

Wang Zejia

MEASUREMENT OF SPANNING
TREE PERFORMANCE BETWEEN
DIFFERENT PROTOCOLS


Bachelor's Thesis
Networking

May 2014



MAMK
University of Applied Sciences

DESCRIPTION

 <p>MAMK University of Applied Sciences.</p>		Date of the bachelor's thesis May 27, 2014	
Author(s) Wang Zejia		Degree programme and option Information Technology Networking	
Name of the bachelor's thesis Measurement of spanning tree performance between different protocols			
Abstract Network is becoming more and more important in daily life, the performance of the network is most critical issue. In this article, we will talk about the spanning tree protocol which is used to prevent bridge loops, broadcast radiation and provide the protection of convergence in order to provide the nice performance of the network. You can see from this article about what it is the spanning tree, how it works, and what kind of enhanced protocols are used. I will test and measure the original protocol and enhanced protocols, and show the results of the performance of convergence time. We will see which protocol have nice performance. After that we can decide which protocol is suit for our network and provide the best performance.			
Subject headings, (keywords) STP, PVST+, RSTP, MSTP, Performance			
Pages 70	Language English	URN	
Remarks, notes on appendices			
Tutor Matti Koivisto		Employer of the bachelor's thesis MAMK	

CONTENTS

1 INTRODUCTION	1
2 ORIGINAL SPANNING TREE PROTOCOL.....	2
2.1 STP Basic Components	2
2.2 Spanning Tree Algorithm.....	2
2.3 STP Port	2
2.3.1 Port Roles	3
2.3.2 Port States.....	4
2.4 The Root Bridge and BID.....	6
2.4.1 BID Fields	6
2.4.2 Extend System ID.....	7
2.4.3 Bridge Priority.....	8
2.4.4 MAC Address.....	8
2.5 BPDU	9
2.5.1 BPDU Fields	9
2.5.2 BPDU Timers.....	10
3 ENHANCED STP	11
3.1 Per VLAN Spanning Tree	12
3.2 Per VLAN Spanning Tree Plus	13
3.3 Rapid Spanning Tree Protocol	15
3.4 Multiple Spanning Tree Protocol	18
3.4.1 Instance and Region	18
3.4.2 CIST Root and Region Root	20
3.4.3 Path Cost and Ports	20
4 MEASUREMENT DESIGN	21
5 DESIGN IMPLEMENTATION AND MEASUREMENT	22
5.1 Performance Measurement of PVST+	22
5.1.1 Implementing PVST+	23
5.1.2 Test and Result.....	24
5.2 Performance Measurement of RSTP.....	26
5.2.1 Implementing RSTP.....	27
5.2.2 Test and Result.....	27
5.3 Performance Measurement of MSTP	29

5.3.1 Implementing MSTP	30
5.3.2 Test and Result	31
6 CONCLUSION.....	32
BIBLIOGRAPHY.....	33
APPENDIX	34

1 INTRODUCTION

In local area networks we want to have redundant paths. This way we can in the cases of a link failure redirect the traffic to the alternative path. However for technical reasons we cannot have loops in a LAN topology. Therefore we have introduced a Spanning Tree Protocol (STP). The STP generates the spanning tree to ensure that Ethernet local area network works with loop-free topology.

STP has two main functions: one is in the use of spanning tree algorithm to prevent bridge loops and the broadcast radiation. The second is in the Ethernet network topology change, through the spanning tree protocol to achieve the purpose of protection of convergence (ZhangCun, 13). Bridges exchange BPDU data to monitoring loops, and then close the selected bridge interfaces cancel the loop. STP is standardized as IEEE802.1D.

STP has one limitation and it is the time required to do topology changes. In current media rich networks a faster solution is needed. The aim of the study is to get familiar with original and more advanced spanning tree protocols. In the practical part of my thesis I will implement original STP, Rapid Spanning Tree Protocols and Multiple Spanning Tree Protocol in a network. I will identify the operation principle for each protocol, the performance for different configuration. I will also analyze which protocol is suitable choice for different environments.

The structure of the study is as follow. In Chapter 2, the original STP is explained. The theoretical and the working principle are introduced there. Chapter 3 is the theoretical part of the enhanced STPs and their working principle. We talk about more advanced functions of enhanced STPs. In Chapter 4 the measurement design is described for original STP and enhanced STPs. Chapter 5 is the configuration and measurement part for the test. The details of configurations and measurement result will be shown in the chapter. Chapter 6 dedicates to give a conclusion and summary of my thesis.

2 ORIGINAL SPANNING TREE PROTOCOL

2.1 STP Basic Components

Like other protocols, STP is constantly upgrading with the continuous development of the network. STP is invented by the famous engineer Radia Perlman from Sun Microsystems Inc (Radia Perlman, 2014). By this method, the bridge can achieve the ideal of layer 2 routing with redundant and loop free operation. We can imagine an STP as a bridge device in mind for the process of optimization and fault tolerance to send data. STP is defined in the IEEE 802.1D (ZhangCun, 13). It is a link management protocol for network to provide path redundancy while preventing undesirable loops. To make Ethernet work better, it can only have one active path between two workstations. Network loops occur for variety of reasons, the most common one is intentionally generated redundancy that in case of a link or switch fails, and there will be another link or switch to replace it.

