ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

A Survey on Generation of Automatic Test Packet for Fault Localisation

Anusha Rao, Sonal Fatangare

ME Student, Dept. of Computer Engineering, RMD SSOE, Warje, Pune, India

Assistant Professor, Dept. of Computer Engineering, RMD SSOE, Warje, Pune, India

ABSTRACT: Troubleshooting a chain of networks is getting to be harder for organizations due to basic instruments like ping and trace route so a system is implemented to test and debug the network path by generating test packets to act upon every connection. This approach also detects functional problems and generates packets in an automatic way to evaluate performance and sets the guidelines in processing the packet between nodes. Special mechanisms are used to detect flaws by sending test packets periodically. While transferring data, if there is any physical or software problems with the path, it leads to loss of data. With the help of the proposed system, there will be no loss of data and can minimize the cost of transferring. The proposed system makes use of Header Space Analysis framework. A Test packet selection algorithm is used for computing a small number of test packets which is used to test the entire packet processing rules and also Fault localization algorithm is useful for pointing out the faulty rules and perform the end-to-end testing in the network path.

KEYWORDS: Packet generation, fault localization, data plane analysis, network troubleshooting

I. INTRODUCTION

Operating a current modernized network is not an easy task. Every day engineers strive hard with the links which are misconfigured, software bugs and endless other bounds that make the network work improperly or lead to network breakdown. Administrators contribute compelling effort in ensuring that the network meets their intended policies. While the recent effort on checking accuracy, reliability and meeting such type of policies have become a huge struggling task. Network operators depend on the basic tools such as ping, traceroute etc. Debugging the networks is getting harder as networks are becoming greater in number (a current server center accompanying 15,000 switches, a college grid serving 50,000 to 1 lakh users) and becoming more sophisticated.

For some great reasons, troubleshooting is a burdensome work. The primary reason is due to forwarding state is distributed over different switches and firewalls and is determined by forwarding tables and other composition parameters. The second reason is, forwarding state is very difficult to examine because it typically requires manual way of logging into each and every box in the network via Command Line Interface (CLI). The third reason is, there are various programs, convention sets, rules and individuals modernizing the forwarding state concurrently which makes the debugging a complex task. Because of these issues, network engineers deserve better tools other than ping and traceroute. To address this challenge, the proposed system performs an automated process in a standardized way to detect bugs and faults among the networks by generating test packets to act on each and every link in the network and also on every rule in the network. Need of automatic test packet generation are as follows:

- To test and debug the network path in a computerized and standardized manner by generating test packets.
- To act upon every connection and also on every network processing rule in the network.
- This approach also detects functional problems and generates packets in an automatic way to evaluate performance and sets the guidelines in processing the packet between nodes.
- To avoid loss of data and minimize the cost of transferring.
- It is also required for computing a small number of test packets which is used to test all the packet processing rules and for pointing out the faulty rules and perform the end-to-end testing in the network path.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

II. RELATED WORK

There are several systems proposed for automatic generation of test packets.

Offline Tools Supporting Automatic Test Packet Generation

One of the offline tools which have been used for generating test packets automatically in control plane is NICE. NICE stands for "no bugs in controller execution". The bugs brought by NICE in controller program to user's notice more well by means of the assistance of model checking and symbolic execution in open flow system. Working of NICE is shown in Figure 2. NICE programmer has to provide controller program along with topology of network which consist of requirement of switches and hosts. The NICE according to fixed plan looks into the probable system behavior and checks it with exactness properties supplied by the programmer. The traces of property abuse is provided by NICE or properties that are not up to the mark by means of their indications as output.

Header Space Analysis

Header space analysis framework is used by the automatic test packet generation, in which it uses a geometric replica, which allows the ATPG system to verify the network specifications and configurations to compact with important classes of failures such as traffic isolation, forwarding loops, linkage problem and reachability failures. Header space analysis is also capable to do slicing. Slicing assures isolation between system hosts, users or traffic. Each slice has the separate control plane, and it is up to its owner to decide how packets are routed and processed in that slice.

