ORIGINAL RESEARCH ARTICLE | UHD JOURNAL OF SCIENCE AND TECHNOLOGY

Performance Evaluation using Spanning Tree Protocol, Rapid Spanning Tree Protocol, Per-VLAN Spanning Tree, and Multiple Spanning Tree

Rawyer Asaad Rashid[1], Dana Faiq Abd[2], Danyar Awat Othman[1], Hero Abdulqader[1]

[1]*Department of Computer Science, University of Human Development, Sulaymaniyah, Kurdistan Region, Iraq*
[2]*Department of Information Technology, University of Human Development, Sulaymaniyah, Kurdistan Region, Iraq*

ABSTRACT

This paper examines the concepts and practical applications of the spanning tree protocol (STP). It also covers per-VLAN spanning tree (PVST), multiple spanning tree (MST), and rapid STP (RSTP). Moreover, practical scenarios are presented to help the reader understand the concepts and implementations of these protocols. This study analyzes protocols using seven metrics. All protocols have been evaluated using these metrics in both small and big topology scenarios to obtain the best results. In addition, all metrics are mentioned in the introduction chapter, and the way used to apply tests on the metrics is described in the methodology chapter. Based on the experiments, different STPs performance are compared, including STP, RSTP, PVST, and MST. In summary, findings show that STP is easy to use and performs well overall, but it consistently has high latency issues. RSTP is suitable for small networks and has quick convergence, but it cannot handle as much load as STP. PVST performed the best in the experiments, as it demonstrated high scalability and the ability to handle a lot of pressure, although it requires strong hardware. However, MST did not perform as well as expected, as it struggled with delay problems and high jitter. In conclusion, it is recommended to use RSTP for simple networks that require fast convergence with dependable delay and capacity, or STP for networks that require good scaling and bandwidth. PVST is an excellent option for those who can afford high-performance hardware, while MST is suitable for simple networks or those with outdated hardware.

Index Terms: Spanning-Tree Protocol, Performance, Per-VLAN Spanning Tree, Rapid Spanning Tree Protocol, Multiple Spanning Tree, Convergence

1. INTRODUCTION

The second layer of the OSI model, known as the data link layer, encounters redundancy issues, notably the emergence of network loops when connecting numerous switches through multiple paths. In addition, these challenges necessitate effective protocols to prevent loops and ensure the stability of switched networks. Moreover, the widely employed spanning tree protocol (STP) addresses this concern by implementing a spanning tree technique, eliminating loops within switches, and restoring broken links [1]. This protocol, specified in the IEEE 802.1D standard, designates a root switch and builds a topology tree to optimize the network's efficiency [2]. Finally, the root switch is chosen based on criteria such as the lowest MAC address or priority. STP ensures a loop-free topology by blocking certain links with higher path costs to the root switch [3]. Designated ports,

Access this article online

DOI: 10.21928/uhdjst.v8n1y2024.pp20-30
E-ISSN: 2521-4217
P-ISSN: 2521-4209

Corresponding author's e-mail: Rawyer Asaad Rashid, Department of Computer Science, University of Human Development, Sulaymaniyah, Kurdistan Region, Iraq. E-mail: rawyer.rashid@uhd.edu.iq

Received: 26-11-2023 Accepted: 06-01-2024 Published: 20-01-2024

root ports, and backup ports play crucial roles in determining the network's optimal configuration [4], [5]; moreover, to enhance the convergence speed of (STP), the rapid STP (RSTP) was introduced. In addition, this improved version combines several states of (STP) into a single discarding step [6], [7], significantly reducing convergence time to (30) to (50) seconds. Moreover, the port states in RSTP include disabled, blocking, listening, learning, and forwarding [2], [8]. Recognizing the limitations of a single instance (STP) for each VLAN, Cisco introduced the per-VLAN spanning tree (PVST) protocol. In addition, PVST operates as a distinct instance for each VLAN, addressing the challenge of slowed BPDU transmission [9]. Finally, the evolution of STP continues with the introduction of the multiple STP (MSTP), described in the 802.1S standard [10]. Moreover, MSTP incorporates features from both (RSTP) and VLAN protocols, reducing the number of instances and allowing for better scalability. MSTP introduces the concept of multiple spanning tree instances and regions to organize switches with similar configuration attributes [11], [12].

However, today's publications have a problem. They used only a few metrics to evaluate these protocols, similar to combining delay and convergence to evaluate a protocol. This paper attempts to assess each of the four approaches using around seven metrics. Moreover, these measurements' outputs allow for the determination of the optimal approach in every circumstance. Moreover, neither of the papers evaluated these four protocols together. The tests in this study will be STP, PVST, RSTP, and MSTP. Finally, this paper will evaluate the proposed protocols using key metrics for evaluation including:

- Performance: Measured by factors such as packet loss, jitter, bandwidth, delay, and reliability.
- Convergence: Examining STP's reaction to network changes by reconfiguring ports for forward or block states [5].
- Congestion: Assessing the capacity of links to handle data flow.
- Delay: Measuring the time frames or packets take to reach their destination [13].
- Bandwidth Utilization: Evaluating the rate at which data can flow.
- Reliability: Perform according to its specifications.
- Scalability: Ensuring flexibility for topology growth.

These metrics form the basis for a comprehensive evaluation of the protocols' effectiveness in real-world networking scenarios.

Besides, in the following sections will briefly discuss how each protocol is assessed. Moreover, the test parameters and outcomes will be shown, also methods of measuring observations. Finally, discuss the conclusions made by the evaluators.

