ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 4, April 2013

Multiple Routing Configurations for Fast IP Network Recovery

Gowtham Gajala[1], Nagavarapu Sateesh[2]

Assistant Professor, Department of Information Technology, Kakatiya Institute of Technology & Science, Warangal, India[1]

Assistant Professor, Department of CSE & IT, Malla Reddy Institute of Technology, Secunderabad, India[2]

Abstract: As the Internet takes an increasingly central role in our communications infrastructure; the slow convergence of routing protocols after a network failure becomes a growing problem. To assure fast recovery from link and node failures in IP networks, we present a new recovery scheme called Multiple Routing Configurations (MRC). Our proposed scheme guarantees recovery in all single failure scenarios, using a single mechanism to handle both link and node failures, and without knowing the root cause of the failure. MRC is strictly connectionless, and assumes only destination based hop-by-hop forwarding. MRC is based on keeping additional routing information in the routers, and allows packet forwarding to continue on an alternative output link immediately after the detection of a failure. It can be implemented with only minor changes to existing solutions. In this paper we present MRC, and analyze its performance with respect to scalability, backup path lengths, and load distribution after a failure. We also show how an estimate of the traffic demands in the network can be used to improve the distribution of the recovered traffic, and thus reduce the chances of congestion when MRC is used.

Keywords: Networking, Routing, Multiple routing, IP and Network traffic, communication system routing.

I. INTRODUCTION

In recent years the Internet has been transformed from a special purpose network to a ubiquitous platform for a wide range of everyday communication services. The demands on Internet reliability and availability have increased accordingly. A disruption of a link in central parts of a network has the potential to affect hundreds of thousands of phone conversations or TCP connections, with obvious adverse effects. The ability to recover from failures has always been a central design goal in the Internet. IP networks are intrinsically robust, since IGP routing protocols like OSPF are designed to update the forwarding information based on the changed topology after a failure. This re-convergence assumes full distribution of the new link state to all routers in the network domain. When the new state information is distributed, each router individually calculates new valid routing tables.

This network-wide IP re-convergence is a time consuming process, and a link or node failure is typically followed by a period of routing instability. During this period, packets may be dropped due to invalid routes. This phenomenon has been studied in both IGP and BGP context, and has an adverse effect on real-time applications. Events leading to a re-convergence have been shown to occur frequently. Much effort has been devoted to optimizing the different steps of the convergence of IP routing, i.e., detection, dissemination of information and shortest path calculation, but the convergence time is still too large for applications with real time demands. A key problem is that since most network failures are short lived, too rapid triggering of the re-convergence process can cause route flapping and increased network instability. The IGP convergence process is slow because it is reactive and global. It reacts to a failure after it has happened, and it involves all the routers in the domain. In this paper we present a new scheme for handling link and node failures in IP networks. Multiple Routing Configurations is a proactive and local protection mechanism that allows recovery in the range of milliseconds. MRC allows packet forwarding to continue over preconfigured alternative next-hops immediately after the detection of the failure. Using MRC as a first line of defence against network failures, the normal IP convergence process can be put on hold. This process is then initiated only as a consequence of non-transient failures. Since no global re-routing is performed, fast failure detection mechanisms like fast hellos or hardware alerts can be used to trigger MRC without compromising network stability. MRC guarantees recovery from any single link or node failure, which constitutes a large majority of the failures experienced in a network. MRC makes no assumptions with respect to

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 4, April 2013

the root cause of failure, e.g., whether the packet forwarding is disrupted due to a failed link or a failed router.
The main idea of MRC is to use the network graph and the associated link weights to produce a small set of backup network configurations. The link weights in these backup configurations are manipulated so that for each link and node failure, and regardless of whether it is a link or node failure, the node that detects the failure can safely forward the incoming packets towards the destination on an alternate link. MRC assumes that the network uses shortest path routing and destination based hop-by-hop forwarding. The shifting of traffic to links bypassing the failure can lead to congestion and packet loss in parts of the network. This limits the time that the proactive recovery scheme can be used to forward traffic before the global routing protocol is informed about the failure, and hence reduces the chance that a transient failure can be handled without a full global routing re-convergence.

