to that AS go through a peer with which it has lost communication. If router A has previously announced a route to C through B, which B later withdraws (or B appears to have left the infrastructure), A will search for the next best route to C through itself, and announce that. However, if B's departure leaves A with no known way of routing to C, A in turn must withdraw its announced route to C (rather than announce a new route to C).

The dynamic nature of forwarding table updates raises the specter of instabilities in the routing structure. "Route flapping" describes the phenomenon where the path taken by all packets from one AS to another constantly changes. This has deleterious effects on performance, because increased packet loss (and subsequent retransmission) are typical side effects of route flapping. More dramatic effects are possible as a result of routers dropping in and out of the infrastructure. For instance, if a router announces to its peers a route that is illegally formed, the BGP specification calls for those peers to "reset" themselves with respect to that peer, essentially cutting it out of the infrastructure. It may later come back on line again. BGP calls for peers to periodically notify each other with "keep alive" messages that serve as proof that a router is still operational. If a router becomes so overwhelmed that it misses a keep-alive deadline, its peers will figure it for lost and reset their sessions with it. It can of course later rejoin the infrastructure, potentially causing a flood of new route announcements as it does. These features create the possibility of portions of the routing infrastructure being

pushed into a non-stable alternating state of leaving and rejoining the infrastructure, possibly causing network partition as a result.

BGP routing is the glue that holds the Internet together. It is obviously important to understand the risks and conditions under which the routing infrastructure may become unstable. While routing based on BGP has held together now for nearly a decade, the dynamic growth of the Internet may be causing stress on the infrastructure. Recent events suggest that this is not an academic exercise.

CAUSES OF INSTABILITY

The fact that instability exists is well known. It is suspected that certain intrinsic features of BGP and the Internet contribute to instability. For example, Govindan and Reddy[3] examine impact that internet growth has had on stability. Labovitz et al. likewise look at network topology and the impact it has on instability[6]. There is evidence that BGP itself contributes to the possibility of instability. For example, a critical configuration parameter called MRAI (Minimum Route Advertisement Interval) limits the time between successive sets of route announcements out of a router. MRAI is in some sense the heart-beat of BGP, in that it defines the time-scale of announcement propagation. A very small value of MRAI increases the workload on a router, by increasing the frequency with which it devotes computational resources to analyzing routes and their announcements. A large value of MRAI avoids this problem, but makes BGP insensitive to changes. Either extreme can affect the “convergence time”, or average time needed for the route used by one router to a given AS to stabilize; long convergence times are one form of route instability. For years now the “standard” recommended value of MRAI has been 30 seconds. Recent experimental evidence by Griffin and Premore[4] suggests that this value is much too large. Labovitz et al. [5] conjecture that instability might also be related to BGP implementation, whether or not sender side loop detection is implemented (a router looks for loops in a proposed announcement, before making it) or withdrawal rate limiting is implemented (an MRAI-like parameter, but for withdrawal messages).

Instability can be induced from without as well as within. In July 2001 the Internet worm known as Code Red 2 (CRv2) spread across the globe. It exploited holes in the Microsoft IIS server to gain access and port itself to a machine. Once on-board it randomly chose IP addresses looking for other computers running IIS with the same access vulnerability, and continued the propagation. The intensity of the attack could be monitored as a function of time by sniffing network traffic in a given domain for scans

(i.e. a particular type of connection request) that were characteristic of the worm. Another worm, Nimda, struck in September 2001. It propagated in a similar fashion, using a larger arsenal of vulnerabilities and a faster means of spreading across the Internet. Temporal correlation between these two worm attacks routing instability was noticed by researchers at Renesys Corporation[2]. Figure 1 illustrates an example that plots the number of route withdrawals per 30 second time interval observed at one router, against the rate of “probes” generated by Nimda in the same time period, observed in one network. It turns out that this spike in withdrawals occurs in a wave, across routers situated all over the globe. Recall that a router withdraws a route to an AS only when it cannot reach that AS. The global wave shows that *somehow* the traffic induced by the worm caused a wave of distress among the routers.

