LOAD BALANCING IN SOFTWARE DEFINED NETWORK

Anusha Vegesna, Ashwini Swain , Karan Savla, Kushal Ajmera
Information Technology, Dwarkadas J Sanghvi College of Engineering, India

Abstract

Software-defined networking (SDN) is an emerging approach in computer networks, which solves problems related to traditional networks like scalability, latency, etc. Load balancing in SDN distributes or balances traffic in the network to minimize network latency and improve resource utilization. However commonly used algorithms like Round Robin, which is statically configured, unevenly balances the traffic and are thus not very efficient. Dynamic Load Balancing algorithms like Shortest Load First, Djikstra's etc routes load dynamically in the network. By using Djikstra's algorithm, we conclude that it gives better results as compared to round robin algorithm.

Keywords: Networks, Load Balancing, Djikstra, Floodlight Controller, Mininet.

1. Introduction

A computer network is a set of connected computers digital communication network which allows nodes in the network to communicate and share resources. In traditional networks, the connections between the nodes are established using either physical cable media or by wireless media.

Software-defined networking (SDN) [1][2][3] is an upcoming approach in computer networks which allows network administrators to programmatically configure, control and manage network and its behavior dynamically. SDN is used to overcome the shortcomings of traditional networks, which don't support dynamic and scalable computing thereby making the network agile and flexible.

The idea behind SDN is the separation of the control plane of the network, which makes decisions about how to route or send packets in the network which is the data plane, where the actual routing of packets takes place. In traditional network when a packet arrived at a switch, rules hard coded in the switch's hardware directs the switch where to forward the packet. The switch then sends all the packets for the same destination through the same path. However in SDN, rules for packet routing and management are sent to the switch from a central controller, which is an application running on a server. All the communicating nodes and switches are connected to the controller. The switches query the controller for instructions when needed, and provide it with information about traffic they are handling. Figure 1 shows the difference between a traditional network and Software Defined Network.

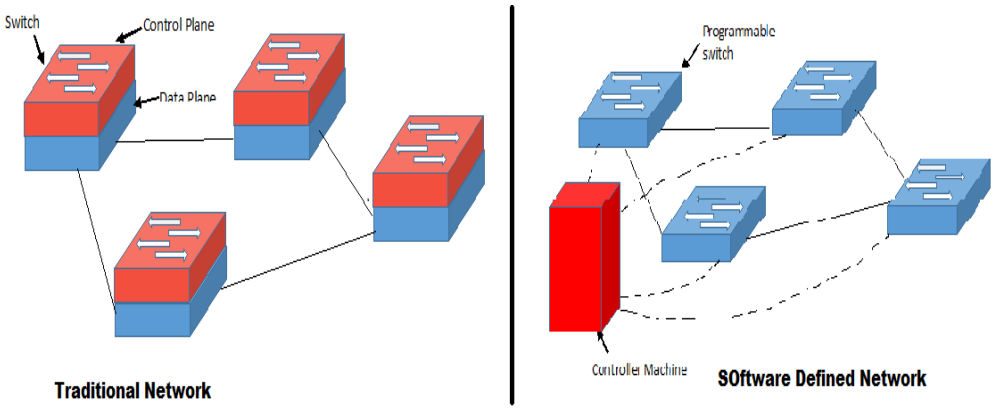

Figure 1. Difference between Traditional and Software Defined network.

2. Load Balancing in Software Defined Network

Load balancing is a technique of distributing traffic efficiently among the servers so that the servers are not overwhelmed by sudden changes in the traffic patterns or load. It can save power, reduce data loss and decrease network latency thereby improving resource utilization [5].

The Load Balancing algorithm does the main task of choosing the next server or the best path for routing the data packets from a congested route to an under-loaded route. It is a challenging task in SDN as the amount of data flow is very high.

3. Existing System

The Round Robin balancing is one of the most used algorithm for load balancing. Floodlight Controller also has round robin implemented into it by default. The working of the algorithm is very simple: forward all the incoming packets to all the available switches over the network. This however ignores the fact that not all the switches have the same load handling capacity, thus it can overload lower capacity switches [6].

Limitations:

- Statically configured: They are not fault tolerant, i.e. if there is any change in the network, the traffic route will not be changed, leading to congestion.
- Uneven load balancing[1]

4. Proposed Solution

Our proposed solution uses Djikstra's algorithm for dynamic load balancing in SDN using the floodlight Controller. Djikstra's algorithm calculates the shortest path between the communicating nodes, and routes the packets along the same path. In case of congestion in the current data path, the algorithm dynamically examines all the remaining paths between the two nodes, calculates the cost of each path and pushes the path with the least cost on to the controller. The controller then routes the packets along the new path.

