ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Solution for Premature Entries Deficit in IP Networks

X.Nancy*, S.Selvanayaki
*M.E Student, Assistant Professor, Vel Tech Multi Tech Dr. Rangarajan Dr. Sakunthala Engineering College, Chennai, Tamil Nadu, India

Abstracts

The Internet is to connect multiple computer networks for linking many devices. Service disruptions occur in many networks through link and node failures. Therefore, for each service providers, the major challenge is to provide service without any interruption. For recovering IP networks from multiple failures, Localized On-demand Link State (LOLS) routing can be used. In LOLS, a blacklist is carried along with the packet. The blacklist consists of set of links that are failed along the path. Based on the destination and blacklist, the next hop can be found and the blacklist is reset whenever the packet moves forward to destination. The main contribution of this system is to handle the dual link and detect the single node failures.

Keywords: Blacklist, Localized On-demand Link State routing

Introduction

Aim of the project

A Solution for premature entries deficit in IP networks is to detect the link and node failures. This System is to handle the dual link, and handle the single node failures. Then mainly handle the another failure is Failure Carrying Packets (FCP) and Packet Recycle tries to forward packets to reachable destinations even in case of arbitrary number of failures.

Overview of the project

The Internet is increasingly being used for mission critical applications and it is expected to be always available. Unfortunately, service disruptions happen even in well-managed networks due to link and node failures. There have been some studies [11][3] on frequency, duration, and type of failures in an IP backbone network. [2] reported that failures are fairly common and most of them are transient: 46% last less than a minute and 86% last less than ten minutes. To support emerging time-sensitive applications in today's Internet, these networks need to survive failures with minimal service disruption. For example, a disruption time of longer than 50 ms is considered intolerable for mission-critical applications [4]. Therefore, providing uninterrupted service availability despite *transient* failures is a major challenge for service providers.

While a majority of the failures were observed to be single failures, one study [2] has found that approximately 30% of unplanned failures (which constitute 80% of all failures) involve multiple links, which is a significant fraction that needs to be addressed. Moreover, the extent of service disruption caused by multiple failures can be quite significant. Hence, it is important to devise schemes that protect the network against not only single failures but also *multiple independent failures*. Our work is motivated by this need, which is also the focus of some of the recently proposed routing schemes [5]–[7].

The commonly deployed link state routing protocols such as OSPF and ISIS are designed to route around failed links but they lack the resiliency needed to support high availability [11]. The remedies suggested in [8], [9] can achieve convergence in less than one second. However, bringing it down below the 50ms threshold runs the risk of introducing routing instability due to hot-potato routing, which can cause relatively small internal link-state changes to trigger a large churn of external routes [10]. MPLS [16] can handle transient failures effectively with its label stacking capability. However, we argue that it is not scalable to configure many backup label switched paths for protection against various combinations of multiple independent failures. In [12], authors attempt to make MPLS based recovery scalable to multiple failures, but assume that probable failure patterns based on past statistics on the network failures are known to the MPLS control plane.

There have been several fast reroute proposals for handling transient failures in IP networks by having the adjacent nodes perform local rerouting without notifying the whole network about a failure [13]–[14]. However, most of these schemes are designed to deal with single or correlated failures only. Recently, [7] proposed an approach to handle dual link, but only single node failures. On the other hand, failure carrying packets

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

(FCP) [5] and packet recycle (PR) [6] try to forward packets to reachable destinations even in case of arbitrary number of failures. The drawbacks, however, are that FCP carries failure information in each packet all the way to the destination whereas PR forwards packets along long detours.

We propose a scalable *Localized On-demand Link State* (LOLS) routing [15] for protection against multiple failures. LOLS considers a link as *degraded*1 if its current state (say "down") is worse than its *globally advertised* state (say "up"). Under LOLS, each packet carries a *blacklist* (a minimal set of degraded links encountered along its path), and the next hop is determined by excluding the blacklisted links. A packet's blacklist is initially empty and remains empty when there is no discrepancy between the current and the advertised states of links along its path. But when a packet arrives at a node with a degraded link adjacent to its next hop, that link is added to the packet's blacklist. The packet is then forwarded to an alternate next hop. The packet's blacklist is reset to empty when the next hop makes *forward progress*, i.e., the next hop has a shorter path to the destination than any of the nodes traversed by the packet. With these simple steps, LOLS propagates the state of degraded links only when needed, and as far as necessary, and ensures loop-free delivery to all reachable destinations.

