Advanced Engineering Forum *Online: 2011-09-09*
ISSN: 2234-991X, Vol. 1, pp 320-324
doi:10.4028/www.scientific.net/AEF.1.320
© 2011 The Author(s). Published by Trans Tech Publications Ltd, Switzerland.

Virtual Area Routing: A scalable intra-domain routing scheme

Dan Zhao[1, a], Chunqing Wu[1, b], Xiaofeng Hu[1, c] and Hongjun Liu[1, d]

[1]School of Computer, National University of Defense Technology, Changsha, Hunan, China

[a]danzhao.nudt@gmail.com, [b]xixiwu2001@yahoo.com.cn, [c]xfhu@nudt.edu.cn,

[d]seeker_lhj@163.com

Keywords: intra-domain routing, link state, scalability, OSPF area, control and data separation

Abstract. As a distributed link state protocol, OSPF shows poor scalability when dealing with intra-domain routing in large networks. In this paper we present a new routing scheme called Virtual Area Routing (VAR) aimed to overcome the scalability problems of OSPF. The control plane is separated from data plane and is undertaken by dedicated elements in VAR. We show that VAR can simplify network configuration and management as well as avoid unnecessary route computation. Finally we conduct experiments in real topology and the result shows VAR can reduce control overhead and improve network performance.

1. Introduction

Autonomous System (AS) employs distributed link state protocols to take on intra-domain routing nowadays. OSPF [1] is the most prevalent link state protocol which is widely used in intra-domain routing. Each router running OSPF perceives the link state change and floods the information in the form of Link State Advertisement (LSA) to others. After receiving all LSAs about each single link, the router calculates routing table using Dijkstra's algorithm and updates forwarding table.

As a network grows, OSPF gradually exposes scalability problems. Once the network changes, LSA flooding will result in a large amount of protocol traffic. Ref. [2] describes a severe phenomenon called *LSA Storm* in OSPF. A network experiencing LSA Storm which causes high CPU and memory utilization at the router may drive the network to an unstable state. On the other hand, flooding will enforce each router to compute Shortest-Path Tree (SPT) to acquire feasible routes. Intuitively, for links not in SPT, their changes can't cause any route update, where the computation is unnecessary. The overhead of route computation based on network-wide view is such high that is likely to result in instabilities when the network is experiencing LSA Storm. The convergence process is also slowed down due to the large scale flooding of LSAs which in turn affects traffic delivery.

To restrict the flooding scale, OSPF area emerges. In an OSPF network with areas, flooding never goes beyond the area boundary. The Area Border Router (ABR) summarizes the distance information and announces them into neighboring area. OSPF area seems to solve the scalability problem, but it introduces other disadvantages [3]. A notable one is the configuration and management complexity when dealing with large network with areas; the other is suboptimal inter-area routing. In fact, if ABR is not deployed and configured properly, the message overhead can be worse than flooding.

One may ask the question: "Why these problems come about?" It seems that all protocols based on link state are born to be like this. We still try to answer the question, and conclude the fundamental reasons below:

1. Distributed protocol implementation. Routers can only be aware of local topology change, and no router knows the exactly network-wide view that is necessary for constructing stable routing involving network dynamics. In order to communicate the link state changes, flooding is essentially needed, so scaling problems come forth.

2. The tight coupling of control plane and data plane. Large network may occasionally experience LSA Storm which can probably result in congestion when the underlying network is overloaded by data traffic, or even worsen the congestion situation. Other abnormal interactions between control and data plane are beyond this paper's scope, but yet undeniable [7].

To overcome the inherent limitations of OSPF, we present an approach called ***Virtual Area Routing (VAR)***. In VAR, control plane is separated from data plane by aggregating control logic of single OSPF instance into a dedicated element. A *Virtual Area* is a region under control by a specific

This article is an open access article under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0)

control element. All control elements together consist of the control network that is totally detached from traditional network. We refer to the separated networks as control and data network without confusion, and control and data node in each network respectively. Under VAR circumstance, flooding is replaced by end-to-end communication between specific control and data node pair. We show that VAR not only simplifies network configuration and management, but also avoids unnecessary computation without losing routing optimality and network stability.