Now we firstly get familiar with the original STP. The basic components of STP are the STP algorithm, a bridge ID, a bridge protocol data unit, BPDU timers, path cost and port states. I will explain all of them in more details below.

2.2 Spanning Tree Algorithm

In order to guarantee the continuous network traffic, the network is designed with redundant links and devices. The purpose of this design is to prevent a failure to lead to loss of functionality of the entire network. Although the redundant design can eliminate single point of failure problems, but also it leads loops to occur, broadcast storms, multiple copies of the same frame and unstable MAC address tables in the network topology. When implementing the STP that there is only one logical path on the network. STP uses the Spanning Tree Algorithm (STA) to determine which ports in a switch need to be blocked in order to prevent loops. Therefore, a switched network needs to have a mechanism to prevent the loops (SPANNING TREE, 2014).

2.3 STP Port

2.3.1 Port Roles

There are four port roles in STP. They are described below in more details.

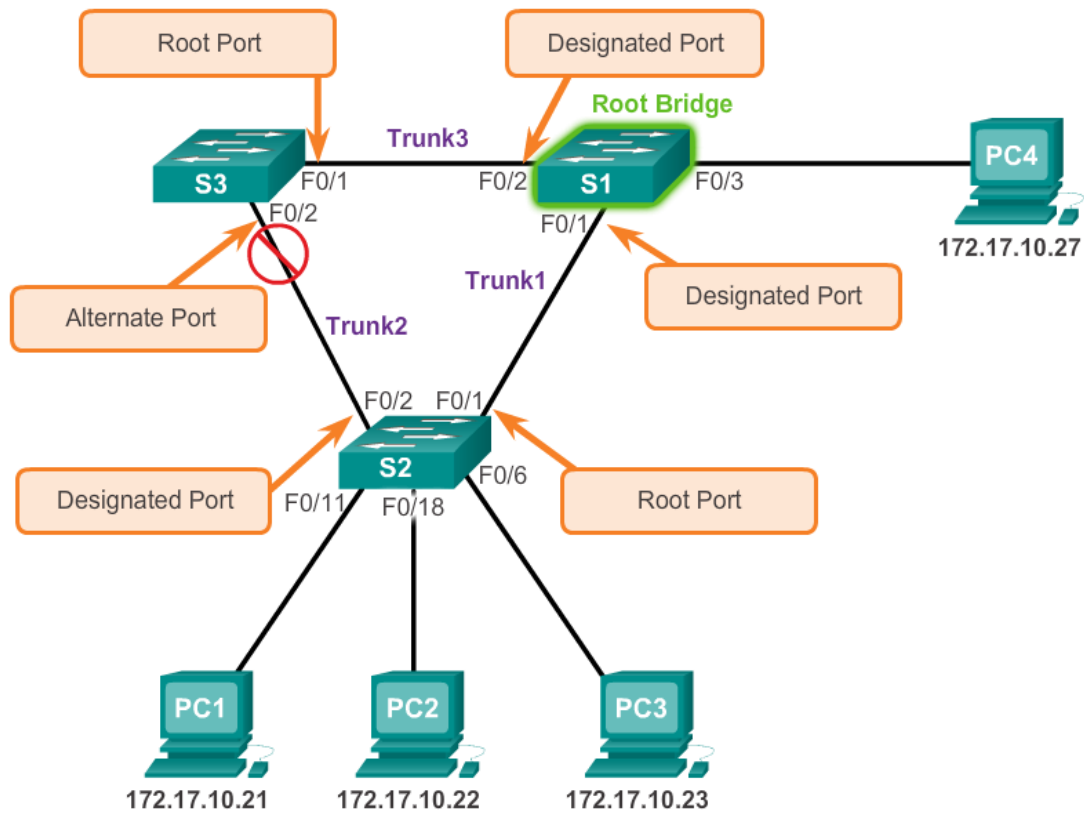


Figure 1 Port Roles (Cisco, LAN Redundancy, 2013)

Root ports: The root port exists on the non-root bridge. The port has the best path to the root bridge. Root port forwards traffic to the root bridge. The root port uses received frame source MAC address to populate the MAC table. One bridge can have only one root port. In Figure 1, the root port is F0/1 on switch S2 with configured for the trunk link between switch S2 and switch S1 (Cisco, LAN Redundancy, 2013).

Designated ports: The designated port exists on the root bridge and non-root bridges. All ports of the root bridge are designated ports. For non-root bridge, designated port is needed to receive frames of forward frames towards the root bridge. In Figure 1, switch ports F0/1 and F0/2 on switch S1 are designated ports (Cisco, LAN Redundancy, 2013).