LOLS has several attractive features: 1) When there are no degraded links, forwarding under LOLS is identical to shortest path forwarding; 2) Even with degraded links, LOLS paths deviate from the optimal only by a small stretch; 3) LOLS forwarding entries can be precomputed for a given scenario of failures requiring protection; 4) Due to localized propagation of a packet's blacklist, it can be conveyed in just a few bits. With these features, LOLS compares favorably against FCP and PR. In short, unlike FCP, LOLS propagates failure information only locally. Compared to PR, forwarding paths are much shorter with LOLS. We provide a detailed contrast of LOLS with these and other related works in the next section.

System analysis

Existing system

A Localized-on-demand link state routing [1] is used for handling multiple failures In IP backbone networks. A core idea behind LOLS is to have packets carry a black list of degraded links encountered along the path that to be avoided in order to ensure loop free forwarding. The Key feature of LOLS is that a packets blacklist is reset as soon as it makes forward progress towards the destination, limiting the propagation of failure information to just few hops. We have provided that LOLS guarantees loop-free forwarding to reachable destination regardless of the number of failures in the network.

In local rerouting provide greater network availability of despite failures.

Drawbacks

- Practical version of LOLS is not implemented in this system.
- Needs addresses for handling any two link/node failures.
- Slower failure detection capability.

Proposed system

We have evaluated the overhead due to LOLS using several large real topologies. It shown that it Scales better than the recently proposed scheme FCP which has similar failure resilience objectives. We also presented a practical version of LOLS for protecting against predefined failures. It shown that needs only a modest number of header bits or not via addresses for handling any two link/node failures. Our plan is to implement a prototype of LOLS using mini net system to demonstrate its deploying ability.

Advantages

- Practical version of LOLS for protecting against predefined failures.
- Needs only a modest number of header bits for handling any two link/node failures.

System design

System architecture

In finding the solution for premature entries deficit in IP networks, first there is a necessity of establishing a connection through access point. The host name is selected after logging into this system. The selected host name is then converted into IP address[14] since we use mini net system to handle the dual link and detect single node failures. The packets that are to be send are located and the process begins from source. The blacklist files are nothing but the set of all details regarding the transmission of data packets starting from source to destination.. The data packet that is located is to be encoded for converting the packet into bit data. After encoding, Interleaving is done for the purpose of error checking at each node.

The nodes through which the data packets are to be transferred are kept ready to use. The datas in each node are in bit format and thus the data packets are received to the destination[4]. In destination, the data packets are received completely only after performing De-

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

Interleaving and Decoding.The data packet along with its details such as Efficiency, Coding rate, Block length and Interleaving depth are obtained as the result.