2. ILLUSTRATIONS OF STP, RSTP, PVST, AND MST

Figures illustrating protocols will be provided in this part so that their operation can be better understood. As can be seen, Fig. 1 (STP) represents the easiest design to comprehend and the one with the simplest loop. Finally, to shorten the path and obtain a free-loop design, one of the ports is being blocked.

Moreover, in Fig. 2, which depicts how (RSTP) operates, (RSTP) includes more ports, including backup and alternate ports. However, the concept behind STP remains the same, with the exception that RSTP converges more quickly. PVST, as depicted in Fig. 3. The root bridge for each VLAN can be a different switch. Every link is functional in PVST mode. According to VLAN 10, one of the links is blocked, in addition for VLAN 20, another link is blocked. The number (STP) instances are equal to the number of VLANs.

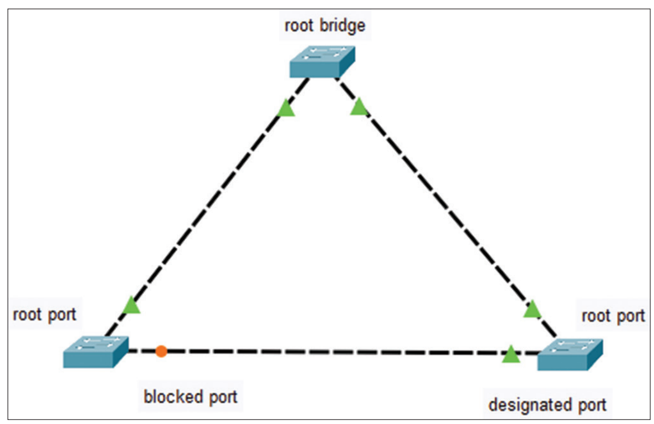

Fig. 1. An illustration of spanning tree protocol.

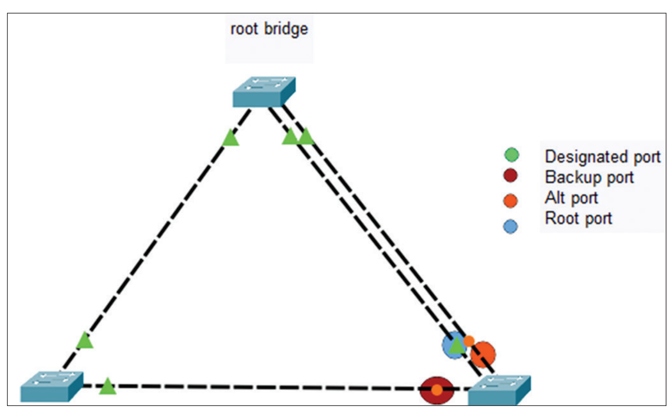

Fig. 2. An illustration of rapid spanning tree protocol.

A diagram of MST is shown in Fig. 4. Furthermore, it is well known that (MST) uses regions to divide switches with identical configurations. Moreover, the diagram makes it simple to see the regions. Those switches that have the same configurations will be in the same region.

3. LITERATURE REVIEW

The majority of papers discussing this topic only cover a small portion of the evaluation. However, the majority of the publications employed almost two evaluation protocols, or a limited set of metrics has been used for testing these protocols. This work attempts to assess seven metrics and four protocols using improved tools such as GNS3. Consequently, some of the papers cited in this work are old. Based on the findings of today's research, it can be stated that there are just a few different kinds of studies on the performance evaluation of free-loop algorithms. However, to describe the convergence time between (STP) and (RSTP), Sergio *et al.* (2018) conducted an environment using the (GNS3) tool. To calculate the topology's convergence time and connectivity loss, he needed a steady stream of data as can be seen in Tables 1 and 2.

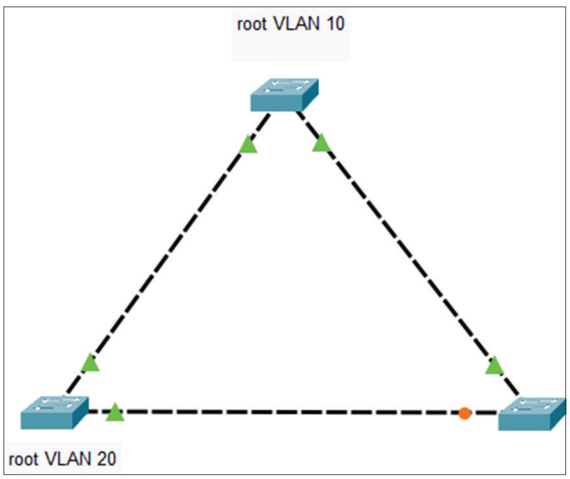

Fig. 3. An illustration of per-VLAN spanning tree.

In Sánchez-Herranz [14], moreover, according to Lapukhov *et al.* (2010), only under specific circumstances can (RSTP) convergence occur in less than one second, demonstrated by the tests in this paper, it cannot benefit from (RSTP's) fast convergence if the topology expands to include more than five bridges [15]. According to evaluations by Wang on various iterations of the STP protocol, PVST convergence is much slower than RSTP and MSTP, taking longer than 30 s as shown in Tables 3 and 4 [16].

