Design Project

Shalva Kushashvili
The City College Of New York
M.S. Cybersecurity
Computer Communication Networks EE F6000 1GH

ABSTRACT

In this project, I am going to build and do simulations on a seven-node network using a network modeling simulation tool called **Riverbed Modeler,** which provides high-fidelity modeling, scalable simulation, and detailed analysis of a broad range of wired and wireless networks. To observe the network's performance, I used two routing algorithms called RIP and OSPF. The performance is based on the factors such as – link delay, average link flow, number of hops, and so on. It's also worth mentioning that these factors may vary in different situations.

1. INTRODUCTION

In a computer network, interconnected devices are able to exchange data and share resources with each other. These networked devices use communication protocols, which are a set of rules which transmits information over physical or wireless technologies. The history of modern computer networking technology goes back to 1969 when ARPANET became the first connected computer network. It implemented a protocol called TCP/IP, which later became the internet. Computer networks were initially used for military and defense purposes, but with the advent of internet technologies, a computer network has become indispensable for enterprises. Modern-day networks deliver more than connectivity. They've become critical for the digital transformation and success of businesses. Latest computer networks can operate virtually, integrate on a large scale, respond quickly to changing conditions, and of course provide data security.

Nodes and links are the main building blocks in computer networks. A network node may be data communication equipment such as a modem, hub, switch, or data terminal equipment such as two or more computers and printers. A link is the transmission media connecting two nodes. Links may be physical, such as cable wires or optical fibers, or they can be wireless. The combination and arrangement of links and nodes is called network topology. They can be configured in many different ways to get different results. Some types of network topologies are – Bus topology, Ring topology, Star topology, and so on. In this project, we have a custom topology where we have 7 nodes. With OPNET (also known as Riverbed Modeler), we can simulate routing algorithms on this given topology and see how each router decides where to send a packet based on possible routing costs from one node to another.

2. DESCRIPTION OF THE SIMULATED MODEL

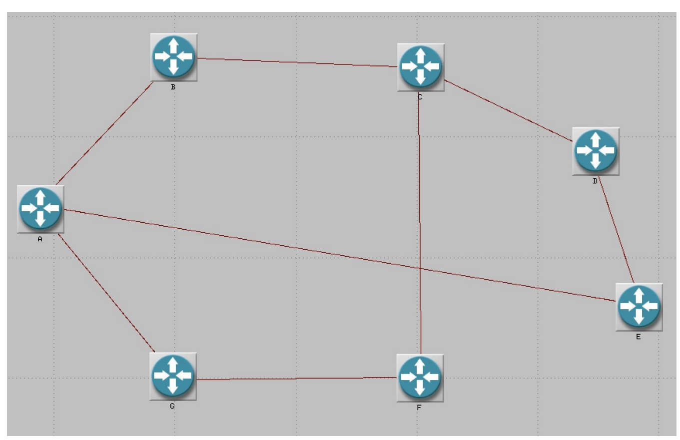

Figure 1: Seven-node network

To simulate the given project, I used router_slip_dc_100_upgrade_adv as the nodes. As for the links, I used 100BaseT duplex links, which operate at 100Mbps amongst each of the nodes. As for the traffic flow between the sender and receiver, I configured it based on the routing matrix which was given to us for the project. It's also worth mentioning that IPs are assigned dynamically.

	A	B	C	D	E	F	G
A	-	900	600	500	700	800	200
B	900	-	300	200	900	500	300
C	600	300	-	700	100	400	500
D	500	200	700	-	400	700	900
E	700	900	100	400	-	500	400
F	800	500	400	700	500	-	200
G	200	300	500	900	400	200	-

Table 1: The traffic matrix between each node (in Megabits per second)

3. SIMULATION RESULTS

For this experiment, I used two routing algorithms called RIP and OSPF. RIP uses a **distance vector algorithm** to decide which path to put a packet on to get to its destination. Each RIP router maintains a routing table, which is a list of all the destinations the router knows how to reach. As for OSPF, It uses a **link state routing (LSR) algorithm** and falls into the group of interior gateway protocols, operating within a single autonomous system. After setting up traffic flow between each node, Let's find out the average link utilization. As we can see from the results, the average point-to-point utilization in RIP and OSPF looks almost similar to one another, but if we take a closer look, we can see that the traffic between some nodes is different in RIP and OSPF (For example, traffic between G and F).