Overview of the system

a) Origination of Test Packets
b) Error Fixing
1) The ATPG system begins by gathering the forwarding state from network.
2) The header space analysis is used by ATPG system to figure out scope of each terminal.
3) The outcome of second step is taken as input by test packet generation algorithm to gauge smallest number of test packets sufficient to test all rules.
4) These test packets are sent regularly by test terminals as a penultimate step.
5) Lastly, if an error is disclosed ATPG appeals to fault localization algorithm to curtail root of error.

a) Origination of Test Packets

1) ATPG system begins by estimating the entire set of test packet headers that can be forwarded from each test terminal to every other test
2) Afterwards, ATPG selects greater than or equal to one test packet from identical class of test packets to use every rule which is within reachable distance. This method is capable of finding only those faults for which all packets screened by same rule suffer the same fault.
3) Lastly in the process of generating test packets ATPG goes to compression.

b) Error Fixing

ATPG sends the set of test packets at regular intervals. If in case test packets fail to reach their desired target, ATPG is capable of identifying errors that induced the problem. If watched performance of a rule is different from its normal behaviour then a rule is neglected, in other words it fails. ATPG monitors that where rules fail by applying a result function R on rule r in a packet pk. A result function takes value 1 if packet pk follows rule r, if not it takes value 0. A forwarding of a rule fails if a test packet is not provided to its planned output port on the other extreme. Forwarding of a rule is successful if either a test packet is provided to its planned output port, or in case it is a drop rule, it is addressed rightly if it is dropped. A link collapse can be characterized by failure of forwarding rule in topology function.

III. SURVEY OF AUTOMATIC TEST PACKET

In most of the cases in the existing works, the admin used to manually decide which ping packets to send. Here, the current system methodologies designed can avert errors in software logic, however, neglect to detect the errors caused

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

by fizzled connections and switches. Testing liveness of a network is a fundamental problem for ISPs and large data center operators. Every pair of edge ports which receives probes is neither exhaustive nor scalable. It finds a least set of end-to-end packets that pass through each link. However, doing this requires a way of abstracting transversely device specific configuration files, generating headers and the links they reach, and finally determining a minimum set of test packets (Min-Set-Cover) which check enforcing consistency between policy and the configuration.

Working in a static manner is the feature which can be seen in existing network debugging tools. These methodologies are not intended to diagnose failures affected due to damaged routers architecture, or problems generated by congestion in network operational performance. With these approaches, one can prevent logical errors occurred in control plane software. Researchers have proposed tools which enforce consistency between devise configuration and policy.

In 2001, N. Duffield, F. L. Presti, V. Paxson, and D. Towsley[1], explored the use of end-to-end unicast traffic as measurement probes to infer link-level loss rates. Experiments were designed based on the notion of transmitting stripes of packets (with no delay between transmissions of successive packets within a stripe) to two or more receivers. The purpose of stripes is to make certain the correlation in the observations matches of receiver and one when it was replaced by the notional multicast probe. The stripes which are longer can be considered together with the correlation by considering measurements which provide good facts that a packet pair to differentiate receivers.

In 2001, N. G. Duffield and M. Grossglauser[2], proposed a method for all the packets traversing through the network that allows the direct proposal of traffic flows in the course of a domain by observing the trajectories of a subset of all packets. The key advantages of the method are that (i) it does not rely on routing state, (ii) cost of implementation is less, and (iii) the measurement reporting traffic is modest and can be controlled precisely. Sampling of packets is the key idea based on a hash function computed over the packet content. The same sample set will be yielded using the same hash function of packets in the entire domain, and enables to reconstruct packet trajectories.

In 2006, Y. Bejerano and R. Rastogi[3], developed a failure-resilient techniques for monitoring link delays and faults in a Enterprise IP network or Service Provider. Our two-phased stategy tries to minimize both the monitoring infrastructure costs also the additional traffic owing to probe messages. In the first phase approach, computation of the locations of a nominal set of monitoring stations such that all network links are enclosed, yet in the presence of several link failures. Subsequently, in the second phase, the station transmit computation of a minimal set of probe messages to measure link delays and isolate network faults are done.