2. Background and Related Work

Flooding is a widely-used scheme to spread local information over the whole domain and it plays an important part in link state protocol. The primary drawback is that flooding doesn't scale well in large network. There are several OSPF specifications proposed concerning this problem [2,4]. All these proposals definitely import control and implementation complexities, and applying them needs network-wide update and reconfiguration. OSPF Area is seemed to be a practical method to overcome the disadvantage, yet the isolation of knowledge may introduce other problems including configuration complexity, sub-optimal routing, etc. [3]

Another approach to solve the scalability problem is to avoid the burst of control traffic. FCP [5] aims to eliminate convergence period by carrying failure information along the regular packets. Routers can compute an available path with failure information in time and no flooding exists. One can easily notice the possibility of excessive computation when processing FCP and extra overhead, which may increase the possibility of instability. XL [6] is an approximation link state protocol in which the update propagation must satisfy specified properties while others are suppressed. There may be sub-optimal paths produced by the designed approximation factor.

According to our point of view, control and data separation becomes a reasonable way to solve the scalability problem. We are not the first to claim that control and data separation can enhance routing performance. RCP [7] is proposed to improve the flexibility of routing selection as well as facilitate configuration. 4D [8] is an innovative routing architecture which is consisted of Decision plane, Dissemination plane, Discovery plane, Data plane. Others [9,10,11] basically follow the same idea that control functionality is separated from data processing and should be logically centralized where the routing complexity can be better managed. Ref. [12] also proposed a separation of iBGP control plane for inter-domain routing. Unlike the previous proposals about absolute network control, our approach takes control over virtual area to solve the scalability issues referred to link state flooding; meanwhile the complexity and demanding capability of control can be properly managed.

3. Virtual Area Routing

3.1 overview

As depicted in Figure 1, there are two components of intra-domain routing system: a data network for packet forwarding and a control network for routing control. The data and control traffic are clearly classified to be transferred in separate networks, through which the unwanted interference can be eliminated.

The network has only a single area with multiple virtual areas. A virtual area is a logic region controlled by specific control node. All the control nodes together form the control network. Data node discovers neighbors in data network and originates LSA to describe the link state it connects to. All the LSAs are then reported to control nodes where the consistent network-wide view about data network can be constructed through the communication among multiple control nodes. The control node performs SPT calculation rooted at each data node in controlled virtual area. By comparing the SPT with previous one the control node can identify whether a data node's routes will change. For those data nodes whose routes will actually change, disseminate the corresponding LSAs to them, while other data node's update is suppressed.

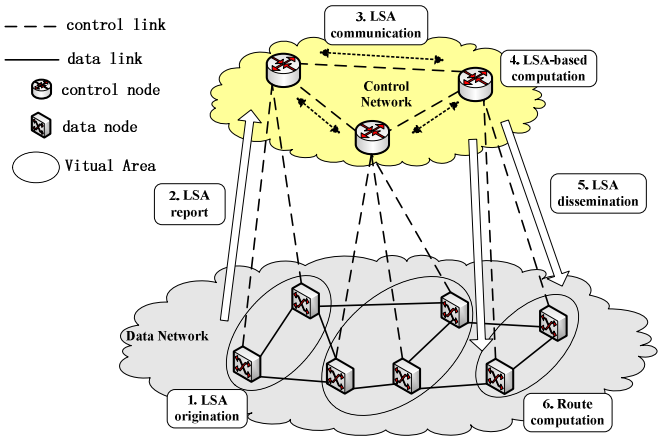

Fig 1: VAR overview

3.2 Virtual Area

Virtual area is quite different from traditional OSPF area. In fact, virtual area is a logical cluster of routers concerning about control. Routers in a virtual area may be geographically distributed, whereas OSPF area requires routers to be locally centralized. Virtual areas are not only neighboring but can also be intersecting upon some routers. Routers are configured in the single area way because no ABR exists, and the only extra configuration needed is the control relationship between virtual area and corresponding control node. Network operator can easily manage virtual area with certain purpose by attaching data node to a specified control node on which the objectives can be integrated.

3.3 Signaling in Control Network

When the data network changes, the originated LSA will transit along a path for signaling. The signaling path must be constructed without any routing protocol involved. We introduce a simple protocol to discover signaling path in control network (shown in Figure 2). This protocol uses designed packet called probe to carry the router-id list indicating the transmitted signaling path. Any node that has received probe packet can acquire the signaling path.

```
Input: probe_pkt received from control network
if ( router-id ∈ probe_pkt.node_list)
  Drop (probe_pkt)
else
do
  Signaling_path = Record (probe_pkt.node_list)
  probe_pkt.node_list ∪= router-id
  Send (probe_pkt)
done
```

Fig 2: Signaling protocol

A control node periodically sends probe packet through all interfaces into control network. On receiving the probe packet, one node checks whether the probe has been processed earlier. If not, the node records the router-id list as signaling path to specific control node with corresponding interface information, and then continues to send this packet with its router-id added through the control interface. Note that probe packet can only be transmitted inside control network. Finally, the signaling path will be available for all nodes inside the control network. Among all signaling paths, the shortest path will be used for LSA transmission for configured virtual area.