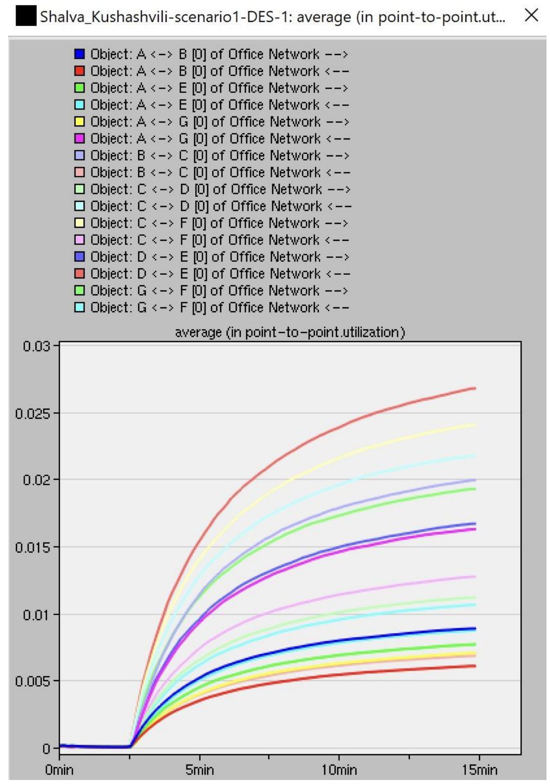

Figure 3: Average point-to-point utilization with RIP

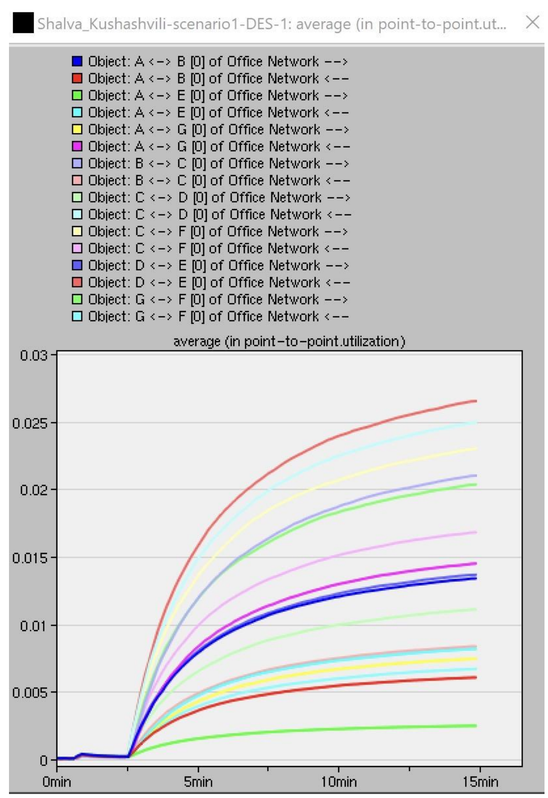