Additional worm attacks can surely be anticipated in the future, with increased ferocity. The known correlation between such attacks and BGP instability makes the goal of understanding root causes of instability all the more urgent. In the case of the worms, the instability appears to be an unintended side-effect, induced somehow by the worm traffic.

The question is, why?

IN SEARCH OF EXPLANATIONS

Our group has for some time been working on developing simulation tools to aid in the search for understanding the behavior of large-scale systems. The SSF (Scalable Simulation Framework) API with bindings in Java and C++ have been a result, along with SSFNet, a public domain depository of network protocol models¹. SSFNet contains a sophisticated model of BGP, and all the infrastructure needed to simulate large networks of BGP routers. Our BGP models are getting substantial use in the networking community. We have ourselves used it to investigate the sensitivity of route convergence to BGP configuration parameters. We have a background of experience in large-scale network simulation, and the tools now to begin to answer the question----could hostile intent (via worms, via compromising routers, via traffic manipulation) bring the Internet down?

CHALLENGES IN SIMULATION

It is natural to use simulation to

¹ See www.ssfnet.org and www.cs.dartmouth.edu/research/DaSSF for downloads, tutorials, reference manuals, links to publications.

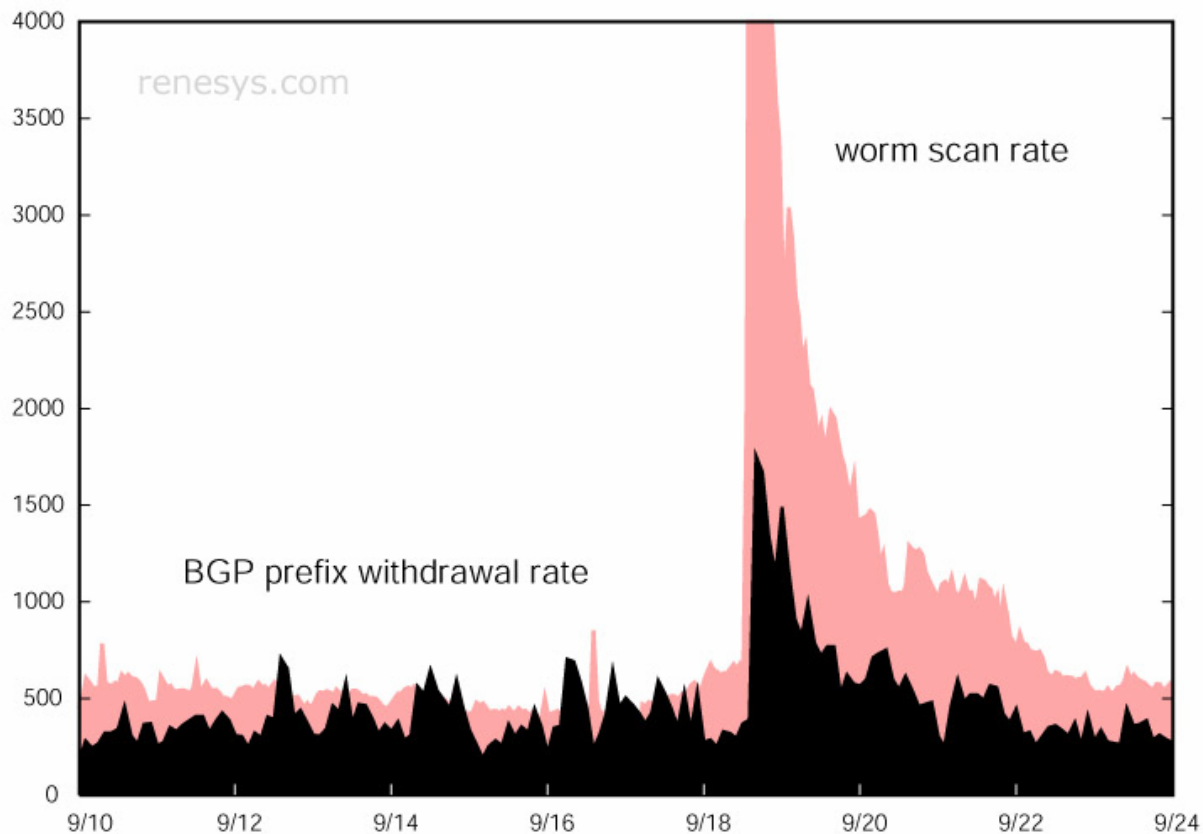


Figure 1. Correlation of BGP route withdrawals with Nimda scans

- try to explain what did happen when the worms struck, and what could happen again.
- Investigate how other traffic, maliciously shaped, might induce instability or even router failure.
- Investigate how one or more compromised routers could through cooperation announce routes that partition the network.

However, a number of challenges must be met.

Behavioral Objectives: One challenge is purely methodological. We need to have a clear notion of when a model we simulate “adequately” generates a routing storm or whatever other deleterious behavior we seek to induce or explain. In addition to simply exhibiting the observed behavior, the model needs to be able to explain clearly why the behavior is occurring.

Topology: Connectivity of ASes in the Internet is very highly variable. A handful of ASes connect with almost $\frac{1}{2}$ of all other ASes, most connect with just a very small

number of ASes, the rest range between these extremes. While there is some preliminary evidence that topological considerations affect instability, we don’t really know the degree to which topology contributes to deleterious behavior induced by traffic. The possibility exists that networks with thousands of routers will have to be simulated in order to generate causally realistic behavior. As we will shortly see, there are significant challenges to simply managing the memory demands of a simulation this large. One challenge is to circumvent the memory issue by designing a parametric means of generating topologies of various sizes, which still have the essential character (whatever that is!) of the larger network.