In addition, ring topology and tree topology were the two different topologies that Pallos *et al.* (2007) tested (RSTP). Above all, this paper made use of the OPNET tool. The first ring implements topology. In addition, different versions are used to test (3, 4, 5, 6, and 7 bridges). The

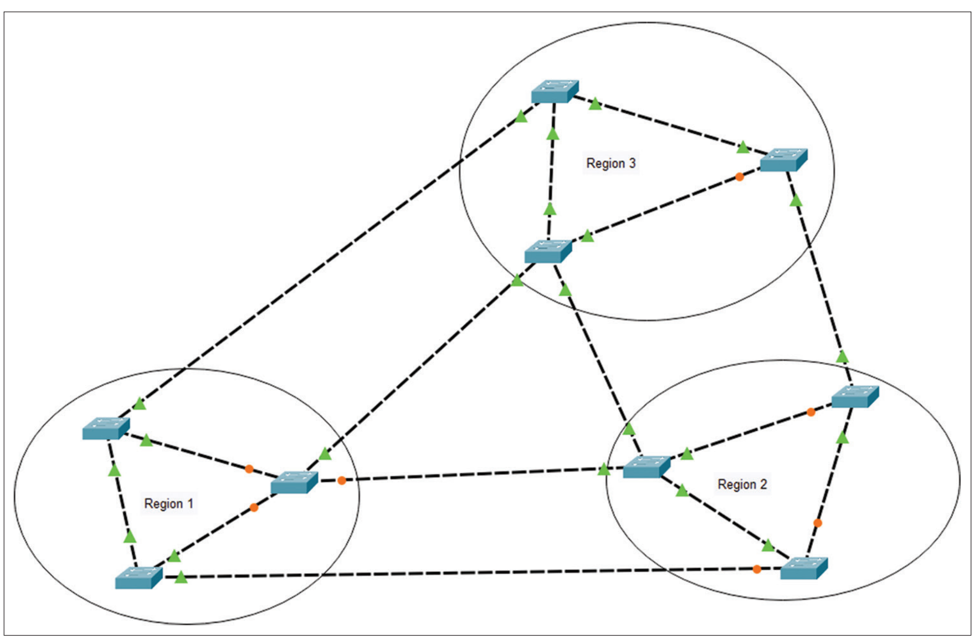

Fig. 4. An illustration of multiple spanning tree.

results demonstrate that the time for recovery increases as the number of bridges increases [17]. Moreover, according to Al-Balushi *et al.* (2012), PVST, MSTP, and RSTP are the STPs employed in this experiment. Data between an (FTP) server and a client on each (STP) mode is captured using a wire shark. The test results demonstrate that all modes may transfer data up to a maximum size of (700); however, (PVST) has captured fewer loops than (MSTP) and (RSTP). This study demonstrated that the optimal mode in terms of performance is PVST [18]. Both (RSTP) and (STP) were put to the test in various topologies by Amit noting that (STP) is more scalable and (RSTP) has a quicker convergence time but in smaller designs [19]. Furthermore, Abuguba *et al.* performed (RSTP) simulations on mesh and ring topologies for networks with 3–30 nodes in various sizes. They found that when the number of nodes grew, message loops might also grow [20]. A work by Craiova *et al.* (2013) was released to speed up (MSTP) convergence. The configurations utilized to lower (MSTP) from 50 ms to 15 ms are as follows: Maximum age is (34); forward delay is (5), and hello timer is (4). They were able to accelerate (MSTP) and even (RSTP) convergence by using various configurations on various topologies. They only mentioned that the theoretical portion of the test was conducted; the practical portion was left out [5]. Farhan *et al.* (2019) employed a spanning tree in some cases to test load balancing on (SDN) networks. The final test, which looked at bandwidth utilization, indicated that there is not a significant difference between systems that use load balancing and those that do not. This research used common (STP) for layer-2 free loop architecture to test (SDN) with load balancing more efficiently [21]. In addition, Joseph *et al.* published a paper in 2013 about testing (STP) and (RSTP) with (OPNET) simulators. The results show that (STP) and (RSTP) both respond to connection failure in roughly 160 s, with (RSTP being somewhat faster). Furthermore, (303) seconds are required for recovery in (STP). In any case, there is not much of a difference between the two protocols. RSTP is faster, taking 3–5 s [8]. Chandan *et al.* (2014) used the port channel to implement several spanning tree research studies. The findings demonstrate that employing the port channel for (STP) has a significant impact on obtaining a faster convergence time from 56 s to around 5 s, as well as having a greater bandwidth of almost twice that of common (STP) with a single physical link as shown in Table 5.

TABLE 1: STP convergence time after connectivity loss

Test number	Time in second
1	44,33
2	42,92
3	47,02

From "Performance comparison of layer 2 convergence protocols. A SDN approach". STP: Spanning tree protocol

TABLE 2: RSTP convergence time after connectivity loss

Test number	Time in second
1	5.26
2	3.42
3	4.87

From "Performance comparison of layer 2 convergence protocols. A SDN approach". RSTP: Rapid spanning tree protocol

TABLE 3: MSTP convergence time

Test number	Time in second	Average
1	6.2	
2	5.7	
3	5.8	5.98
4	6.1	

From "Measurement of spanning tree performance between different protocols: Bachelor's Thesis,". MSTP: Multiple spanning tree protocol

TABLE 4: RSTP convergence time

Test number	Time in second	average
1	4.3	
2	3.4	
3	4.1	4.0
3	4.3	

From "Measurement of Spanning Tree Performance Between Different Protocols: Bachelor's Thesis". RSTP: Rapid spanning tree protocol

Finally, Firmansyah *et al.* (2023), did a study on (STP) and (PVST) together using (Cisco packet tracer). For discovering convergence and overall network performance, the results show that (PVST) can provide faster convergence time and better load balancing. However, PVST needs more configuration and management compared to (STP) [23].

Above all, the main area of research right now is STP convergence. Although there are many crucial metrics to consider when assessing a network design, this paper will

TABLE 5: Port channel test on STP

Properties	STP with port channel	STP without port channel
Bandwidth	100 Mbps	200 Mbps
Convergence	5 s	56 s

From "Optimizing spanning tree protocol using port channel," [22]. STP: Spanning-tree protocol examine overall (STP) modes with various topologies and determine which mode is operating most effectively in all circumstances.