Figure 4: Average point-to-point utilization with OSPF

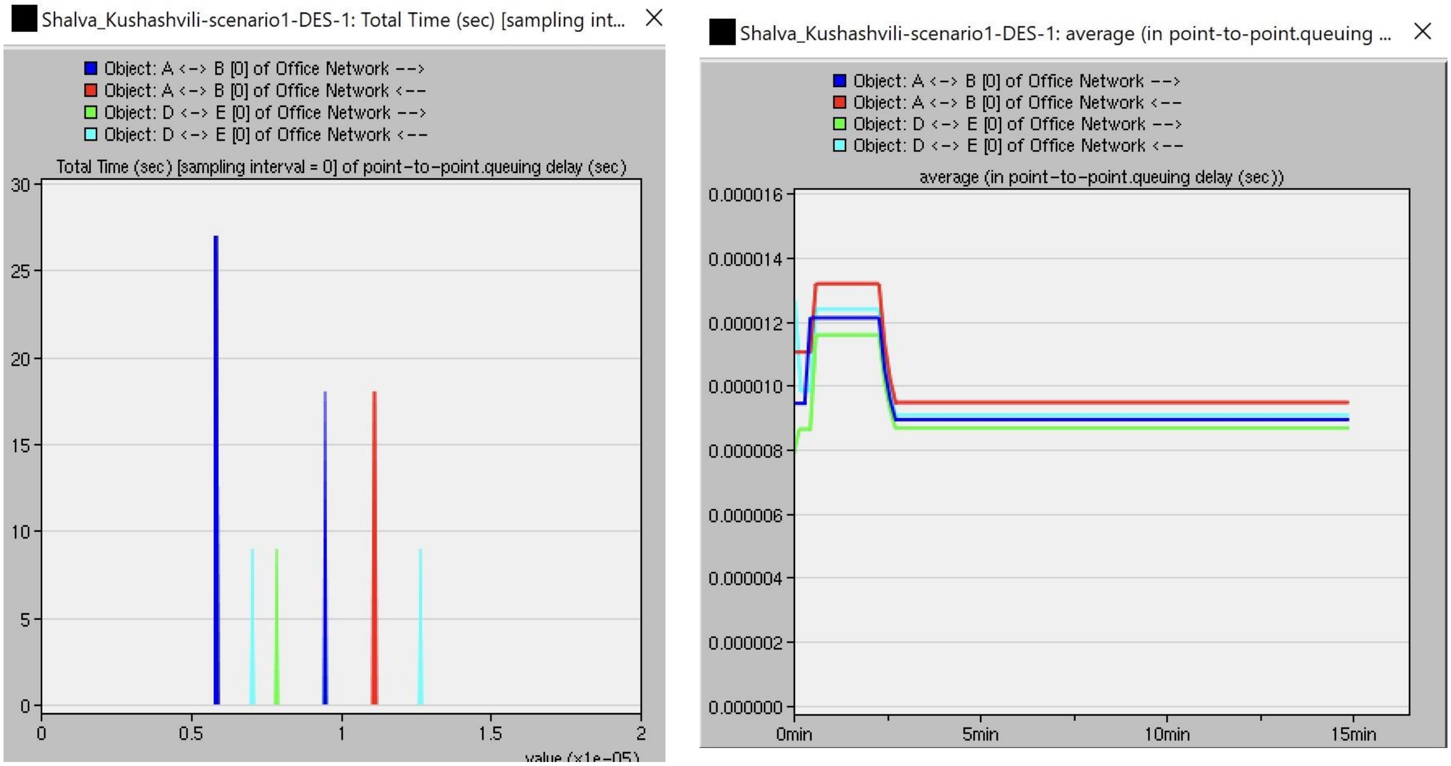

Figure 4: Link delay histogram and average delay for A ←→ B and D ←→ E (RIP)