II. LITERATURE SURVEY

The Internet has seen tremendous growth in the past decade and has now become the critical information infrastructure for both personal and business applications. It is expected to be always available as it is essential to our daily commercial, social and cultural activities. Service disruption for even a short duration could be catastrophic in the world of e-commerce, causing economic damage as well as tarnishing the reputation of a network service provider. In addition, many emerging services such as Voice over IP and virtual private networks for finance and other real-time business applications require stringent service availability and reliability. Unfortunately, failures are fairly common in the everyday operation of a network due to various causes such as link failures etc.

A. *Existing System*

This network-wide IP re-convergence is a time consuming process, and a link or node failure is typically followed by a period of routing instability. During this period, packets may be dropped due to invalid routes. This phenomenon has been studied in both IGP and BGP context, and has an adverse effect on real-time applications. Events leading to a re-convergence have been shown to occur frequently. Much effort has been devoted to optimizing the different steps of the convergence of IP routing, i.e., detection, dissemination of information and shortest path calculation, but the convergence time is still too large for applications with real time demands. A key problem is that since most network failures are short lived, too rapid triggering of the re-convergence process can cause route flapping and increased network instability. The IGP convergence process is slow because it is *reactive* and *global*. It reacts to a failure after it has happened, and it involves all the routers in the domain.
Disadvantages include, a link or node failure is typically followed by a period of routing instability. IGP convergence process is reactive and slow and Time taking process.

B. *Proposed System*

MRC is a proactive and local protection mechanism that allows recovery in the range of milliseconds. MRC allows packet forwarding to continue over preconfigured alternative next-hops immediately after the detection of the failure. Using MRC as a first line of defence against network failures, the normal IP convergence process can be put on hold. This process is then initiated only as a consequence of non-transient failures. Since no global re-routing is performed, fast failure detection mechanisms like fast hellos or hardware alerts can be used to trigger MRC without compromising network stability. MRC guarantees recovery from any single link or node failure, which constitutes a large majority of the failures experienced in a network. MRC makes no assumptions with respect to the *root cause of failure*, e.g., whether the packet forwarding is disrupted due to a failed link or a failed router.
The main idea of MRC is to use the network graph and the associated link weights to produce a small set of backup network configurations. The link weights in these backup configurations are manipulated so that for each link and node failure, and regardless of whether it is a link or node failure, the node that detects the failure can safely forward the incoming packets towards the destination on an alternate link. MRC assumes that the network uses shortest path routing and destination based hop-by-hop forwarding.
Advantages include, guarantee of message delivery and Fast recovery from link and node failures in IP network.
Disadvantages include, at a time if node and link get failure, MRC will not support.

III. EXPERIMENTAL RESULTS

MRC is based on building a small set of backup routing configurations that are used to route recovered traffic on alternate paths after a failure. The backup configurations differ from the normal routing configuration in that link weights are set so as to avoid routing traffic in certain parts of the network. We observe that if all links attached to a node are given sufficiently high link weights, traffic will never be routed through that node. The failure of that node will then only affect traffic that is sourced at or destined for the node itself. Similarly, to exclude a link (or a group of links) from taking part in the routing, we give it infinite weight. The link can then fail without any consequences for the traffic.
Our MRC approach is threefold. First, we create a set of backup configurations, so that every network component is excluded from packet forwarding in one configuration. Second, for each configuration, a standard routing algorithm like OSPF is used to calculate configuration specific shortest

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 4, April 2013

paths and create forwarding tables in each router, based on the configurations. The use of a standard routing algorithm guarantees loop-free forwarding within one configuration. Finally, we design a forwarding process that takes advantage of the backup configurations to provide fast recovery from a component failure.

The main idea of MRC is to use the network graph and the associated link weights to produce a small set of backup network configurations. The link weights in these backup configurations are manipulated so that for each link and node failure, and regard less of whether it is a link or node failure, the node that detects the failure can safely forward the incoming packets towards the destination on an alternate link. MRC assumes that the network uses shortest path routing and destination based hop-by-hop forwarding.

It is important to stress that MRC does not affect the failure free original routing, i.e., when there is no failure, all packets are forwarded according to the original configuration, where all link weights are normal. Upon detection of a failure, only traffic reaching the failure will switch configuration. All other traffic is forwarded according to the original configuration as normal. If a failure lasts for more than a specified time interval, a normal re-convergence will be triggered. MRC does not interfere with this convergence process, or make it longer than normal. However, MRC gives continuous packet forwarding during the convergence, and hence makes it easier to use mechanisms that prevent *micro-loops* during convergence, at the cost of longer convergence times. If a failure is deemed permanent, new configurations must be generated based on the altered topology.