Alternate port and backup ports: All ports configured to be in a blocking state to prevent loops. The port does not forward the frames, it will not use the source address populate to the MAC address table. In Figure 1, the STP configured port F0/2 on switch S3 in the alternate role (Cisco, LAN Redundancy, 2013).

Disabled port: The disabled port is administratively shut down. The disabled port does not participate in the spanning tree process. In Figure 1, there is no disabled port (Cisco, LAN Redundancy, 2013).

2.3.2 Port States

After the startup of the switches, the spanning tree algorithm immediately determines the states of the ports. If a switch port goes directly from blocking to forwarding state, the switches do not understand the topology of the network and a port may cause temporary data loops. Therefore, STP has introduced five port states (Hurgh.org, 2014). The working principle is shown in Figure 2.

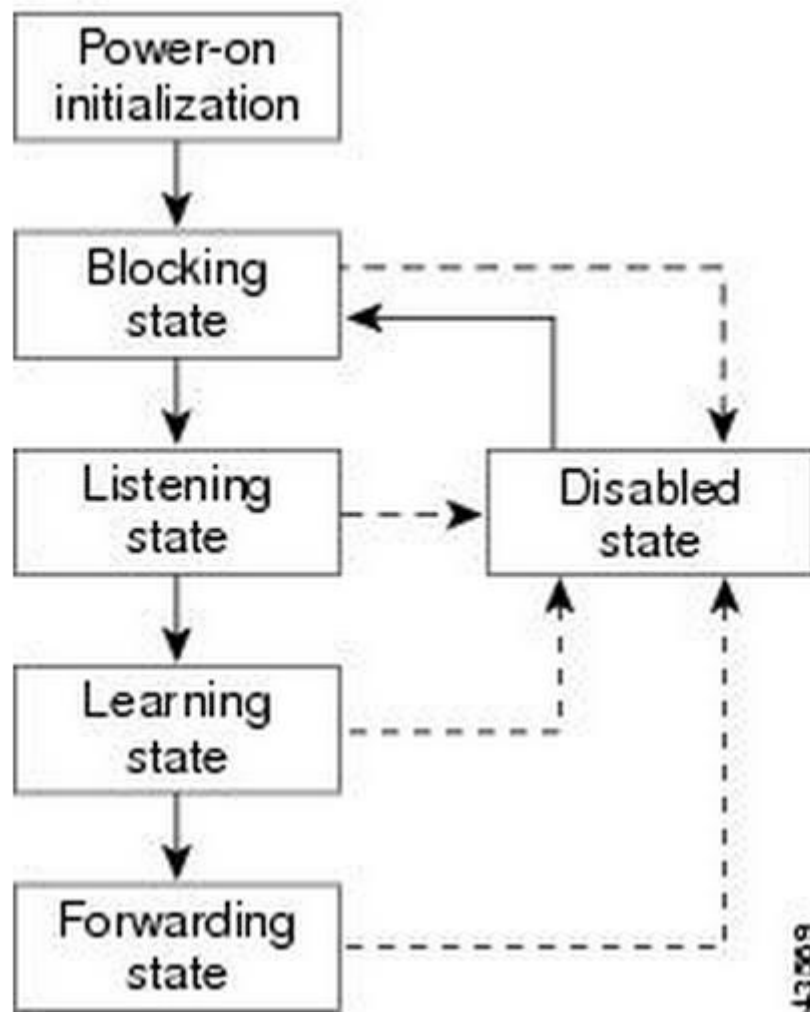


Figure 2 Spanning Tree Port States (Hurgh.org, 2014)

As shown in Figure 2, when a switch is powered up, by default, STP is enabled, and every interface in the switch, VLAN or network goes through the blocking, listening and learning states. After that each interface stabilizes at the forwarding state or blocking state. The details of each state is described below.

Blocking state: The port is alternate port. In blocking state, the interface does not participate in any frame forwarding. The port receives BPDU frames to determine root bridge switch location and root ID (Hurgh.org, 2014).

Listening state: According to received BPDU frames, STP determines if the port can participate in frame forwarding. At this point, the port not only receives the BPDU frames, but also forwards its own BPDU frames, and notifies the adjacent switches that this switch port is preparing to participate in the activities topology (Hurgh.org, 2014).

Learning state: The port is ready to participate in frame forwarding and begin to populate the MAC address table (Hurgh.org, 2014).

Forwarding state: The port is part of the active topology. It will forward frames and sends and receives BPDU frames (Hurgh.org, 2014).

Disabled state: The layer 2 port does not participate in spanning tree and does not forward frames. When administrator closes the port, the port is the disabled state (Hurgh.org, 2014).

2.4 The Root Bridge and BID

In STP, the root bridge is a bridge that exchanges topology information with the designated bridge. When need to change the network topology, the designated bridge notify all the other network bridges in the execution of a spanning tree. The root bridge election is based on the bridge ID (BID) the bridge priority (can be modified) and the MAC address (cannot be modified by user) (Cisco, LAN Redundancy, 2013).

