ISSN NO: 9726-001X
Volume 10 Issue 04 Dec2022

JOURNAL OF CURRENT SCIENCE

The ability to test and monitor large network systems in a scalable, software-based manner

A.krishna Priya
Northern University Bangladesh, Dhaka-1000 - Bangladesh,

ABSTRACT
Despite the growing size and complexity of today's networks, many administrators still use outdated methods like and to troubleshoot issues. We propose a method termed Automatic Test Packet Generation (ATPG) for testing and debugging networks that is both automated and systematic. A device-independent model is created by ATPG after reading router settings. The model is then used to produce a set of test packets large enough to (at a minimum) exercise every link in the network or (at a maximum) exercise every rule in the network. Periodically, test packets are sent, and if a failure is identified, a separate system is activated to pinpoint the source of the problem. Previous work in static checking (which cannot discover liveness or performance defects) and fault localization (which can only locate errors given liveness findings) are complemented by ATPG, but ATPG goes beyond. In this paper, we present the design, development, and testing of an ATPG prototype on two real-world data sets: the Stanford University backbone network and Internet2.As an example, 4000 packets are sufficient to cover all rules in the Stanford backbone network, whereas just 54 are needed to cover all connections. Less than one percent of network capacity is used up when 4000 test packets are sent at 10 per second. The ATPG datasets and source code are freely accessible to the public.

Data plane analysis, test packet creation, and network troubleshooting are some related concepts

INTRODUCTION

Problems with a network may originate from a wide variety of causes, such as misconfigured routers, damaged fiber, mislabeled connections, faulty interfaces, flawed software, sporadic communications, and more. When troubleshooting a network, engineers often resort to just the most basic tools (ping, traceroute, SNMP, etc.) before resorting to a combination of expertise and intuition to pinpoint the origins of the problems they find. The difficulty of troubleshooting a network rises in tandem with its scale and complexity; a modern data center may contain 10,000 switches, while a university network may support 50,000 users and a 100-Gb/s long-haul.

Because it tests all forwarding entries, firewall rules, and packet processing rules independently, ATPG is able to detect and correct network-wide errors. To ensure that all use scenarios are covered, ATPG uses the device configuration files and FIBs to generate an algorithm that generates a fixed amount of test packets. Whether you need to create test packets from every port in your network or can only access those ports via specific routers, ATPG can conform to your demands. It's possible that ATPG will allocate additional test packets so that more crucial rules may be put to the test.

EXISTING SYSTEM

We are not aware of any previously existing techniques for mechanically manufacturing test packets from setups. The closest we have found to such efforts are offline tools that investigate network invariants. NICE aims to symbolically cover all potential controller application code pathways using abstracted representations of switches and hosts. Anteater checks invariants against configurations by modeling them as boolean satisfiability problems on the data plane and then using a SAT solver to find the optimal

ISSN NO: 9726-001X
Volume 10 Issue 04 Dec2022

solution..

Previous systems work demonstrates the extensive usage of end-to-end probes for network failure diagnostics. Recently, there has been a rise in interest in mining imperfect, unstructured data such router settings and network tickets. Using router configuration and data plane information, ATPG fine-tunes detection granularity to the rule level.

PROPOSED SYSTEM

Our proposed system, an Automatic Test Packet Generation (ATPG) framework, generates a limited number of packets on its own to verify the health of the underlying topology and the conformance of the data plane to the configuration. The tool may also be able to automatically generate packets, which can then be used to test performance claims like "packet delay."

Each rule may be tested on the data plane by introducing test packets into the network. Because ATPG examines connections in the same way it examines other forwarding rules, it tests every connection in the network. However, it has been proved that all-pairs ping is not scalable for really large networks like Planet Lab, therefore it cannot guarantee that all connections are tried.

CONCLUSION

Testing the availability of a network is a fundamental difficulty for Internet service providers and operators of large data centers. Sending probes between every feasible pair of edge ports is neither comprehensive nor scalable. But ATPG does a lot more than just verify existence on the same base. Our method also uses the header space structure to provide a fundamental defect localization technique, which aids in the testing process. We believe that network ATPG, when used for automated dynamic testing of production networks, will be just as useful..

REFERENCES

[1] This is the ATPG code repository [1] (Online).

[2] [2]See [2] http://eastzone.github.com/atpg/ for further details.(2013) [Internet] Generating Test Patterns Automatically.Wikipedia hosts Automatic Test Patterns for the Next Generation.

[3] "Network performance anomaly diagnosis and localisation," by P. Barford, N. Duffield, A. Ron, and J. Sommers; Proc. IEEE INFOCOM, April, pages 1377-1385.