4.1. Benefits of Proposed Solution

- Visibility provided to the control plane: Rather than each individual data plane device relying on advertisements from other devices to build its view of the network topology, a single control plane device has a view of the entire network. This provides a platform from which advanced routing, security, and quality decisions can be made, hence the need for programmability.
- With a centralized controller device it is much easier to gain usable data about real time flows on the network, and make decisions (automated or manual) based on that data.
- More intelligent processes can be added to handle complex systems. Real-time decisions can be made for traffic optimization, security, outage, or maintenance. Separate traffic types can be run side-by-side while receiving different paths and forwarding that can respond dynamically to network changes.

5. Implementation

Floodlight and OpenFlow [4] are the two most common, open source SDN controllers. We have used the floodlight controller. Floodlight uses round robin algorithm by default for load balancing, which needs to be removed from the flood light default.properties [8]. For setting up the virtual network, mininet, which is network emulator consisting of virtual hosts and other networking components is used [7]. After the network is connected to the floodlight controller, it uses REST API to get information of the complete network like the source and destination address, port information of the links, etc.

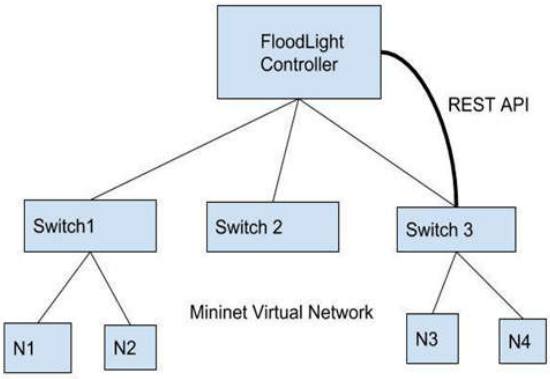

Figure 2. Mininet Virtual network.

After the algorithm calculates the shortest path, the REST API updates the flow rules and the data packets are pushed onto the new path. Information regarding the network such as In-Port, Out-Port, Source IP, Destination IP, Source MAC address, Destination MAC address are pushed into the flows. The algorithm continues to update this information every minute thus making it dynamic.

6. Results

We measure the amount of bytes transferred and the bandwidth using iPerf, which is an open-source tool for measuring network performance.

Transfer (Gbytes) – Round Robin	Transfer (Gbytes) – Djikstra	B/W(Gbits) – Round Robin	B/W(Gbits) – Djikstra
16.3	28.4	14.5	23.4
23.1	32.3	20.1	27.6
24.3	30.1	21.1	28.4
21.6	35.6	18.8	16.3
Avg=21.325	**Avg=31.6**	**Avg=18.625**	**Avg=23.92**

Table 1. Network performance measure of various algorithm.

The above table shows the increase in average data transfer after using the Djikstra's algorithm over the round robin algorithm.

7. Future Scope

The Djikstra's algorithm gives better results over the statically configured algorithm like round robin. This performance can be enhanced further by employing additional algorithm like Shortest Load First, as our system deals with packets of equal load. Shortest Load First with Djikstra's will route higher load packets through the shortest path and may improve the efficiency of the system

8. Conclusion

This paper documents attempts made to develop a dynamic and efficient load balancing algorithm in SDN. We have succeeded in implementing such an algorithm. This idea can be carried forward to make more complex systems with even more efficient algorithm. A Software Defined Network can efficiently manage systems with ever increasing data like Cloud Technologies, Data Centers, etc.

9. References

9.1. Journal Article

[1] Smriti Bhandarkar, Kotla Amjath Khan, “Load Balancing in Software defined Network (SDN) Based on Traffic Volume ”, Advances in Computer Science and Information Technology (ACSIT),Volume 2, Number 7; April –June, 2015.

[2] Yosr Jarraya, Taous Madi, and Mourad Debbabi, "A Survey and a Layered Taxonomy of Software-Defined Networking”, Communications Surveys and Tutorials, IEEE Issue: 99, 2014.

[3] Bruno Astuto A. Nunes,Marc Mendonca,Xuan-Nam Nguyen,Katia Obraczka,Thierry Turletti,”A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks”, IEEE Communications Surveys & Tutorials (Volume: 16, Issue: 3, Third Quarter 2014).

[4] Lara Adrian, Anisha Kolasani, Byrav Ramamurthy, "Network innovation using OpenFlow: A survey", pp. 1-20, 2013.

[5] Mao Qilin, Shen Weikang, ”A Load Balancing Method Based on SDN”, Measuring Technology and Mechatronics Automation (ICMTMA), IEEE 13-14, June 2015.

9.2. Conference Proceedings

[6] Sukhveer Kaur, Krishan Kumar, Japinder Singh, Navtej Singh Ghumman,”Round-robin based load balancing in Software Defined Networking”, Computing for Sustainable Global Development (INDIACom), IEEE 11-13 March 2015.

9.3. Websites:

[7] Mininet Walkthrough :http://mininet.org/walkthrough/

[8] Discussions-Google group: https://groups.google.com/a/openflowhub.org/forum/#!f orum/floodlight-dev