A. Normal Configuration

In Normal Configuration Packet is forwarded according to configuration, i.e. packet is forwarded using the forwarding table. In this process routers are selected randomly and packets are reached to the destination.

B. Router Recovery

When we send the packet from source to destination if one of router get down, then packet will not reach to destination. In order to avoid to this situation we have a mechanism that is router recovery, in our MRC router will recover back in milliseconds.

C. Backup Configuration

When a router detects that a neighbour can no longer be reached through one of its interfaces, it does not immediately inform the rest of the network about the connectivity failure. Instead, packets that would normally be forwarded over the failed interface are marked as belonging to a backup configuration, and forwarded on an alternative interface towards its destination. This process is called backup configuration.

The number and internal structure of backup configurations in a complete set for a given topology may vary depending on the construction model. If more configurations are created, fewer links and nodes need to be isolated per configuration, giving a richer (more connected) backbone in each configuration. On the other hand, if fewer configurations are constructed, the state requirement for the backup routing information storage is reduced. However, calculating the minimum number of configurations for a given topology graph is computationally demanding. One solution would be to find all valid configurations for the input consisting of the topology graph G and its associated normal link weights w_0, and then find the complete set of configurations with lowest cardinality. Finding this set would involve solving the Set Cover problem, which is known to be *NP*-complete. Instead we present a heuristic algorithm that attempts to make all nodes and links in an arbitrary bi-connected topology isolated.

Our algorithm takes as input the directed graph G and the number n of backup configurations that is intended created. If the algorithm terminates successfully, its output is a complete set of valid backup configurations. The algorithm is agnostic to the original link weights w_0, and assigns new link weights only to restricted and isolated links in the backup configurations. For a sufficiently high, the algorithm will always terminate successfully. This algorithm isolates all nodes in the network, and hence requires a bi-connected as input. Topologies where the failure of a single node disconnects the network can be processed by simply ignoring such nodes, which are then left unprotected.

Algorithm 1: Creating backup configurations.

```
1  for i ∈ {1...n} do
2      C_i ← (G, w_0)
3      S_i ← ∅
4      B_i ← C_i
5  end
6  Q_n ← N
7  Q_a ← ∅
8  i ← 1
9  while Q_n ≠ ∅ do
10     u ← first (Q_n)
11     j ← i
12     repeat
13         if connected(B_i \ ({u}, A(u))) then
14             C_tmp ← isolate(C_i, u)
15             if C_tmp ≠ null then
16                 C_i ← C_tmp
17                 S_i ← S_i ∪ {u}
18                 B_i ← B_i \ ({u}, A(u))
19         i ← (i mod n) + 1
20     until u ∈ S_i or i=j
21     if u ∉ S_i then
22         Give up and abort
23 end
```

ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 2, Issue 4, April 2013

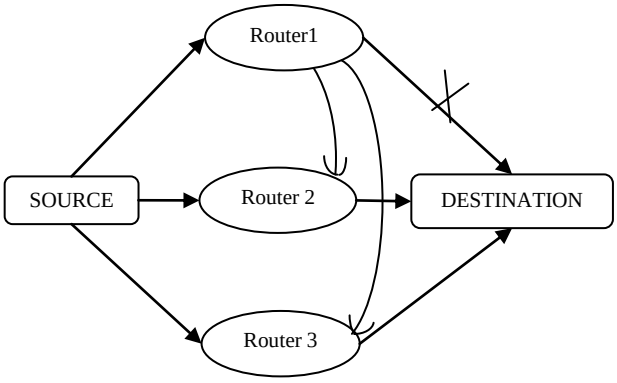

Fig 1: Backup configuration

D. Load Distribution

Whenever there is a heavy traffic (load) on the links or on routers traffic is shifted to alternate links or routers so as to avoid the congestion. This process is called load distribution. Time consumption to send a message to the destination through single router which uses a single channel is more. Load distribution reduces the time consumption. Here, load distribution delivers the data such that the data is shared among the routers.