When the switches start for the first time, each switch assumes to be the root bridge, and forward their Bridge Protocol Data Unit (BPDU) frames. Each switch analyzes the BPDU frames. Both the BID and MAC address are used to elect the root bridge. They first compare the BID, if the BID are the same, then compare the MAC address. After a period of time, spanning tree convergence, all switches agree to a particular bridge which has the highest priority to be the root bridge. If a new switch with a higher priority joins the network, it first announces itself as the root bridge. After communication with other switches, it will be recorded as a new root bridge (Cisco, LAN Redundancy, 2013).

2.4.1 BID Fields

The first type of BID is without the extended system ID. The structure is shown in Figure 3, and it consists of Bridge Priority (2 bytes) + MAC Address (6 bytes). In early STP implementations there were no virtual LANs (VLANs) in use. There was a single original spanning tree across all switches (Chapter 5 - Spanning Tree Protocol, 2011).

The second type of BID is with the extended system ID. The structure is also shown in Figure 3, with Bridge Priority (4 bytes) + Extended System ID (12 bytes) +MAC Address (48 bytes). When VLANs became common for network infrastructure segmentation, the additional function was added into the STP to support the VLANs. Thus, extended system ID field includes the ID of VLAN with that the BPDU is related (Chapter 5 - Spanning Tree Protocol, 2011).

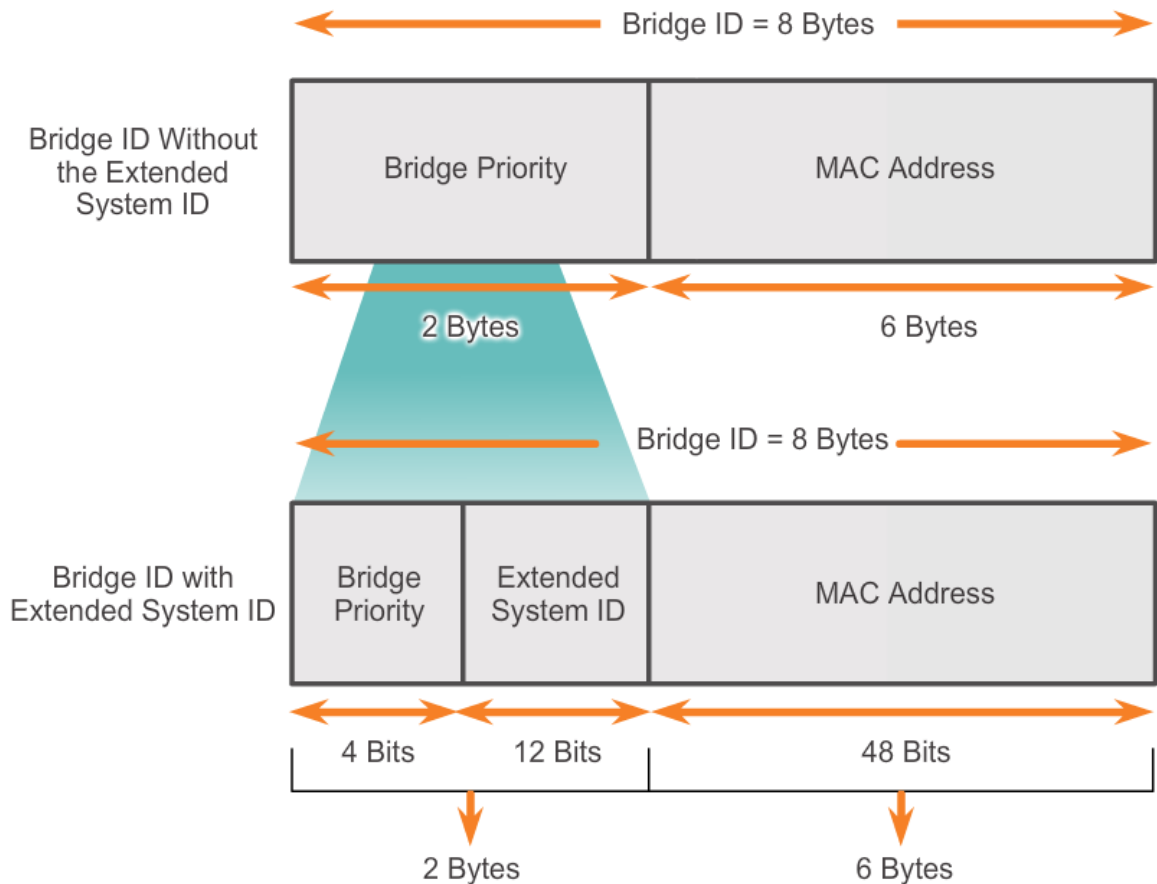


Figure 3 BID Fields (Chapter 5 - Spanning Tree Protocol, 2011)

2.4.2 Extend System ID

As shown in Figure 3, when use the extended system ID, the bridge priority value has been changed by the number of the bits, so the bridge priority value changes increase from 1 to 4096. Bridge priority values can only multiply by 4096 (Chapter 5 - Spanning Tree Protocol, 2011).