In 2010, D. Turner, K. Levchenko, A. C. Snoeren, and S. Savage, propose a methodology which is low quality data sources gathered in production networks syslog, operational mailing lists router configs is combined to reconstruct topology, failure causes and dynamic state. This is a grouping of structured and semi-structured data used to analyze failure events in a large regional network. Structured data consists of router configurations and syslogs whereas semi-structured data consists of email logs.

IV. BACKGROUND

Network Problem

There are many problems in the network. Here are the top most issues that are affecting today's large computer networks.

- Performance Degradation
  Performance Degradation is related to issues like loss of speed, integrity of data due to weak communications. Complex networks are affected due to the additional partitions, endpoints, and additional equipment at midpoints.

- Identification of Host
  Configuration of proper network is key for maintaining the proper host ID. Generally, the mail station can't convey messages without proper addressing, and without proper networking equipment. While simpler networks can easily be arranged by addressing the configuration in a manual way, coming to the complex systems this way of addressing becomes totally unrealistic. One can make expansive and versatile network by taking help of domain controllers, DHCP servers and their essential addressing software and protocols.
- Issues in security
  Maintaining uprightness in the system, blocking unauthorized users from penetrating the system and ensuring the system foreswearing of administration assaults should be followed in order to maintain security.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

Popular Techniques

For every network administrator, network troubleshooting tools are a necessity. It is very hard to detect performance problems and end-to-end fault detection in wide area networks due to increase in the complexity of paths and network dependency. To diagnose and monitor different network metrics and collect monitoring information from a vast number of hosts around the globe, there are several monitoring infrastructures. Here there is description about tools which are used to troubleshoot common networking issues:

Ping: It is one of the common tools in networking, which informs about connections by conducting a basic test between hosts which are requesting and responding. Providing the key idea of the area that is affected in the network is the useful feature in this tool.

Traceroute: Once the key idea of the affected area is provided by the ping tool, traceroute is utilized in order to produce more particular information between hosts and the destination path.

Ipconfig/ifconfig: This tool finds out the particular IP configuration of infected hosts. If addressing is static configuration, this information is already known, but if the addressing is dynamic configuration, all hosts IP addresses can change frequently.

Putty/Tera Term: A Secure Shell or telnet is needed at the point when interfacing with various kinds of equipment; when all these are obliged both the Putty and Tera Term projects have the capacity to give these functionalities.

Speedtest.net/pingtest.net: It is used to test the quality of connection of the Internet. The bandwidth available at a particular time to a specific host can also be known by this tool.

Besides this tools, there are numerous tools ranging from basic model to complex configured systems which are useful for detecting anomalous behavior. All these different types of tools serve for specific purposes and satisfy the user needs. The admin goes through different stages in monitoring the network. The administrator implements the network performance tool, the tools verifies every link in the network and monitors all the paths. If there are any faults in the path, it reports the administrator by different modes like alarming, signaling, etc. All the reports are stored in the database; the operator can rectify the faults which cause inconvenience in the operational network.

V. RESEARCH ELABORATION

The test packets are generated by the system to act upon all the links in the network that covers all the forwarding rules covered and exercised by a minimum of one packet to test. Automatic Test Packet Generation (ATPG) framework that automatically generates a nominal set of packets to test the liveness of the fundamental topology and the similarity among data plane state and configuration specifications. The tool can also automatically generate packets to test performance assertions such as packet latency and loss with test packet. It can also be specialized to generate a minimal set of packets that simply test each link for network liveness. It consists of modules such as:

- Test Packet Generation
- All pair reachability table
- Fault Localisation

Two key contrains are available while generating test packets which must be respected by ATPG. They are:
1) Port: ATPG must only use test terminals that are available;
2) Header: ATPG must only use headers that each test terminal is permitted to send.