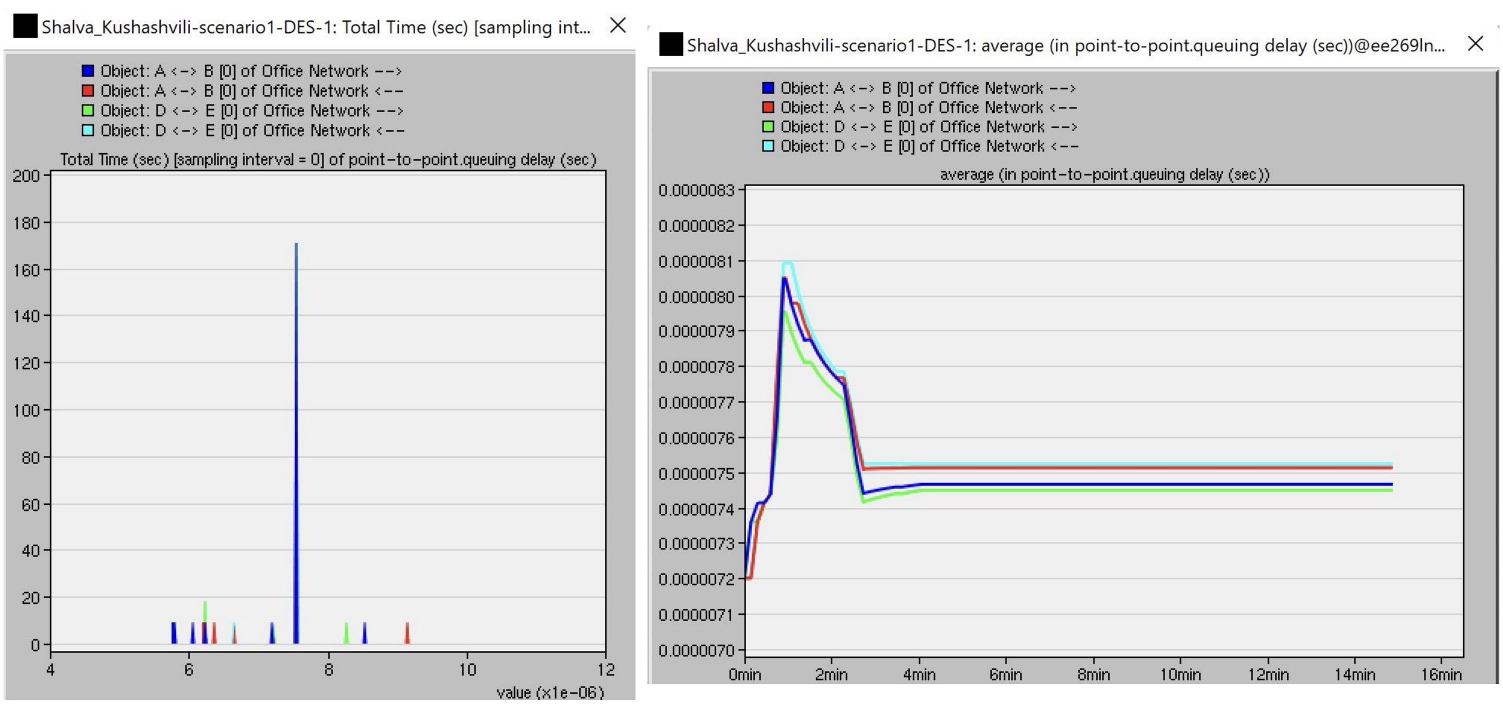

Figure 5: Link delay histogram and average delay for A ←→ B and D ←→ E (OSPF)

From figures 4 and 5, we can see that RIP takes a while to normalize its point-to-point delay with different routers. OSPF normalizes faster because of the speed of convergence since OSPF is known to handle network problems expeditiously. On the contrary, RIP is known to have issues with convergence as it has trouble instituting a workaround route when a given route is blocked. Now let's take a look at the histogram and average link flows for the two other links. Let's choose traffic between B and C, and traffic between G and F.

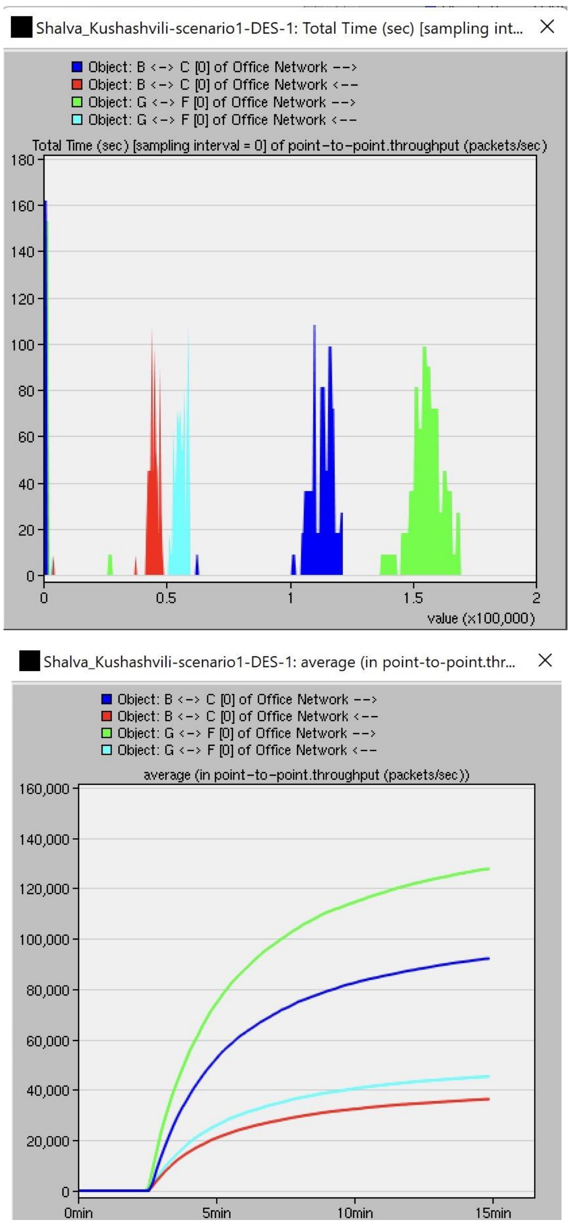

Figure 6: Histogram and Average link flow (RIP)

