Verifying the Correctness of Wide-Area Internet Routing

Nick Feamster and Hari Balakrishnan
{feamster,hari}@csail.mit.edu

1. Motivation

Several studies have shown that wide-area Internet routing is error-prone, with failures occurring for a variety of reasons. Routing fragility is largely due to the flexible and powerful ways that BGP can be configured to perform various tasks, which range from implementing the policies of commercial relationships to configuring backup paths. Configuring routers in an AS is like writing a distributed program, and BGP's flexible configuration and today's relatively low-level configuration languages make the process error-prone. The primary method used by operators to determine whether their complex configurations are correct is to try them out in operation.

Despite the need for tools and techniques for verifying BGP's correctness, there are significant challenges that have thus far prevented verification techniques from becoming used in practice. First, defining what it means for BGP to be "correct" is not easy, because it is hard to define a "specification" for an operational BGP—its many modes of operation and many tunable parameters allow for a great deal of flexibility that is hard to specify. Additionally, BGP's configuration is distributed across many routers, and precisely defining how various aspects of BGP's configuration interact is challenging.

2. Approach

There are at least three ways to improve this state of affairs. The first is to argue that BGP4 has outlived its purpose and to develop a new routing protocol. Of course, that protocol would have to be at least as flexible as BGP4, while being less error-prone. Unfortunately, it is not immediately obvious what we should change in BGP. A second approach would be to argue that errors arise because today's configuration languages are too "low-level" and are not well-designed, leading to programming errors. Again, it is not obvious what specific improvements should be made to configuration languages. A third approach, complementary to the previous two, would be to develop a framework for *analyzing* router configurations prior to deploying them.

Our work takes the third approach and presents several contributions. First, using the *routing logic* [1], we derive both correctness constraints for BGP configuration and conditions under which BGP will (1) originate incorrect routes, (2) propagate incorrect routes, (3) fail to propagate routes when it should, (4) violate intended high-level policy, and (5) exhibit nondeterministic behavior. Second, we use these constraints to design and implement **rcc** ("router configuration checker"), a tool that analyzes router configurations and detects anomalies. rcc can help network operators debug their complex BGP configurations and correct errors *before* deploying them. rcc has been downloaded by 30 network operators to date. Third, we use rcc to find errors in real-world configurations and present the findings of our experimental analysis. This analysis helps us understand *why* routing problems occur and determine whether each problem is due to weaknesses in BGP or problems in specifying configuration. Finally, given an understanding of why configuration errors occur, we recommend specific changes both to BGP and to configuration languages. As the protocols and configuration languages evolve, the ability to detect and fix errors in configuration before deployment will be invaluable to network operators.

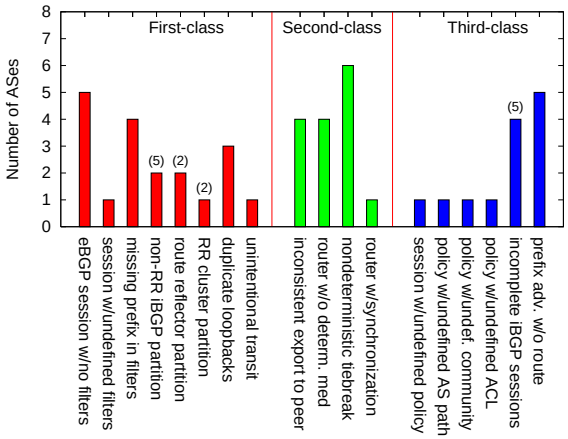

Figure 1: ASes in which each type of error or anomaly occurred at least once. Unless noted in parentheses, each test was run on all 9 ASes.

3. Results

We have used rcc to check BGP configurations from 9 operational networks, testing nearly 700 real-world router configurations in the process. rcc found errors in every network. In most cases, the operators were unaware of these errors. We uncovered many serious errors, including the potential for network partitions caused by route propagation problems, propagation of invalid routes (usually due to improper or non-existent route filtering), and routers forwarding packets in ways that were inconsistent with high-level policy.

Figure 1 summarizes the errors we found with rcc, classified according to three classes of seriousness (first-class errors are the most serious). Most of the errors occurred in more than one AS. Because we used rcc to test configurations that were *already deployed* in live networks, we did not expect rcc to find the types of transient misconfigurations that quickly become apparent to operators when the configuration is deployed. Operators could apply rcc to router configurations *before* deployment to prevent these types of errors.

Although there are many reasons for configuration errors, three reasons explain most errors. First, many errors arise from the complex, obscure mechanisms for propagating routes learned from BGP border routers within a network. Second, even simple policy specifications (*e.g.*, treating a route as a backup) are specified using multiple levels of indirection in configuration files, making mistakes more likely. Finally, many errors reflect the fact that operators have no systematic process for configuring their networks; many of the errors we found could be fixed with better configuration management tools.

4. References

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