4. METHODOLOGY

This study used (GNS3) tool and (vIOS2) ISO for layer-2 switches to evaluate protocols. Using the cloud built within the program and building a bridge out of (GNS3)'s (VM), the topology was connected to the internet. In addition, the connection was established to the actual network and used ping to send traffic to any DNS. Moreover, DHCP server is used to assign IP addresses to the devices. In addition, the techniques to be used to gauge the metrics during the tests must be mentioned. The decision was made by using advanced technology. Accurate data is required to make decisions. However, this cannot be obtained without a large data flow. IPERF tool is being used, which is employed to determine the network's maximum practicable bandwidth. Numerous protocols, including TCP, UDP, and SCTP with IPv4 and IPv6, are supported in IPERF tool. In addition, it returns the average of all parameters along with the bandwidth, loss, and jitter for each test. IPERF has been chosen, because it can be used as a server, a tool for creating load, and a tool for monitoring. Moreover, (IPERF) functions as the client and server idea. Once the (ISO) of (IPERF) has been imported into (GNS3), two (IPERF) devices are configured and connected to the network. As shown in the Fig. 5. It 'is effortless to put this into practice.

Above all, IP addresses are manually assigned to the IPERF devices after configuring the network. Although set up one of them to serve as a client and one as a server, the command to emulate a server on a device is (iperf3 -s) as shown in Fig. 6. The server then creates a port and is prepared to respond to requests.

Finally, on the client device, this command is been used (iperf3 -c 192.168.1.2 -I 1 -t 30 -b 100 m) as shown in Fig. 7.

(-c) denotes that this device is the client, and after that assign the server's IP address. (-i) used to receive information every second. The test's period, which in this test is 30 s, is determined by the value of (-t). The bandwidth is configured with (-b), which in this example is 100 Mb. The test's results are displayed in the Fig. 8.

In summary, the report informs that 107 Mbytes of data were transported during this test with an average

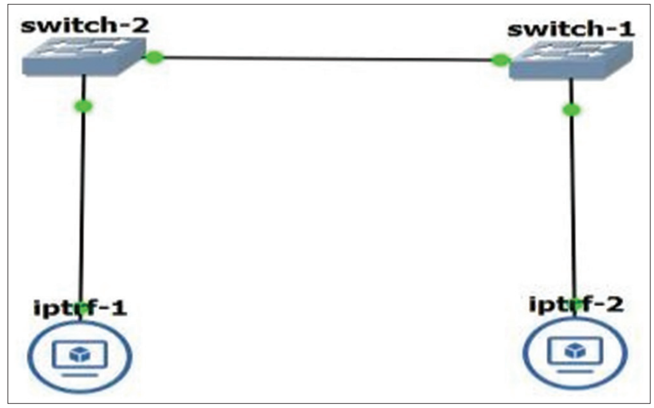

Fig. 5. Connecting IPERF servers to the network.

```
root@iperf-2:~# iperf3 -s
---------------------------------------------
Server Listening on 5201
---------------------------------------------
```

Fig. 6. iperf3 -s command for setting up the server and listening to requests.

```
root@iperf-1:~# iperf3 -c 192.168.1.2 -i 1 -t 30 -b 100m
```

Fig. 7. Command for sending packets to server from client.

bandwidth of roughly 30 Mbits/s. Because IPERF is a sophisticated tool, (-P) can send multiple client requests from various ports on the same IP. For instance, it is possible to assign (-P 10). As a result, when the test starts, the server receives requests for 10 separate devices. Moreover, using the (-u) command, which requests using (UDP) rather than (TCP), an even higher load can be created. As is well known, the network can experience a significant strain due to the use of UDP for streaming. Bandwidth, jitter, and loss on the networks can be assessed by employing (IPERF), which allows network performance decisions more effectively. Moreover, these indicators have an impact on output. In the following tests, data that are more accurate and realistic can be gathered through the utilization of appropriate methods. Finally, by evaluating the requests that fail to arrive at the desired location, other metrics, such as convergence, can be discovered. Ping command (which is a command used to ensure connectivity of two devices that use ICMP protocol) is being used with the (-t) flag to ping repeatedly see Fig. 9.