3.4 Selective LSA Dissemination

In order to eliminate unnecessary route computation, the update LSAs are selectively disseminated to specific data nodes in VAR. According to the complete view of network state, the control node can compute shortest-path trees and acquire all-pairs shortest paths that are actually used for packet forwarding in the data node's point of view inside virtual area. For those data nodes whose shortest-path tree has changed, LSA will be definitely disseminated to them, while other nodes' updates are suppressed.

It is obvious that the update suppression may result in inconsistent topology view observed by different routers after multiple network changes. There stands the fact that distinct LSAs have dependencies on forwarding path, that is, the suppressed link may be used in later forwarding path after another update though this link is unavailable. It is computationally infeasible to identify such dependencies during each update process because of the combination explosion considering a series of LSAs. We solve this problem by maintaining historical LSAs information for each data node since its last update. Whenever the update is invoked, the control node sends all the suppressed LSAs to enforce the consistency of network view. After receiving the LSAs, the data node can compute correct routes based on the most recent network topology without missing details. The dissemination overhead can be significantly decreased by encapsulating several LSAs into single packet as is also adopted in standard OSPF.

3.5 Routing Properties

The fundamental goal of VAR is to acquire feasible paths for packet forwarding without OSPF area and flooding. Given the data network topology view $N_d(V_d, E_d)$ and control network view $N_c(V_c, E_c)$ with node set V, link set E respectively, we present several routing properties as follows.

Definition 1. *Routing Completeness: A routing is complete if for any node $v \in V_d$, all routes of v can be acquired.*

Definition 2. *Routing Optimality: A routing is optimal if for any node $v \in V_d$, each route of v is the shortest path.*

Definition 3. *N_d is consistent if it represents the entire network topology information.*

Now we state our theorem that VAR satisfies routing completeness and optimality with proof.

Theorem. *VAR is Complete and Optimal.*

Proof. Firstly, control node $c \in V_c$ can receive all network changes of N_d since LSAs are reported to c, that is, N_d on c reflects the entire data network state at that time. So N_d owned by c is consistent.

Secondly, the link state protocol basically employs Dijkstra's algorithm to compute SPT for constructing routes. SPT based on N_d consists of *all shortest paths* from root to other nodes, which means routing according to SPT based on consistent N_d is complete and optimal.

Thirdly, c will compute the SPT rooted at $d \in V_d$ in virtual area using N_d. By comparing shortest paths in SPT, c can know the differences between SPT and the previous one SPT'. Then c will send to d all historical LSAs related to d since its last update. d can construct consistent N_d that reflect the most recent network state, so the SPT based on N_d is identical with the one in c. If the update is suppressed, that means both SPTs owned by c and d have no change since last update. Then it is obviously true that the SPT of d is identical with the SPT rooted at d owned by c inside virtual area.

By combining all statements above, we can conclude that VAR is complete and optimal.

4. Evaluations

We have implemented a prototype of VAR on Quagga [13] by extending OSPF. We build a real network topology with 15 nodes and 40 links in lab to evaluate VAR. The link delays are 10ms on the average, and link costs are randomly chose between 1 and 5. Other parameters are set to default value in Quagga.

We compare the message overhead undertaken by control plane. Figure 3 shows the number of LSAs after random links in both OSPF and VAR. Through our experiments, the number of LSAs can be reduced by about 70% compared with that of OSPF. This means VAR can significantly reduce the control message overhead to cut down the network process load.

In Figure 4 we investigate the transmission delay that is a key component of convergence. Generally, the transmission delay depends on the location of failures. In VAR the transmission of LSAs is in the charge of control network that is much smaller than traditional network, so the transmission delay decreased definitely. In our experiments, we observe that the VAR result in a 10-20ms reduction of transmission delay. We believe the speedup of convergence would be far better in large networks.