Using Test Packet Selection (TPS) algorithm, ATPG selects test packets. All equivalent classes present between each pair of available ports is found by TPS. TPS selects test packets by sampling each class and then find the minimum covering set by compressing the resulting set of test packets.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

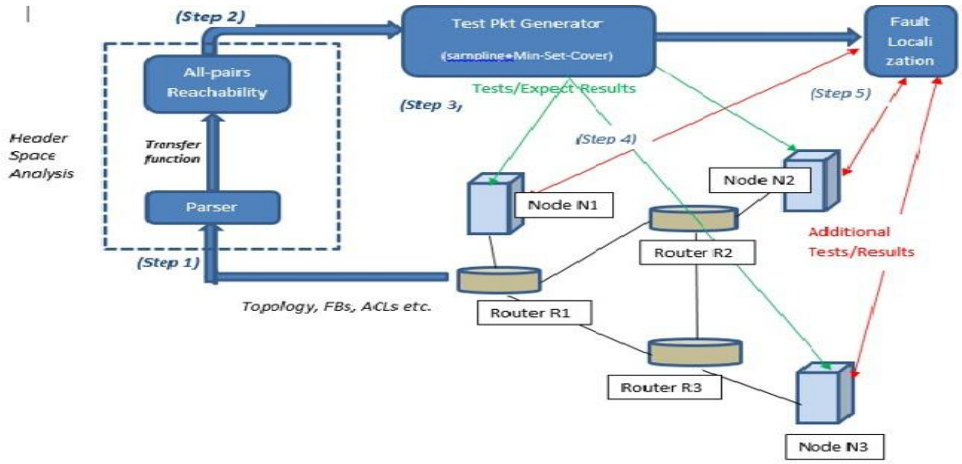

Fig.1. System Architecture

- **Test Packet Generation**

Let us assume that the sending and receiving of test packets is occurring in between a set of test terminals. Generating least number of "test packets" to execute all the packet processing rules in every node is the main aim of the proposed work. The system must respect two key constraints, while generating the test packets. 1) Port - The tool should only make use of available test terminals, 2) Header - The tool uses the headers for the purpose of granting send permission to every testing terminal.

Test Packet Selection:

The switching functions T1,..Tn, and network topology function Γ, the Test Packet Selection (TPS) algorithm can be used for selecting the test packets to exercise all packet processing rules which can be reachable, subject to the header and port constraints in a network. In between each and every pair of available nodes, Test Packet Selection algorithm (TPS) is used for finding the equivalent set of classes.

- **Generating table with all-pair reachable conditions**

While the packet is transferred in between one test terminal to another test terminal, the tool starts working by calculating the total set of packet headers in the first step. Then it computes an entire rule sets so that it exercises along the path from each such packet header. It takes use of all-pair reachability algorithm described in order to find the possible pairs.

Sampling:

In the next step, the tool selects minimum of a test packet to act upon all the reachability rules. Selecting a single packet per class in some random manner is the simplest mechanism.

Compression:

In this step, the packets are compressed to minimum number. So, the system selects a minimum set of packets from the sampling step, such that it covers all the rules of the union of rule histories. The cover is selected in such a way that all connection or all router configurations are covered. This is known as Min-Set-Cover problem. While NP-Hard, greedy O(N2) gives a decent approximation, here N represents the test packets. The obtained minimum set of test packets is referred as regular test packets fp1,p2, p3, p4, p5g and the remaining packets which are not selected comes under reserved test packets fp6g.

- **Localization of faults**

After sending the minimum set of test packets across the node, it tests for failures. If any faults are present in the path, the system diagnosis the defects. A set of test packets are sent periodically bt ATPG. If test packets fail, ATPG pinpoints the fault(s) that caused the problem. A rule fails if its experimental behavior differs from its expected

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

behavior. ATPG keeps track of where rules fail using a result function ”Success” and ”failure” depend on the nature of the rule: A forwarding rule fails if a test packet is not delivered to the proposed output port, where a drop rule behaves properly when packets are dropped. Similarly, a link failure is a breakdown of a forwarding rule in the topology function. On the other hand, if an output link is congested, failure is obtained through the latency of a test packet going above a threshold.

VI. DISCUSSION

The complex nature of modern networking system, the hierarchy among the components and security levels in the system become more sophisticated, which make the administrator difficult in understanding the complex interactions between components behavior. This lead to the difficulties in the prediction and control of faults in the network. Packet forwarding in advanced systems is a complex methodology, including mutually dependent functions running on large number of devices. The Existing tools are inadequate since they take timescales of seconds to hours, since they operate offline and check network configuration files and the data-plane state. The concept of evolving an automated test packet generation device prompts inspiration to work towards developing up this project.