2.4.3 Bridge Priority

The priority of the bridge is a variable value and the range is from 1 to 65535. 1 is the highest bridge priority. You can change the value and determine which switch could become the root bridge. Election mechanism is based on the lowest priority value which means the highest priority. The switch with the lowest priority value could become the root bridge (Cisco, LAN Redundancy, 2013).

As shown in Figure 4, switch 1 is configured with the different priority compared to switch 2 and switch 3. Based on this situation, the switches elect the root bridge with the lowest bridge priority. The switch 1 has the lowest bridge priority (value is 24577). It is elected to be the root bridge.

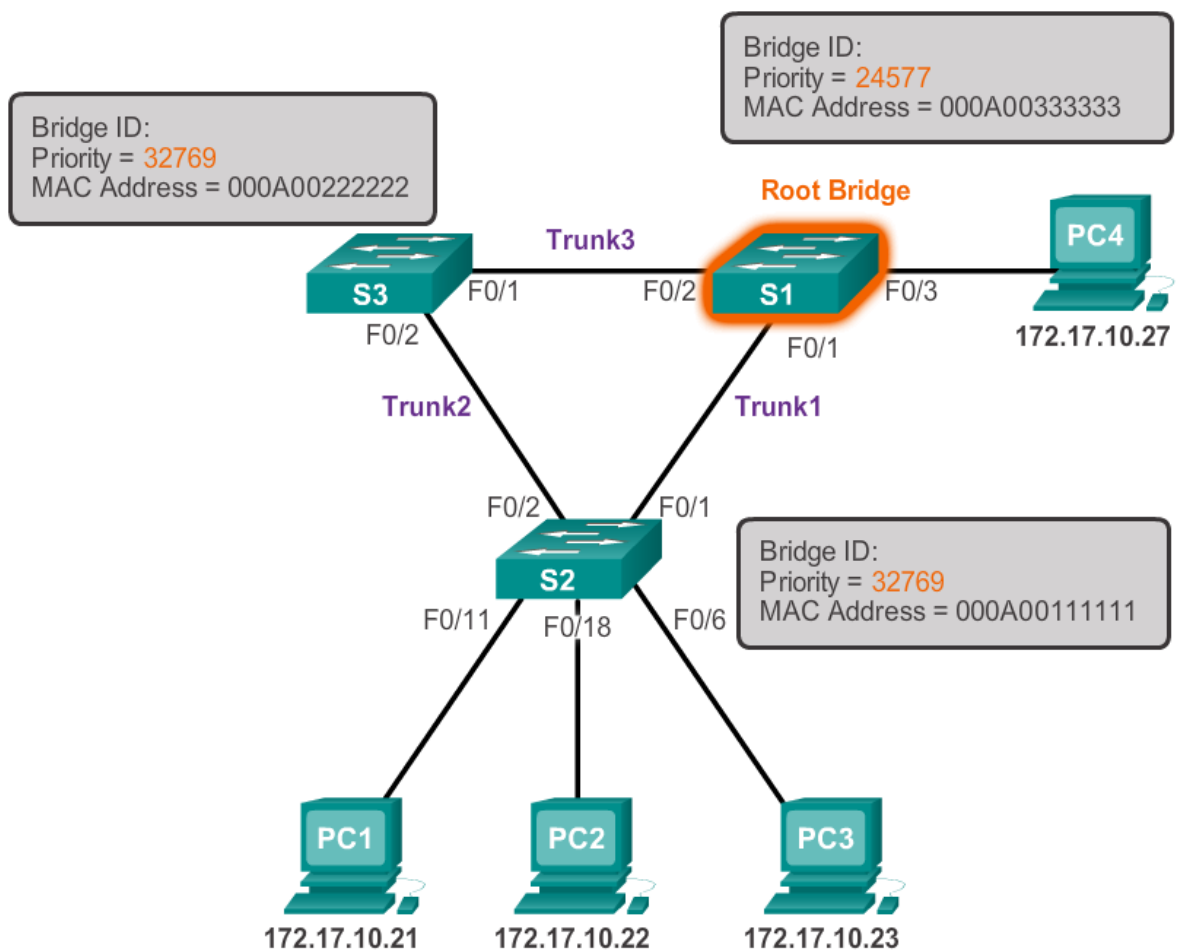


Figure 4 Priority-based Decision (Chapter 5 - Spanning Tree Protocol, 2011)

2.4.4 MAC Address

It is possible that all the switches are configured with the same bridge priority, and also they have the same extended system ID. As shown in Figure 5, based on this situation, the switches elect the root bridge by MAC address which has the lowest hexadecimal value. Switch 2 has the lowest MAC address value which is 000A00111111, so switch 2 is elected to be the root bridge (Cisco, LAN Redundancy, 2013).