However, the average response time for a ping request is 1 s, within this time frame, there were six general failures

```
root@iptrf-1:~# iperf3 -c 192.168.1.2 -i 1 -t 30 -b 100m
Connecting to host 192.168.1.2, port 5201
[  4] local 192.168.1.1 port 40456 connected to 192.168.1.2 port 5201
[ ID] Interval           Transfer     Bandwidth       Retr  Cwnd
[  4]   0.00-1.00   sec  2.80 MBytes  23.4 Mbits/sec    0    144 KBytes
[  4]   1.00-2.01   sec  4.75 MBytes  39.6 Mbits/sec   32    112 KBytes
[  4]   2.01-3.01   sec  4.25 MBytes  35.8 Mbits/sec    2   97.6 KBytes
[  4]   3.01-4.00   sec  4.50 MBytes  37.9 Mbits/sec    0    129 KBytes
[  4]   4.00-5.01   sec  4.50 MBytes  37.5 Mbits/sec    3    115 KBytes
[  4]   5.01-6.00   sec  4.38 MBytes  37.0 Mbits/sec    3   99.0 KBytes
[  4]   6.00-10.57  sec  3.75 MBytes  6.88 Mbits/sec    0    123 KBytes
[  4]  10.57-10.57  sec  0.00 Bytes  0.00 bits/sec    0    123 KBytes
[  4]  10.57-10.57  sec  0.00 Bytes  0.00 bits/sec    0    123 KBytes
[  4]  10.57-10.57  sec  0.00 Bytes  0.00 bits/sec    0    123 KBytes
[  4]  10.57-11.00  sec  1.75 MBytes  34.5 Mbits/sec    3    126 KBytes
[  4]  11.00-12.00  sec  4.12 MBytes  34.5 Mbits/sec    0    122 KBytes
[  4]  12.00-13.00  sec  4.00 MBytes  33.6 Mbits/sec    1    106 KBytes
[  4]  13.00-14.00  sec  4.75 MBytes  39.9 Mbits/sec    2   93.3 KBytes
[  4]  14.00-15.00  sec  4.00 MBytes  33.6 Mbits/sec    0    123 KBytes
[  4]  15.00-16.00  sec  4.00 MBytes  33.5 Mbits/sec    1    107 KBytes
[  4]  16.00-17.00  sec  3.25 MBytes  27.3 Mbits/sec    0    127 KBytes
[  4]  17.00-18.00  sec  4.12 MBytes  34.6 Mbits/sec    3    113 KBytes
[  4]  18.00-19.00  sec  4.25 MBytes  35.7 Mbits/sec    1   99.0 KBytes
[  4]  19.00-20.00  sec  4.62 MBytes  38.8 Mbits/sec    0    130 KBytes
[  4]  20.00-21.00  sec  3.88 MBytes  32.5 Mbits/sec    1    117 KBytes
[  4]  21.00-22.00  sec  3.38 MBytes  28.3 Mbits/sec    0    137 KBytes
[  4]  22.00-23.00  sec  3.75 MBytes  31.5 Mbits/sec    2    116 KBytes
[  4]  23.00-24.00  sec  3.88 MBytes  32.4 Mbits/sec    1    102 KBytes
[  4]  24.00-25.00  sec  3.12 MBytes  26.2 Mbits/sec    0    122 KBytes
[  4]  25.00-26.00  sec  2.75 MBytes  23.1 Mbits/sec    1    105 KBytes
[  4]  26.00-27.00  sec  4.12 MBytes  34.5 Mbits/sec    0    123 KBytes
[  4]  27.00-28.00  sec  4.25 MBytes  35.7 Mbits/sec    1    110 KBytes
[  4]  28.00-29.00  sec  5.50 MBytes  46.2 Mbits/sec    1    105 KBytes
[  4]  29.00-30.00  sec  4.50 MBytes  37.7 Mbits/sec    0    133 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth       Retr
[  4]   0.00-30.00  sec   107 MBytes  29.9 Mbits/sec   58             sender
[  4]   0.00-30.00  sec   106 MBytes  29.7 Mbits/sec                  receiver
```

Fig. 8. Results shown after a connection between IPEERF client and server happens.

```
C:\Users\Rawye>ping 8.8.8.8 -t

Pinging 8.8.8.8 with 32 bytes of data:
Reply from 8.8.8.8: bytes=32 time=196ms TTL=55
Reply from 8.8.8.8: bytes=32 time=108ms TTL=55
Reply from 8.8.8.8: bytes=32 time=185ms TTL=55
General failure.
General failure.
General failure.
General failure.
General failure.
General failure.
Reply from 8.8.8.8: bytes=32 time=70ms TTL=55
Reply from 8.8.8.8: bytes=32 time=120ms TTL=55
Reply from 8.8.8.8: bytes=32 time=66ms TTL=55
Reply from 8.8.8.8: bytes=32 time=63ms TTL=55
Reply from 8.8.8.8: bytes=32 time=60ms TTL=55
```

Fig. 9. Discovering convergence by calculating the time connection loss.

(requests timed out). It can be inferred from this that there was a disconnection from the internet for approximately 6 s. Convergence in STP is measured using the same methodology. The topology was manually adjusted and made a ping (-t), finally timed-out requests will be measured till the requests get a replay once more. No switch path requests were sent throughout the convergence period. This period occurred when STP rebuilt the topology (to put it simply, the internet ceased to function). To enable pinging and receive accurate data from the internet, DHCP is used to link the devices to the cloud and assign an IP address to them. Based on a comparison of the output from the small and big topologies, reliability and scalability must be determined. In addition, the final issue is congestion; a massive load must be created utilizing (UDP) and roughly 100 parallel networks. They are all making data requests at once.

5. EVALUATION

5.1. Test (STP)

Testing STP is the first test that is going to be done. Because the configuration of this protocol is the simplest. In addition, 50 parallel networks were employed. UDP protocol was used. A bandwidth of 1000 Mbits was requested for around 60 s. Moreover, to measure actual (delay) data, the network was connected to the cloud. Finally, another test was done by employing 100 parallel networks to create a massive load that lasted 100 s. In summary, (STP) can manage enormous amounts of data.

The small and large test topology is shown in Figs. 10 and 11 see Tables 6 and 7 for LAN characteristics.

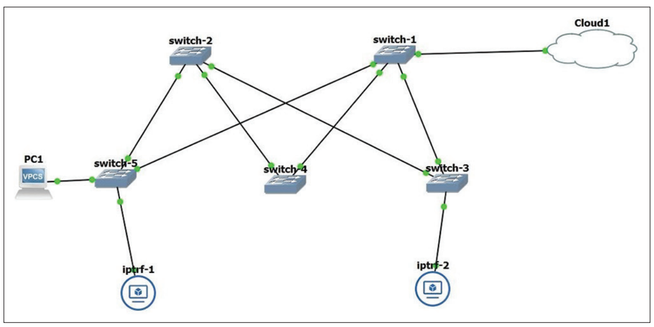

Fig. 10. Small topology.

The data measured on (STP) for both small and big topologies are shown in Table 8.