VII. CONCLUSION

Today’s systems require an abundant human interaction to keep them working. As systems get complex there is colossal enthusiasm for robotizing the control, error reporting, investigating and trouble shooting. For the operators with huge information centers and Inter Service Providers, testing the network health is a very cumbersome task, since the ISP’s should offer their customers more services rapidly without any interruption. The developed system can test the liveliness of the network by automatically generating the minimum set of test packets (Min-Set-Cover by using union of rule histories). It reads information about forwarding tables from all the routers and builds up a device independent model based on Header-Space Analysis.

Automatic Test Packet Generation type tools are in dire need in checking liveliness of the network. The developed tool works in dynamic way of checking liveliness by testing all the rules, thereby achieving the reachable policies. In case of any faults during the information exchange, a fault localization algorithm is used for detecting the failed links across the nodes in the system. The execution likewise expanded testing with a straightforward fault localization algorithm which is additionally developed by utilizing the header space structure. As in programming testing, the formal model assisted to increase the testing analysis thereby decreasing the number of test packets. The outcome generated displays that every link in the network can be operated with limited amount of packets [8]. One can trust the developed project will be just as helpful for robotized element testing of complex systems since it can minimize the data loss by sending the original message after checking the network links and devices with test packets.

REFERENCES

1. Hongyi Zeng, Peyman Kazemian,George Varghese,and Nick McKeown,“Automatic Test Packet Generation”,VOL. 22, NO. 2, APRIL 2014.
2. Y. Bejerano and R. Rastogi, “Robust monitoring of link delays and faults in IP networks,” IEEE/ACM Trans. Netw., vol. 14, no. 5, pp.1092–1103, Oct. 2006.
3. N. Duffield, F. L. Presti, V. Paxson, and D. Towsley, “Inferring link loss using striped unicast probes,” in Proc. IEEE INFOCOM, 2001, vol. 2, pp. 915–923.
4. N. G. Duffield and M. Grossglauser, “Trajectory sampling for direct traffic observation,” IEEE/ACM Trans. Netw., vol. 9, no. 3, pp.280–292, Jun. 2001.
5. “Automatic Test Pattern Generation,” 2013 [Online]. Available:http://en.wikipedia.org/wiki/Automatic test pattern generation.
6. Troubleshooting the network survey, 2012 [Online]. Available: http://eastzone.github.com/atpg/docs/NetDebugSurvey.pdf
7. D. Turner, K. Levchenko, A. C. Snoeren, and S. Savage, California fault lines: Understanding the causes and impact of network failures, Comput. Commun. Rev., vol. 41, no. 4, pp. 315326, Aug. 2010.
8. “Automatic Test Pattern Generation,” 2013 [Online]. Available: http://en.wikipedia.org/wiki/Automatic_test_pattern_generation
9. P. Gill, N. Jain, and N. Nagappan, “Understanding network failures in data centers: Measurement, analysis, and implications,” in *Proc. ACM SIGCOMM*, 2011, pp. 350–361.
10. R. R. Kompella, J. Yates, A. Greenberg, and A. C. Snoeren, “IP fault localization via risk modeling,” in *Proc. NSDI*, Berkeley, CA, USA, 2005, vol. 2, pp. 57–70.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 12, December 2015

BIOGRAPHY

Anusha P. Rao is a Master of Engineering student in the Computer Engineering Department, RMD Sinhgad School of Engineering, Savitribai Phule Pune University. She received Bachelor of Engineering degree in 2012 from DPCOE, Wagholi, Pune. Her research interests are Image Processing, Data Mining, Network security etc.

Sonal Fatangare is an Assistant Professor in the Computer Engineering Department, RMD Sinhgad School of Engineering, Savitribai Phule Pune University. She received Master of Engineering degree in 2014 from JSPM, BSIOTR, Wagholi, Pune. Her research interests are Data Mining and Network security.