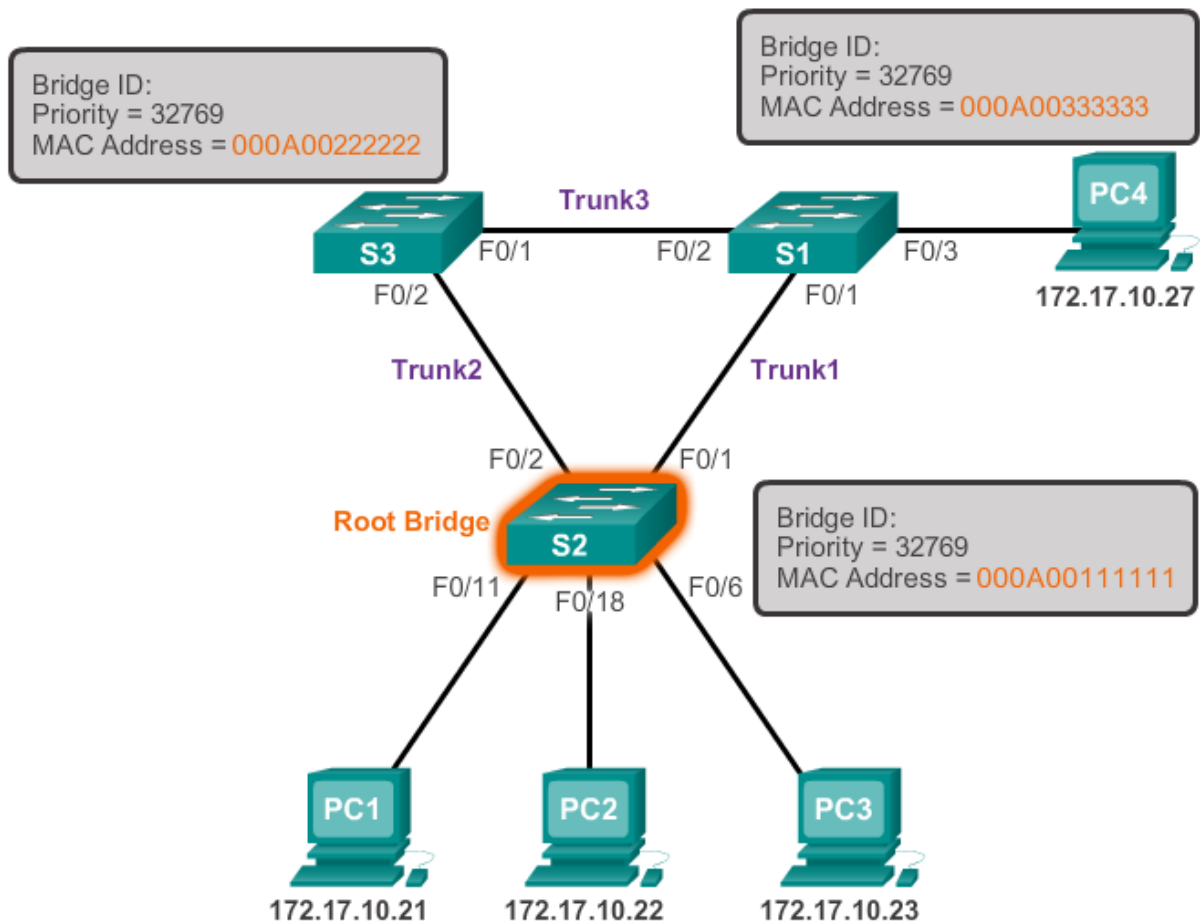


Figure 5 MAC Address-based Decision (Cisco, LAN Redundancy, 2013)

2.5 BPDU

2.5.1 BPDU Fields

BPDU frame contains 12 distinct fields as Figure 6 shows. BPDU is the frame that the STP switches use to exchange the messages. BPDU contains the path and priority information for

STP. STP uses this information to determine the root bridge and the path to the root bridge (Chapter 5 - Spanning Tree Protocol, 2011).

Field Number	Bytes	Field
1-4	2	Protocol ID
	1	Version
	1	Message type
	1	Flags
5-8	8	Root ID
	4	Cost of path
	8	Bridge ID
	2	Port ID
9-12	2	Message age
	2	Max age
	2	Hello time
	2	Forward delay

Figure 6 BPDU Fields (Chapter 5 - Spanning Tree Protocol, 2011)

Configuration BPDU (CBPDU) is used when each of the bridges send initial BPDUs, assuming itself as the root bridge. After convergence, BPDUs are only send out from the root bridge. Other bridges have trunk downward after the root port receives BPDU (Chapter 5 - Spanning Tree Protocol, 2011).

Topology Change Notification (TCN) BPDU is when the topology has changed. The other bridges send the BPDU from the root port to the root bridge.

2.5.2 BPDU Timers

BPDU timers contain Hello time, forward delay and Maximum age. The port stays keep time depends on the BPDU timers. Only the root bridge switch can adjust the timers by send information through the tree.