The chart demonstrates that (STP) can give an excellent result overall. The huge topology has reduced delay; 1-s change in convergence time is an excellent result. It scales very well. However, the jitter's output reveals that there has been a significant change. It is crucial to note how well 100 parallel networks performed on (STP). The only issue with (STP) is the delay and jitter, which for small networks is a little high at 146 ms.

5.2. Test (RSTP)

However, testing (RSTP) is the second test. The same (IPERF) parameters are used for (STP). Moreover, the test

TABLE 6: LAN characteristics of Figure 10

Device	LAN characteristics	Link capacity
Iperfe-1	Ip: 192.168.1.1	Gigabit Ethernet
Iperf-2	Ip: 192.168.1.2	Gigabit Ethernet
PC1	DHCP	Gigabit Ethernet

TABLE 7: LAN characteristics of Figure 11

Device	LAN characteristics	Link capacity
Iperfe-1	Ip: 192.168.1.1	Gigabit Ethernet
Iperf-2	Ip: 192.168.1.2	Gigabit Ethernet
PC1	DHCP	Gigabit Ethernet
PC2	DHCP	Gigabit Ethernet

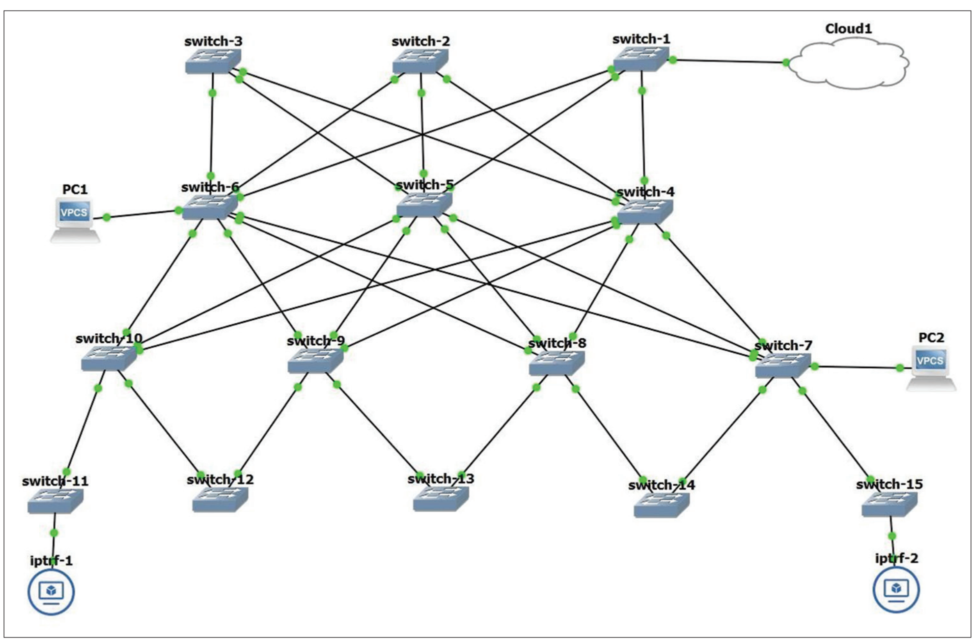

Fig. 11. Large topology.

will last 60 seconds. It has a combined bandwidth of 1000 Mbits and 50 parallel networks. Plus, the (UDP) protocol. First, the compact design was set with five switches and two (IPERF) components. In addition, the cloud-assisted in identifying (delays) in both topologies. Similar to (STP), 100 simultaneous networks are requested with a bandwidth of 1000 Mbits to detect congestion. The test was not finished since connection was lost on devices, in summary (RSTP) cannot handle that much demand because it has repeatedly lost connection.

The data measured on (RSTP) for both small and big topologies are shown in Table 9.

Table 9 shows that (RSTP) performs well in delay and convergence. However, for other metrics, such as jitter and congestion, it is arguable that (RSTP) cannot be relied on for large networks. Moreover, it can be beneficial for small networks that do not need to expand in the future. Another issue with RSTP is a connection loss, which caused it to repeatedly disconnect from the cloud while running the ping command (ping 8.8.8.8 -t). However, it did not perform well in the congestion test. Finally, the small design transferred 143 Gigabytes of data. This amount is very good but not particularly useful due to connection loss issues.

5.3. Test (PVST)

First, PVST was set up by turning all links into trunks, except those that connected to end devices. Moreover, IPERF devices' IP addresses are assigned and configured. One of them is set to be the server and another as the client. As in STP and RSTP, the command (iperf3 -c 192.168.1.1 -I 1 -t 60 -u -b 1000m -P 50) is used.

The data measured on PVST for both small and big topologies are shown in Table 10.

These results indicate that (PVST) fared the best across all fields. Many metrics in the largetopology decreased, indicating that (PVST) has a very high degree of scalability. In summary, the delay decreased from 121 ms to 77 ms, and congestion decreased from 32 s to 26 s. Jitter increased by 1.642. Moreover, it operates best when all 100 networks are

TABLE 8: STP results

Metrics	Small topology	Large topology
Delay	146.875 MS	79.227 MS
Convergence	60 s	61 s
Reliability	Good	
Scalability	Very good	
Performance	Good	
Jitter	6.571 MS	11.738 MS
Loss	99.9915% of packets reached the destination (9819639/9820475)	99.9901% of packets reached the destination (10214613/10215625)
Congestion	Very good	
Bandwidth Utilization	127 Gigabytes transferred with an average bandwidth of 18.1 Gigabits/s*	128 Gigabytes transferred with an average bandwidth of 18.4 Gigabits/s

*50 parallel networks are used, when 18.1 is divided by 50, it gets the number of bits that y network used.