Model Size: A given router can in theory can be asked to route to any other AS in the entire Internet. There are on the order of 11,000 active ASes at the time of this writing. A representative of a major router vendor has told us privately that the aggregate storage associated with one route in a router requires on the order of 1000 bytes. A router capable of forwarding packets towards any one of

10,000 ASes requires on the order of 10Mb storage, just for the routes. A detailed simulation of 10,000 routers, each with this memory load, requires on the order of 10Gb of memory. This is not completely out of reach with 2002 technology, particularly if one uses large-scale parallel simulation, and/or out-of-core techniques. Ultimately one will need to validate the behavior of smaller models by comparison with as large a model as possible, against the possibility that small models miss emergent behavior which manifests itself only on the large model.

Traffic Models: Studies of instability induced by traffic will need appropriate traffic models. We can use CRv2 and Nimda as examples for propagation behavior (scanning IP address space.) The larger challenge will be to develop a family of infection models that allow us to explore the sensitivity of the routing infrastructure to infection behavior of potential new worms. Frightening scenarios are easy to imagine---e.g. a worm so cripples the system that patches to protect against it cannot be downloaded. A significant challenge is that a simulation of a major portion of the Internet cannot hope to simulate networks of major size on a packet-by-packet basis, existing models of network components and protocols are packet-oriented. We will need to develop a high level model that captures the diverse nature of worm traffic, as well as capturing the heavy-tailed behavior of normal traffic.

Router Behavior: The correlation between worm behavior and routing protocol behavior poses the question of whether the routers' responses are induced by the BGP protocol itself (e.g., as certain instabilities are induced by the choice of MRAI), by subtleties of implementation of BGP on routers, or by subtleties of packet forwarding *execution* on those routers. While we cannot hope to address dependence on implementation, we can explore how the BGP algorithm responds to dramatic changes in the nature of traffic. We can also model a router's software architecture (e.g., Cisco router software architecture is described in [1]), and use simulation to determine how a router's execution behavior responds to dramatic changes in traffic behavior.