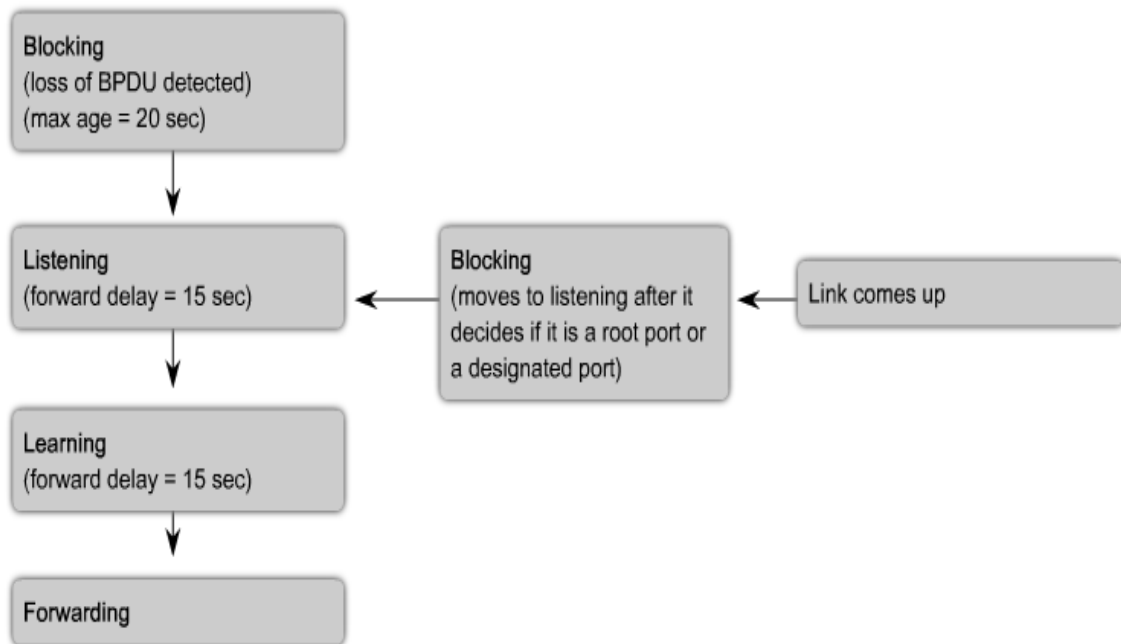


Figure 7 BPDU Timers (Chapter 5 - Spanning Tree Protocol, 2011)

As shown in Figure 7, when the STP is enabled, after the blocking state of each port in switch in the network, they will go through temporarily states at power up. These are listening and learning states. The port state change to the forwarding or blocking state. When the network topology changes, a port implement the listening and learning states for a specified period is forward delay interval.

HELLO TIME: It is the time between every BPDU frame send on a port. This is equal to 2 seconds by default. The range is 1 to 10 seconds.

FORWARD DELAY: Listening and learning time, this is equal to 15 seconds by default. The range can be 4 to 30 seconds.

MAXIMUM AGE: Configuration BPDU retention time, this is 20 seconds by default. The range can be 6 to 40 seconds (Chapter 5 - Spanning Tree Protocol, 2011).

3 ENHANCED STP

STP has bring new life to the transparent bridge. However, with the deepening of the application and development of network technology, its shortcoming in the application has also been

exposed (ZhangCun, 13) The STP defects mainly in the convergence speed. Figure 8 shows the STP variants.

Cisco Proprietary	PVST <ul style="list-style-type: none"> • Uses the Cisco proprietary ISL trunking protocol • Each VLAN has an instance of spanning tree • Ability to load balance traffic at layer-2 • Includes extensions BackboneFast, UplinkFast, and PortFast
	PVST+ <ul style="list-style-type: none"> • Supports ISL and IEEE 802.1Q trunking • Supports Cisco proprietary STP extensions • Adds BPDU guard and Root guard enhancements
	rapid-PVST+ <ul style="list-style-type: none"> • Based on IEEE802.1w standard • Has faster convergence than 802.1D
IEEE Standard	RSTP <ul style="list-style-type: none"> • Introduced in 1982 provides faster convergence than 802.1D • Implements generic versions of the Cisco proprietary STP extensions • IEEE has incorporated RSTP into 802.1D, identifying the specification as IEEE 802.1D-2004
	MSTP <ul style="list-style-type: none"> • Multiple VLANs can be mapped to the same spanning-tree instance • Inspired by the Cisco Multiple Instances Spanning Tree Protocol (MISTP) • IEEE 802.1Q-2003 now includes MSTP

Figure 8 STP Variants for Cisco Proprietary and IEEE Standard (House of Technology , 2013)

Now let us get familiar with each of the enhanced STP and the working principle.

3.1 Per VLAN Spanning Tree

Per VLAN Spanning Tree (PVST) is Cisco solution to solve the spanning tree in VLAN processing. PVST runs separate spanning tree instance for each VLAN. Under normal circumstances, PVST requirement in running on Cisco switches between trunk links of Interior Switching Link (ISL). PVST maintains a spanning tree instance for each configured VLAN. It uses ISL trunk and allows a VLAN trunk when the other VLANs blocking some forwarding. Since PVST treats each VLAN as a separate network, it has the ability (at layer 2) load balancing communication in the trunk and other VLANs in spanning tree loop does not cause another trunk VLANs (Baidubaike, 2013).