Algorithm 2: Load Distribution.

i← i^{th} router failed
R_i← Router failed // for all i Є N
if (R_i failed) {
if (Load distribution) {
divide Msg into n-1 parts such that
$Msg_{total}=Msg_1+Msg_2+ \ldots +Msg_{n-1}$
}
for (i =0; i<n-1; i++) {
end Msg_i through R_i
}
}

The requirements that must be put on the backup configurations used in MRC, we propose an algorithm that can be used to automatically create such configurations. The algorithm will typically be run once at the initial start-up of the network, and each time a node or link is permanently added or removed.

The backup configurations so that for all links and nodes in the network, there is a configuration where that link or node is not used to forward traffic. Thus, for any single link or node failure, there will exist a configuration that will route the traffic to its destination on a path that avoids the failed element. Also, the backup configurations must be constructed so that all nodes are reachable in all configurations, i.e., there is a valid path with a finite cost between each node pair. Shared Risk Groups can also be protected, by regarding such a group as a single component that must be avoided in a particular configuration.

IV. Conclusion and Future Enhancement

Ideally, a proactive recovery scheme should not only guarantee connectivity after a failure, but also do so in a manner that does not cause an unacceptable load distribution. This requirement has been noted as being one of the principal challenges for pre calculated IP recovery schemes. With MRC, the link weights are set individually in each backup configuration. This gives great flexibility with respect to how the recovered traffic is routed. The backup configuration used after a failure is selected based on the failure instance, and thus we can choose link weights in the backup configurations that are well suited for only a subset of failure instances. MRC is based on providing the routers with additional routing configurations, allowing them to forward packets along routes that avoid a failed component. MRC guarantees recovery from any single node or link failure in an arbitrary bi-connected network. By calculating backup configurations in advance, and operating based on locally available information only, MRC can act promptly after failure discovery. To assure fast recovery from link and node failures in IP networks, we present a new recovery scheme called Multiple Routing Configurations (MRC). Our proposed scheme guarantees recovery in all single failure scenarios, using a single mechanism to handle both link and node failures, and without knowing the root cause of the failure. In future enhancement we can handle at a time multiple links and node failures.

References

[1] D. D. Clark, "The design philosophy of theDARPAinternet protocols," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 18, no. 4, pp. 106–114, Aug. 1988.

[2] A. Basu and J. G. Riecke, "Stability issues in OSPF routing," in *Proc. ACM SIGCOMM*, San Diego, CA, Aug. 2001, pp. 225–236.

[3] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, "Delayed internet routing convergence," *IEEE/ACM Trans. Networking*, vol. 9, no. 3, pp. 293–306, Jun. 2001.

[4] C. Boutremans, G. Iannaccone, and C. Diot, "Impact of link failures on VoIP performance," in *Proc. Int. Workshop on Network and Operating System Support for Digital Audio and Video*, 2002, pp. 63–71.

[5] D.Watson, F. Jahanian, and C. Labovitz, "Experiences with monitoring OSPF on a regional service provider network," in *Proc. 23rd Int. Conf. Distributed Computing Systems (ICDCS'03)*, Washington, DC, 2003, pp. 204–213, IEEE Computer Society.

[6] P. Francois, C. Filsfils, J. Evans, and O. Bonaventure, "Achieving sub-second IGP convergence in large IP networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 2, pp. 35–44, Jul. 2005.

[7] A. Markopoulou, G. Iannaccone, S. Bhattacharyya, C.-N. Chuah, andC. Diot, "Characterization of failures in an IP backbone network," in *Proc. IEEE INFOCOM*, Mar. 2004, vol. 4, pp. 2307–2317.

[8] S. Nelakuditi, S. Lee, Y. Yu, Z.-L. Zhang, and C.-N. Chuah, "Fast local rerouting for handling transient link failures," *IEEE/ACM Trans. Networking*, vol. 15, no. 2, pp. 359–372, Apr. 2007.

[9] S. Iyer, S. Bhattacharyya, N. Taft, and C. Diot, "An approach to alleviate link overload as observed on an IP backbone," in *Proc. IEEE INFOCOM*, Mar. 2003, pp. 406–416.

[10] S. Rai, B. Mukherjee, and O. Deshpande, "IP resilience within an autonomous system: Current approaches, challenges, and future directions," *IEEE Commun. Mag.*, vol. 43, no. 10, pp. 142–149, Oct. 2005.