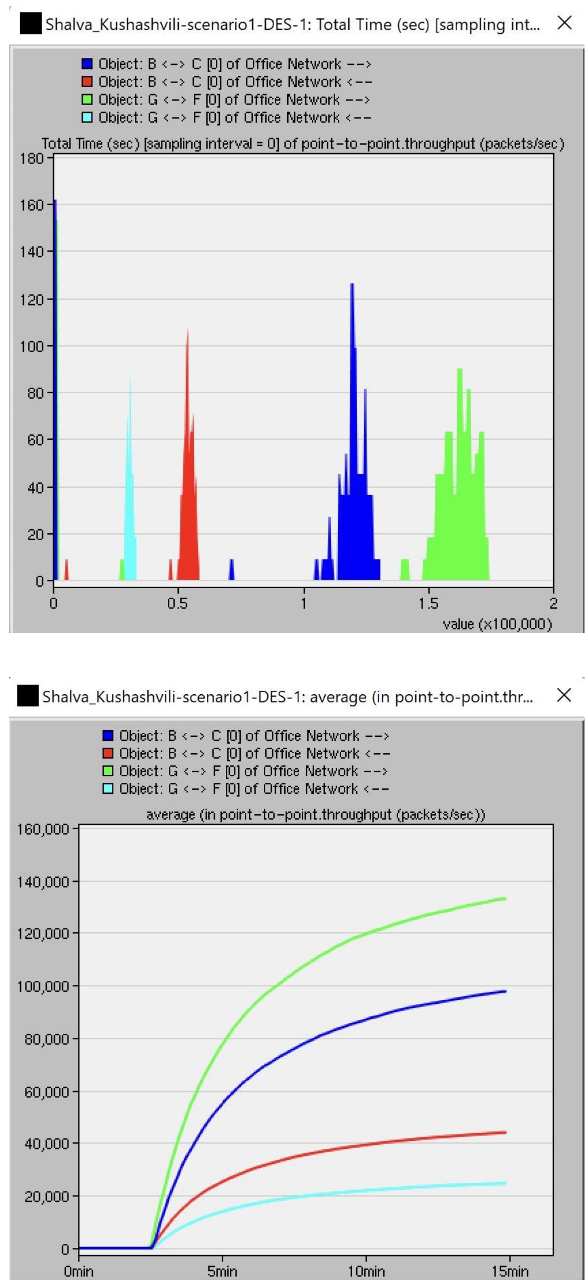

Figure 7: Histogram and Average link flow (OSPF)

As we can see from these results. the link flows are also different from each other. We can tell the total time of the link flows is also not the same. Now, Let's check the average end-to-end delay and the average number of hops in the network using these two routing algorithms.