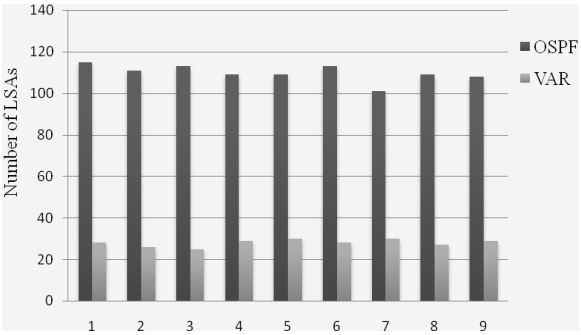

Fig 3: Number of LSAs after random link failures

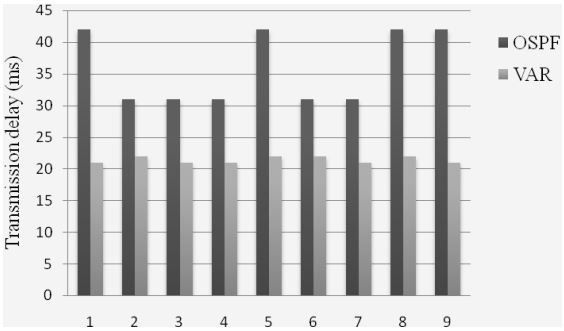

Fig 4: LSA transmission delay after random link failures

5. Conclusions

We have presented VAR, a novel intra-domain routing scheme. VAR separates traditional control and data plane into detached networks, and allows OSPF network to be configured in only one area way. In this paper we have proved that VAR is complete and optimal, and can reduce configuration and management complexity. We finally use real network topology to evaluate VAR. The result shows VAR can reduce LSA message overhead and the transmission delay. Overall, VAR can properly solve the OSPF scalability problems in intra-domain routing.

This research is supported by National Natural Science Foundation of China under Grant No. 61070199

References

[1] J. Moy, OSPF Version 2, RFC 2328, April 1998
[2] G. Choudhury, Ed., Prioritized Treatment of Specific OSPF Version 2 Packets and Congestion Avoidance, RFC 4222, October 2005
[3] M. Thorup, OSPF Areas Considered Harmful, Private paper, Apr 2003.
[4] P. Pillay-Esnault, OSPF Refresh and Flooding Reduction in Stable Topologies, RFC 4136, July 2005
[5] Karthik Lakshminarayanan, Matthew Caesar, Murali Rangan, Tom Anderson, Scott Shenker, Ion Stoica, Achieving Convergence-Free Routing using Failure-Carrying Packets, In: Proc. of ACM SIGCOMM'07
[6] Kirill Levchenko, Geoffrey M. Voelker, Ramamohan Paturi, and Stefan Savage, XL: An Efficient Network Routing Algorithm, In: Proc. of ACM SIGCOMM'08
[7] Nick Feamster, Jennifer Rexford, The Case for Separating Routing from Routers, In: Proc. of SIGCOMM'04 Workshop
[8] Albert Greenberg, Gisli Hjalmtysson, David A. Maltz, Andy Myers, Jennifer Rexford, Geoffrey Xie, Hong Yan, Jibin Zhan, Hui Zhang, A Clean Slate 4D Approach to Network Control and Management, ACM SIGCOMM Computer Communication Review 35 (5) (2005)
[9] Matthew Caesar, Donald Caldwell, Nick Feamster, Jennifer Rexford, Aman Shaikh, Jacobus van der Merwe, Design and Implementation of a Routing Control Platform, In: Proc. of 2th USENIX Symposium on Networked Systems Design & Implementation(NSDI), 2005
[10] Hong Yan, David A. Maltz, T. S. Eugene Ng, Hemant Gogineni, Hui Zhang, Zheng Cai, Tesseract: A 4D Network Control Plane, In: Proc. of 4th USENIX Symposium on Networked Systems Design & Implementation(NSDI), 2007
[11] Yi Wang, Ioannis Avramopoulos, Jennifer Rexford, Morpheus: Enabling Flexible Interdomain Routing Policies, In: Proc. of 6th USENIX Symposium on Networked Systems Design & Implementation (NSDI), 2009.
[12] Iuniana Oprescu, Mickael Meulle, Steve Uhlig, Cristel Pelsser, Olaf Maennel, Philippe Owezarski, Rethinking iBGP Routing, In: Proc. of ACM SIGCOMM'10
[13] Quagga Routing Suite, http://www.quagga.net