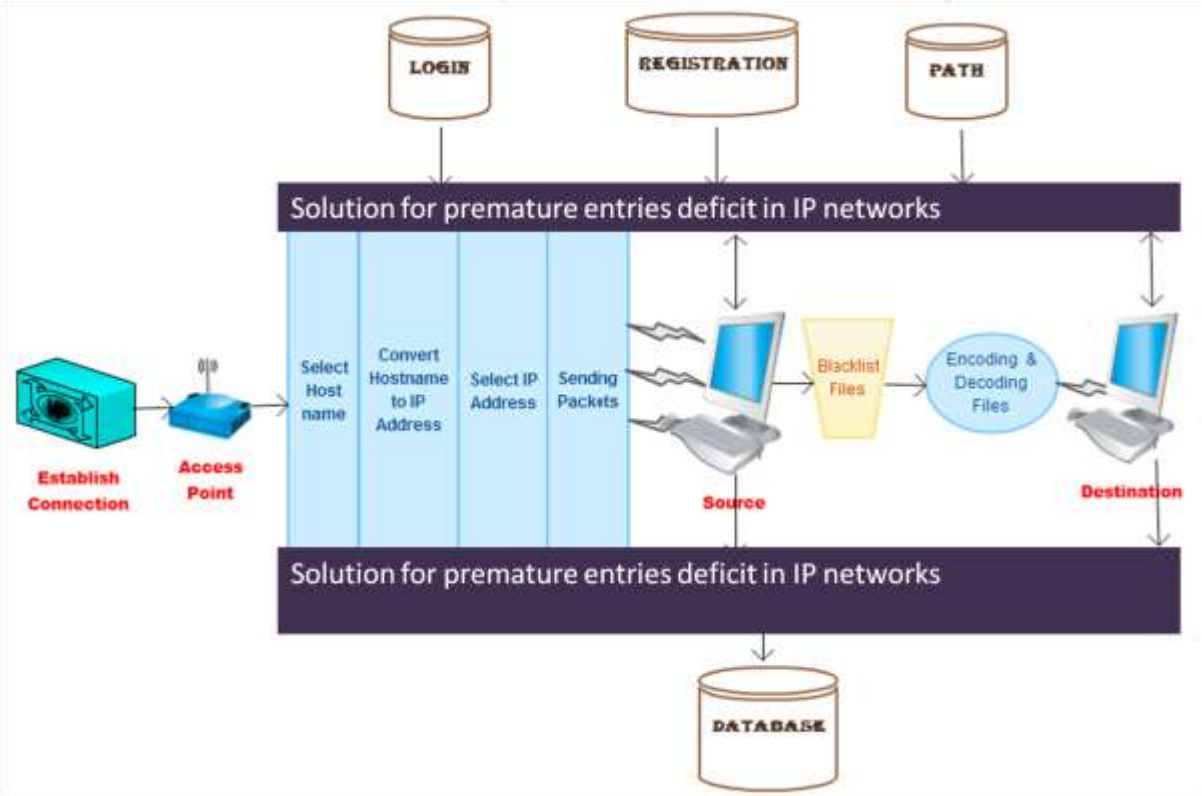

Fig.1: Architecture diagram

Module explanation

- Finding Multiple Failures
- Finding Path
- Blacklist Files
- Encoding and Decoding Blacklist Files
- Data Transfer

Finding multiple failures

In network data communication, we find the multiple failures[5] between source and destination. In this failure such as handle the dual link, detect the single node failures. And other failure is Failure Carrying Packets (FCP) and Packet Recycle tries to forward packets reachable destinations even in case of arbitrary number of failures.

Finding path

Consider a network, where is a set of nodes and is a set of directed links connecting the nodes. We find the shortest path from all nodes using Dijkstra's algorithm is executed on to improve on zero-delay paths[7] .The above polynomials solvable special case with integer delays points out a heuristic solution for the general NP-complete problem with arbitrary Nodes.

Blacklist files

We first describe the forwarding procedure and then build upon it to develop the blacklist based forwarding algorithm[15]. We also show that blacklist- based forwarding can be performed by a simple table lookup based on both destination and blacklist fields of a packet. We show the LOLS forwarding procedure in Alg. In LOLS, we first look for a next hop with the smallest path cost and forward progress without the links in the packet's blacklist

Encoding and decoding blacklist files

A straightforward way to convey a blacklist in the packet header is to represent it as a list of link identifiers. While that is a reasonable approach when dealing with arbitrary number of failures[15], pre computation of potential blacklists and forwarding tables at each node for protection against predefined failure scenarios offers an opportunity to reduce the number of bits needed to convey the blacklist information considerably.

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

Data transfer
After Encoding and Decoding Blacklist Files we sent to the Source to Destination. The encoding and Decoding is Checking Error from forwarding packets[13]. And also find the packet length and Encoding and Decoding Rate. We find sending path and packet with timing.

Data flow diagram
A data flow diagram is graphical tool used to describe and analyze movement of data through a system. These are the central tool and the basis from which the other components are developed. The transformation of data from input to output, through processed, may be described logically and independently of physical components associated with the system. These are known as the logical data flow diagrams. The physical data flow diagrams show the actual implements and movement of data between people, departments and workstations. A full description of a system actually consists of a set of data flow diagrams. Using two familiar notations Yourdon, Gane and Sarson notation develops the data flow diagrams. Each component in a DFD is labeled with a descriptive name. Process is further identified with a number that will be used for identification purpose. The development of DFD'S is done in several levels. Each process in lower level diagrams can be broken down into a more detailed DFD in the next level. The lop-level diagram is often called context diagram. It consist a single process bit, which plays vital role in studying the current system. The process in the context level diagram is exploded into other process at the first level DFD.