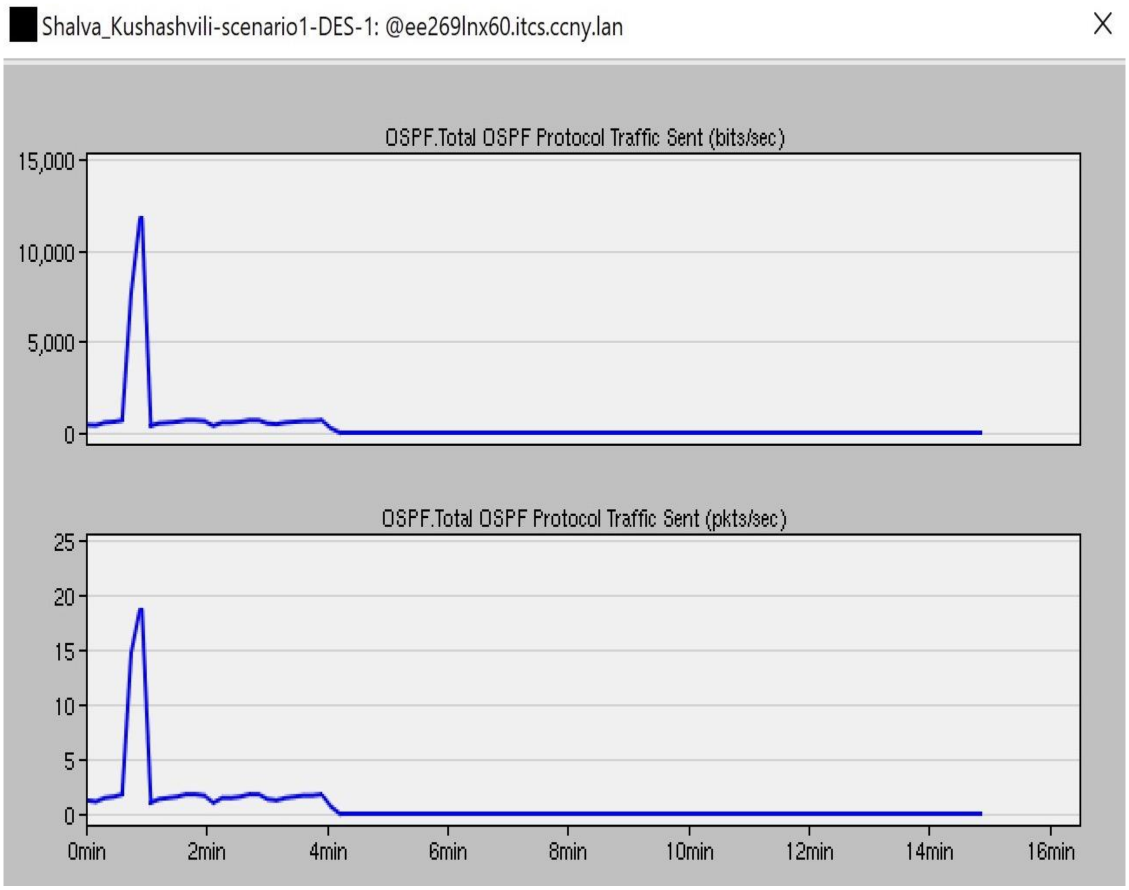

Figure 8: Traffic sent and received in OSPF

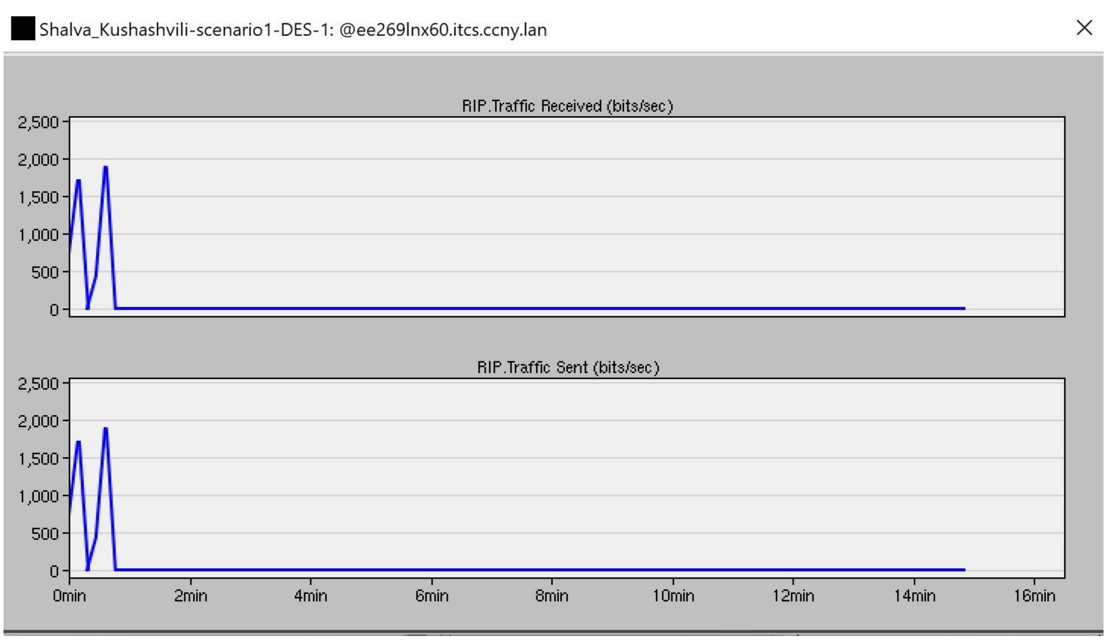

Figure 9: Traffic sent and received in RIP

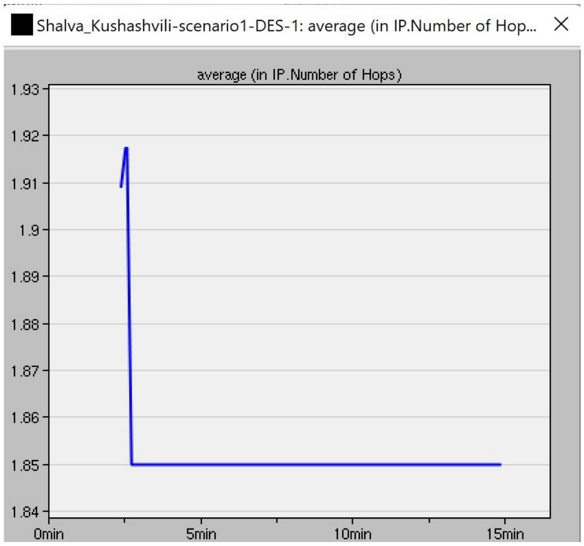

Figure 10: Average # of hops in RIP

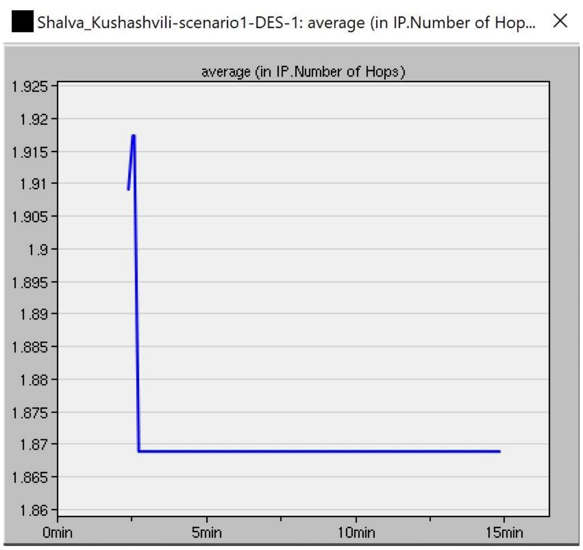

Figure 11: Average # of hops in OSPF

4. COMPARISON OF RIP AND OSPF

It's pretty clear that RIP and OSPF are distinct from each other, but let's go more in-depth about what the main differences between these two are. RIP uses the Bellman-Ford algorithm, it is a Distance Vector protocol and it uses the distance or hops count to determine the transmission path. RIP utilizes less memory compared to OSPF but it is CPU-intensive. Furthermore, a Maximum of 15 hops are allowed. Also, it Consumes a lot of bandwidth because of greater network resource requirements in sending the whole routing table. In short, RIP is ideal for small networks that are simple and non-hierarchial, however, RIP can create a traffic bottleneck as it broadcasts its updates every 30 seconds. Since any routing update in RIP take ups great bandwidth, the resources for critical IT processes are limited.

As for OSPF – it uses the Dijkstra algorithm. OSPF is a link-state protocol and it analyzes different sources like the speed, cost, and path congestion while identifying the shortest path. OSPF device resource requirements are CPU-intensive and memory. There are also no restrictions on the hop count, and it consumes less bandwidth as only part of the routing table is to send. Overall, the OSPF routing protocol has complete knowledge of network topology, allowing routers to calculate routes based on incoming requests. This fits perfectly for large and hierarchical enterprise networks, but we need to consider that OSPF demands advanced knowledge about complex networks, making it not as easy to learn as some other protocols.

CONCLUSION

In this paper, we took a look at what computer networks are, and then ran an experiment on a custom 7-node network with a network simulation tool OPNET. Last but not least, I also broke down the differences between RIP and OSPF.

REFERENCE

[1] OPNET Network Simulator (2020) *Opnet Projects*. https://opnetprojects.com/opnet-network-simulator/

[2] What is a Computer Network? *https://www.techtarget.com/searchnetworking/definition/network*