TABLE 9: RSTP results

Metrics	Small topology	Large topology
Delay	97.07 ms	88.57 ms
Convergence	19 s	23 s
Reliability	Bad (connection loss problem)	
Scalability	Bad	
Performance	Normal	
Jitter	3.844 ms	12.947 ms
Loss	99.9911% of packets reached the destination (11240599/11241598)	99.9889% of packets reached the destination (9326858/9327883)
Congestion	Very bad	
Bandwidth utilization	143 Gigabytes transferred with an average bandwidth of 20.4 Gigabits/s*	115 Gigabytes transferred with an average bandwidth of 16.4 Gigabits/s

STP: Spanning tree protocol

simultaneously requested. Given its excellent capabilities, PVST can be used for sophisticated designs and networks.

5.4. Test (MST)

In the beginning, the network was inaccessible even with switch configurations. However, MST test was conducted several times. This indicates that the network requires time to prepare for use. The examination was extended for an additional 10–15 min. Moreover, the results of the mini and large tests are shown in Table 11. However, (delay) encountered difficulty during testing since the connection was repeatedly lost. In addition, the design went completely down while 50 parallel networks had been used. Finally, the test had to be stopped. In addition, 100 parallel requests using

TABLE 10: PVST results

Metrics	Small topology	Large topology
Delay	121.051 MS	77.214 MS
Convergence	32 s	26 s
Reliability	Very good	
Scalability	Very good	
Performance	Very good	
Jitter	6.873 MS	8.515 MS
Loss	99.9897% of packets reached the destination (10274683/10275735)	99.9910% of Packets reached the destination (10680190/10681153)
Congestion	Very good	
Bandwidth utilization	127 Gigabytes transferred with an average bandwidth of 18.1 Gigabits/s	132 Gigabytes of data transferred with an average bandwidth of 18.9 Gigabits/s

PVST: Per‑VLAN spanning tree

TABLE 11: MST results

Metrics	Small topology	Large topology
Delay	143.98 ms	147.39 ms
Convergence	49 s	25 s
Reliability	Bad (good for simple topology)	
Scalability	Very bad	
Performance	Normal	
Jitter	12.424 ms	0.795 ms
Loss	99.9936% of packets reached the destination (14558717/14559654)	99.5300% of Packets reached the destination (555976/558612)
Congestion	Very bad (Gets down immediately)	
Bandwidth utilization	127 Gigabytes transferred with an average bandwidth of 18.1 Gigabits/s	6.97 Gigabytes of data transferred with an average. The bandwidth of 998 Megabits/s

MST: Multiple spanning tree

TABLE 12: STP, RSTP, PVST, and MST results

STP MODE	STP		RSTP		PVST		MST	
	Small	Large	Small	Large	Small	Large	Small	Large
Delay	146.875ms	79.227ms	97.07ms	88.57ms	121.051ms	77.214ms	143.98ms	147.39ms
Convergence	60s	61s	19s	23s	32s	26s	49s	25s
Reliability	Good		Bad		Very good		Bad	
Scalability	Very good		Bad		Very good		Very bad	
Performance	Good		Normal		Very good		Normal	
Jitter	6.571 ms	11.738 ms	3.844 ms	12.947 ms	6.873 ms	8.515 ms	12.424 ms	0.795 ms
Loss (% reached packets)	99.9915%	99.9901%	99.9911%	99.9889%	99.9897%	99.9897%	99.9936%	99.5300%
Congestion	Very good		Very bad		Very good		Very bad	
Bandwidth utilization	18.1 Gigabits/s	18.4 Gigabits/s	20.4 Gigabits/s	16.4 Gigabits/s	18.1 Gigabits/s	18.9 Gigabits/s	18.1 Gigabits/s	998 Megabits/s
Transferred (throughput)	127GB	128GB	143GB	115GB	127GB	132GB	127 GB	6.97 GB

STP: Spanning tree protocol, RSTP: Rapid spanning tree protocol, PVST: Per‑VLAN spanning tree, MST: Multiple spanning tree

TCP were tested and there was no issue. In summary, UDP was used for MST with a single parallel network.

The data measured on MST for small and big topologies are shown in Table 11.

MSTs performance fell short of what was predicted in the other articles. It scales poorly and has a large delay. In addition, due to very poor congestion capacity, just one parallel network was used for all MST tests on big designs. However, using more than one network caused it to lose connectivity. The record for the worst outcome was broken by the jitter in the small network.

The data that were measured on (all modes) for both small and big topologies are shown in Table 12.

A comparison of all modes for all metrics with small and large design tests is shown in the table. According to the findings, PVST can be claimed to be in the top spot. As can be seen in the table, it had the best performance across all criteria. Moreover, STP and PVST worked best for congestion. Regarding the delay, RSTP comes in first for small topologies. Moreover, for large topologies, PVST comes in first. PVST or STP can be options if a network needs to be designed to be scalable. In addition, RSTP is recommended for throughput. However, due to the issue of connectivity loss, PVST can be helpful in that area. In terms of packet loss, all modes fared the best. The best performance, according to the statistics in the table, belongs to PVST.