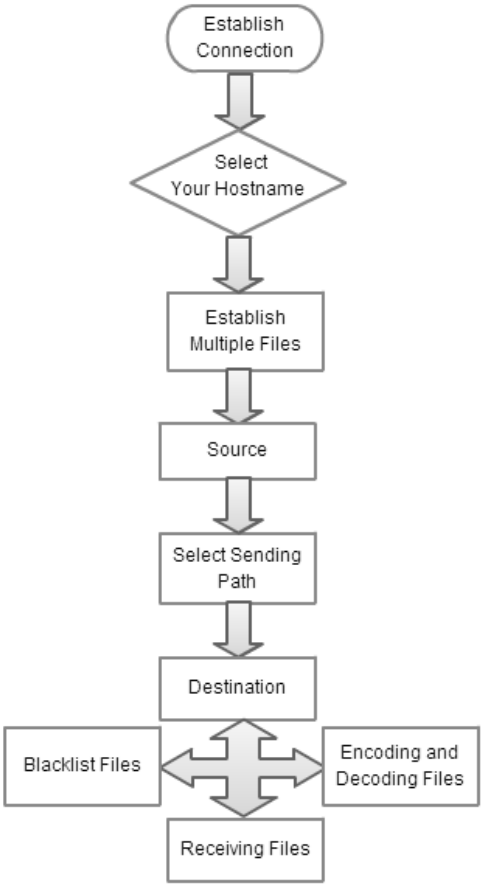

Fig 2: Data Flow Diagram

System specification

Hardware requirements

- Hard Disk : 40GB and Above
- RAM : 512MB and Above
- Processor : Intel

Software requirements

- Windows XP
- Net Beans 7.1.2
- MySql 5.0
- HeidiSQL

Conclusion and future work

Conclusion
We have proposed Handling Multiple Failures in Internet Protocol system to handle the dual link, detect the single node failures. And also detect the another failure is Failure Carrying Packets (FCP) and Failure Carrying Packet Recycle then Resend the forward packets reachable destinations even in case of arbitrary number of failures.

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

Future enhancement

In Upcoming year's we propose blacklist-aided forwarding for wireless mesh networks, provide loop-free forwarding even in the presence of multiple failed links in the network but requires that each packet carry a blacklist of failed links encountered along its path. Our aim is to achieve the best of both these approaches, i.e., successfully deliver packets while ensuring loop-freedom even in case of multiple failures without changing packet format. We propose blacklist-based interface-specific forwarding (BISF) that infers a blacklist, a list of links that might have failed, based on a packet's incoming interface and its destination, and determines the next-hop by excluding the blacklisted links. We show that BISF is loop-free regardless of the number of failures in the network while forwarding packets.

References

1. Glenn Robertson and Srihari Nelakuditi, "Handling Multiple Failures in IP Networks through Localized On-Demand Link State Routing," September 2012, IEEE *transactions on network and service management, vol. 9, no. 3*, pp. 293-305
2. A. Markopoulou, G. Iannaccone, S. Bhattacharyya, C.-N. Chuah, Y. Ganjali, and C. Diot, "Characterization of failures in an operational IP backbone network," *IEEE/ACM Trans. Netw.*, vol. 16, no. 4, pp. 749–762, Aug. 2008. Available: http://dx.doi.org/10.1109/TNET.2007.902727
3. A. Gonzalez and B. Helvik, "Analysis of failures characteristics in the uninett IP backbone network," in *Proc. 2011 IEEE Workshops of International Conference on Advanced Information Networking and Applications*, pp. 198–203.
4. O. B. et al, "Achieving sub-50 milliseconds recovery upon BGP peering link failures," in *Proc. 2005 CoNEXT*.
5. K. Lakshminarayanan, M. Caesar, M. Rangan, T. Anderson, S. Shenker, and I. Stoica, "Achieving convergence-free routing using failure-carrying packets," in *Proc. 2007 SIGCOMM*, pp. 241–252.
6. S. S. Lor, R. Landa, and M. Rio, "Packet re-cycling: eliminating packet losses due to network failures," in *Proc. 2010 HotNets*.
7. S. Kini, S. Ramasubramanian, A. Kvalbein, and A. Hansen, "Fast recovery from dual link or single node failures in IP networks using tunneling," *IEEE/ACM Trans. Networking*, vol. 18, no. 6, pp. 1988– 1999, Dec. 2010.
8. C. Alattinoglu and S. Casner, "ISIS routing on the Qwest backbone: a recipe for subsecond ISIS convergence," NANOG meeting, Feb. 2002
9. P. Francois, C. Filsfils, J. Evans, and O. Bonaventure, "Achieving subsecond IGP convergence in large IP networks," in *ACM SIGCOMM Computer Commun. Rev.*, July 2005.
10. R. Teixeira, A. Shaikh, T. Griffin, and J. Rexford, "Dynamics of hotpotato routing in IP networks," in *Proc. 2004 ACM Sigmetrics*.
11. G. I. *et al.*, "Analysis of link failures in an IP backbone," in *Proc. 2002 ACM IMW*.
12. M. Tacca, K. Wu, A. Fumagalli, and J.-P. Vasseur, "Local detection and recovery from multi-failure patterns in MPLS-TE networks," in *Proc. 2006 IEEE International Conference on Communications*, vol. 2, pp. 658–663.
13. S. Bryant, M. Shand, and S. Previdi, "IP fast reroute using notvia addresses," Internet Draft(work in progress), Mar. 2006, draftbryantshand- IPFRR-notvia-addresses-02.txt.
14. S. Rai, B. Mukherjee, and O. Deshpande, "IP resilience within an autonomous system: current approaches, challenges, and future directions," *IEEE Commun. Mag.*, pp. 142–149, Oct. 2005.
15. S. N. et al, "Blacklist-aided forwarding in static multihop wireless networks," in *2005 SECON*.
16. V. Sharma and F. Hellstrand, "Framework for MPLS-based recovery," RFC 3469, Feb. 2003.