BGP Policy: BGP is very flexible in allowing the owner of a router to craft policies to reflect the owner's business interests and contractual relationship with ASes with which it peers. These policies govern the priority of routes that are announced, and whether certain routes are even announced at all. One of our challenges is to determine the degree to which these policies contribute to instability. The intuition is that policies that are very restrictive limit the (logical) connectivity potential of the router infrastructure, and hence make it more prone to generate and propagate

route withdrawals in response to traffic that negatively affects router behavior.

Protocol Complexity: BGP is a very complex algorithm. We have already very detailed models of BGP, but we strongly suspect that for many of the experiments we would do, we don't need or want that much detail. Our challenge then is to model BGP in a way that adequately captures its sensitivity to route withdrawals and peer failures, but avoids the complexity of protocol elements that do not contribute directly to the behavior of interest.

Effect of Instability on the Internet: Ultimately we are interested in determining what risk routing instability imposes on the Internet. What threat is there of partition? What threat (and under what conditions) might portions of the routing infrastructure get into a mode where it oscillates between being up and being down. Once we have a simulation model that "works", a significant challenge is exploring the parameter space to find regions where very deleterious behavior is observed.

THE GRAND CHALLENGE

From the point of view of simulation and modeling methodology, there is nothing particularly unique about the problem of using simulation to explain the routing instability we observe. However, this particular modeling problem is representative of a whole class of modeling problems. To achieve our goal we will have to develop models of various components of the system at levels of abstraction that will allow us to evaluate sizeable systems. The models we build will have to focus on those features of the real system which most critically influence the behavior of interest---except that we don't know which features are critical. The models we build will have to have to be validated against data from systems which contain implementation choices, configurations, and bugs about which we'll never know. There will be a great deal of groping in the dark looking for explanations, and understanding. We have the tools for the task. We have for several years been developing the Scalable Simulation Framework for addressing modeling and simulation issues in complex communication networks. That toolset provides us with an excellent starting point for our investigation.

A tremendous amount rides on our ability to solve the puzzle, determine what makes routing instable, determine what risks the Internet is exposed to because of instability, and engineer solutions to the problems exposed by the simulation based analysis. This is a grand challenge worth taking on.

References

- [1] V. Bollapragada, C. Murphy, R. White, *Inside Cisco IOS Software Architecture*, Cisco Press, 2000.
- [2] J. Cowie, A. Ogielski, B. Premore, Y. Yuan
"Global Routing Instabilities during Code Red II and Nimda Worm Propagation."
[{11/20/01}](http://www.renesys.com/projects/bgp_instability) }.
- [3] R. Govindan and A. Reddy. An analysis of Internet inter-domain topology and route stability. In *Proceedings of IEEE INFOCOM*, April 1997.
- [4] T. Griffin and B. Premore. An Experimental Analysis of BGP Convergence Time. In *Proceedings of the 9th International Conference on Network Protocols*, November 2001.
- [5] C. Labovitz, A. Ahuja, A. Bose and F. Jahanian. Delayed Internet routing convergence. In *Proceedings of ACM SIGCOMM*, August/September 2000.
- [6] C. Labovitz, R. Wattenhofer, S. Venkatachary, and A. Ahuja. The impact of Internet policy and topology on delayed routing convergence. In *Proceedings of IEEE INFOCOM*, April 2001.

Acknowledgements

We appreciate the comments made on drafts of this paper by Brian Premore, Yougu Yuan, Andy Ogielski, and Jim Cowie. Figure 1 is included by permission of Renesys Corporation. The research reported here is supported in part by DARPA Contract N66001-96-C-8530, NSF Grant ANI-98 08964, NSF Grant EIA-98-02068, and Dept. of Justice contract 2000-CX-K001.

Biography

David M. Nicol is Professor and Chair of the Department of Computer Science at Dartmouth College, and a Principle Investigator at the Institute for Security Technology Studies. His research interests lie in high performance simulation, networking, and security. He is Editor-in-Chief of ACM Transactions on Modeling and Computer Simulation. He received the B.A. in mathematics from Carleton College in 1979, and the Ph.D. in computer science from the University of Virginia in 1985. See www.cs.dartmouth.edu/~nicol for other details.