6. CONCLUSION

STP, MST, PVST, and RSTP examinations produced several significant conclusions. Above all, it was discovered that each of these protocols has distinct advantages and disadvantages and that the ideal protocol to use depends on the particular needs of the network. In summary, the test results show that STP is easy to use and achieves average or even good performance overall. However, it has an issue with delay, which consistently has a high latency. On the other hand, RSTP performs well on small networks and has quick convergence, which is a benefit, but it cannot support as much load as STP. In addition, PVST performed best in the experiments. It is notable from its outputs that it can withstand a lot of pressure and is quite scalable. The only issue with PVST is that it requires strong hardware. PVST is a good option if a very high-performance network design is needed. While MST is thought of as a fairly dependable way of operating, but sadly, MST did not perform as well as it had anticipated. It cannot handle large amounts of data and has delay problems as well as very high jitter. Finally, while no mode is 100% flawless, it is suggestible to use (RSTP) if the network is simple and has fast convergence with dependable delay and capacity. Moreover, STP can be utilized if latency is not focused on. Because it can bear pressure during irregular times and is good at scaling and bandwidth. Finally, PVST can be chosen if pricey, cutting-edge hardware can be afforded. It is incredibly high-performance and capable and serves well in every area. MST applies to both simple networks and networks with outdated or subpar hardware that cannot execute at a high level of performance.

REFERENCES

[1] I. A. Alimi. “Bandwidth management and loop prevention in redundant networks”. *The American Journal of Medicine*, vol. 2, pp. 1-12, 2016.

[2] M. S. Guragain. “*Evaluating the Future of the Spanning Tree Protocol*”. Sybex, United States, 2016.

[3] F. Shahriar and J. Fan. “*Performance Analysis of FHRP in a VLAN Network with STP. In: EEE 3rd International Conference on Electronics Technology (ICET), Chengdu, China*”. 2020.

[4] G. Mirjalily. “*Balanced Spanning Tree Protocol in Hierarchical NextGen Ethernet Networks*”. Preprint, 2022.

[5] R. Stanica and A. Informatics. “Improving network convergence with multiple spanning tree protocol”. vol. 15, no. 1, CEAI, United States, pp. 79-87, 2013.

[6] S. Kasu, L. Hash, J. Marsh and R. Bull. “*Spanning Tree Protocol*”. State University of New York Polytechnic Institute, New York, 2015.

[7] J. Maia, C. A. Severiano, F. G. Guimarães, C. L. Castro, A. Lemos, J. C. F. Galindo and M. Weiss. Evolving clustering algorithm based on a mixture of typicalities for stream data mining”. *Future Generation Computer Systems*, vol. 106, pp. 672-684, 2020.

[8] J. Lu, S. Jiang and T. Xiong. “*Evaluation and Comparison of Spanning Tree Protocol and Rapid Spanning Tree Protocol on Cisco Switches via OPNET”*. ENSC 427: Communication Networks Report, 2013.

[9] N. Al-Balushi, R. Al-Klabani and F. Hajmohideen. “Performance evaluation using STP across layer 2 VLANs”. *International Journal of Communication and Networking System*, vol. 1, pp. 1-8, 2012.

[10] S. Kasu, L. Hash, J. Marsh and R. Bull. “*Spanning Tree Protocol*”. State University of New York Polytechnic Institute, New York, 2015.

[11] A. de Sousa and G. Soares. “*Improving Load Balance and Minimizing Service Disruption on Ethernet Networks with IEEE 802.1 S MSTP. In: Workshop on IP QoS and Traffic Control*”. pp. 25-35, 2007.

[12] J. Chen, X. Zheng, L. Zhang, L. Peng and X. Liu. “MSTP protocol simulation experiment based on ENSP. *Journal of Physics: Conference Series*, vol. 1486, no. 7, p. 072029, 2020.

[13] A. Agarwal. “*Limits on Interconnection Network Performance. In: IEEE Transactions on Parallel and Distributed Systems*”. vol. 2, no. 4, pp. 398-412, 1991.

[14] S. Sánchez Herranz. “*Performance Comparison of Layer 2*

Convergence Protocols. A SDN Approach". (Master's Thesis), 2018.

[15] P. Lapukhov. "*Understanding STP and RSTP Convergence"*. INE Blog Documents, 2010.

[16] Z. Wang. "*Measurement of Spanning Tree Performance between Different Protocols*". (Bachelor's Thesis), 2014.

[17] R. Pallos, J. Farkas, I. Moldován and C. Lukovszki. "*Performance of Rapid Spanning Tree Protocol in Access and Metro Networks, In: 2007 Second International Conference on Access Networks and Workshops*". IEEE, New Jersey, pp. 1-8, 2007.

[18] N. Al-Balushi, R. Al-Klabani and F. Hajmohideen. "Performance evaluation using STP across layer 2 VLANs". *International Journal of Communication and Networking System*, vol. 1, pp. 1-8, 2012.

[19] B. B. Amit Kumar. "Performance evaluation of spanning tree protocol and rapid spanning tree protocol in computer networks". *IJSRD - International Journal for Scientific Research and Development*, vol. 7, no. 6, pp.214-217, 2019.

[20] S. Abuguba, I. Moldován and C. J. B. E. Lukovszki. "*Verification of RSTP Convergence and Scalability by Measurements and Simulations, In: Proceedings Broad Band Europe 2006"*. CISCO, United States, 2006.

[21] F. Chahlaoui, M. R. El-Fenni and H. Dahmouni. "*Performance Analysis of Load Balancing Mechanisms in SDN Networks, In: Proceedings of the 2nd International Conference on Networking, Information Systems and Security*", pp. 1-8, 2019.

[22] Y. N. Krishnan, C. N. Bhagwat and A. P. Utpat. "*Optimizing Spanning Tree Protocol Using the Port Channel, In: 2014 International Conference on Electronics and Communication Systems* (*ICECS*)", IEEE, New Jersey, pp. 1-5, 2014.

[23] F. Firmansyah, T. A. A. Sandi, A. Fauzi and R. S. Anwar. "Analisis performa redundancy link menggunakan metode spanning tree protocol dan per VLAN spanning tree". *Jurnal Infortech*, vol. 5, no. 1, pp. 47-52, 2023.