In IEEE 802.1D STP topology, as shown in Figure 9, A-D1 is active link. The A-D2 link is blocked. The entire switched network has been seen as a spanning tree instance, it is not allowed the loop in the instance. In Figure 10, each VLAN as a separate network in PVST topology, in A-D1 trunk link forward VLAN 501-1000 information and block VLAN 1-500. In A-D2 trunk link forward VLAN 1-500 information and block VLAN 501-1000. So the PVST topology keep load balancing communication in the trunk (Da, 2011).

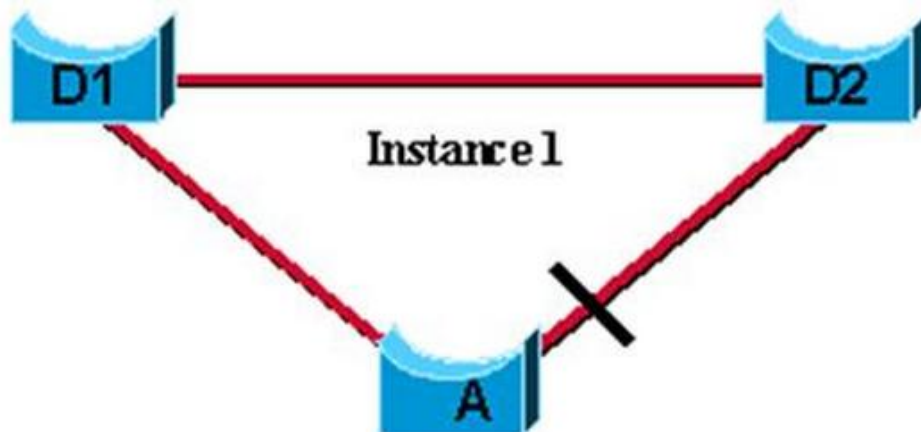


Figure 9 IEEE 802.1D STP Unbalanced Load Topology (Da, 2011)

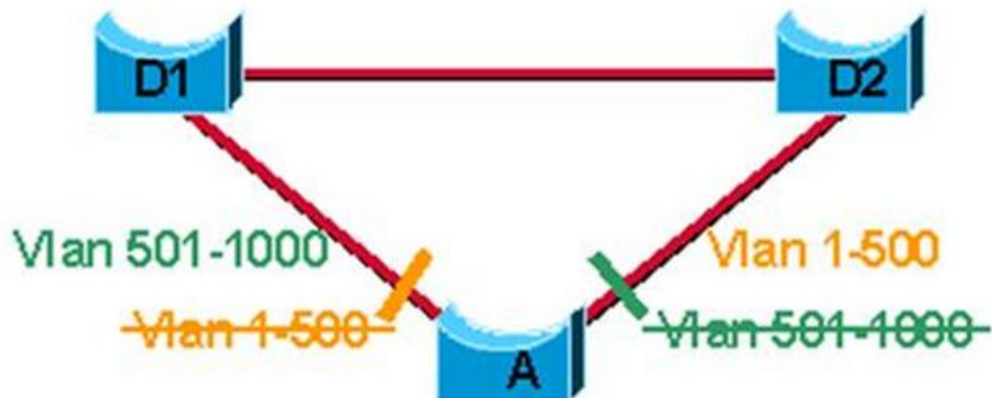


Figure 10 Cisco PVST Load Balancing Topology (Da, 2011)

3.2 Per VLAN Spanning Tree Plus

Per VLAN Spanning Tree Plus (PVST+) is another program to solve the processing spanning tree problem in VLAN of Cisco. The original IEEE 802.1D standard defines the Common Spanning Tree (CST) assumes one spanning tree instance for the entire bridge network, one link if active and must block for all VLANs, regardless of the number of VLANs (Cisco,

2012). PVST+ allows the CST information send to PVST, in order to operate with other vendors. PVST+ can use 802.1Q encapsulation and it in Catalyst 802.1Q trunk is automatically activated. PVST+ also can achieve the load balancing in layer 2. PVST+ is not supported on non-Cisco devices. PVST+ divided into three types of regions: PVST region, PVST+ region and single spanning tree region (Cisco, LAN Redundancy, 2013).

As shown in Figure 11, in the PVST+ environment, the half of VLANs also can work to forward to each uplink trunk which spanning tree parameters can be tuned. On S2, the F0/2 port is the forwarding port for VLAN 10 and blocking port for VLAN 20, the F0/3 port is the forwarding port for VLAN 20 and blocking port for VLAN 10. This can be done by configure one switch to be elected the half of the VLANs of the root bridge in the network. The second switch to be elected the other half of the VLANs of the root bridge. So the result is S1 is the root bridge for VLAN 10, S3 is the root bridge for VLAN 20.

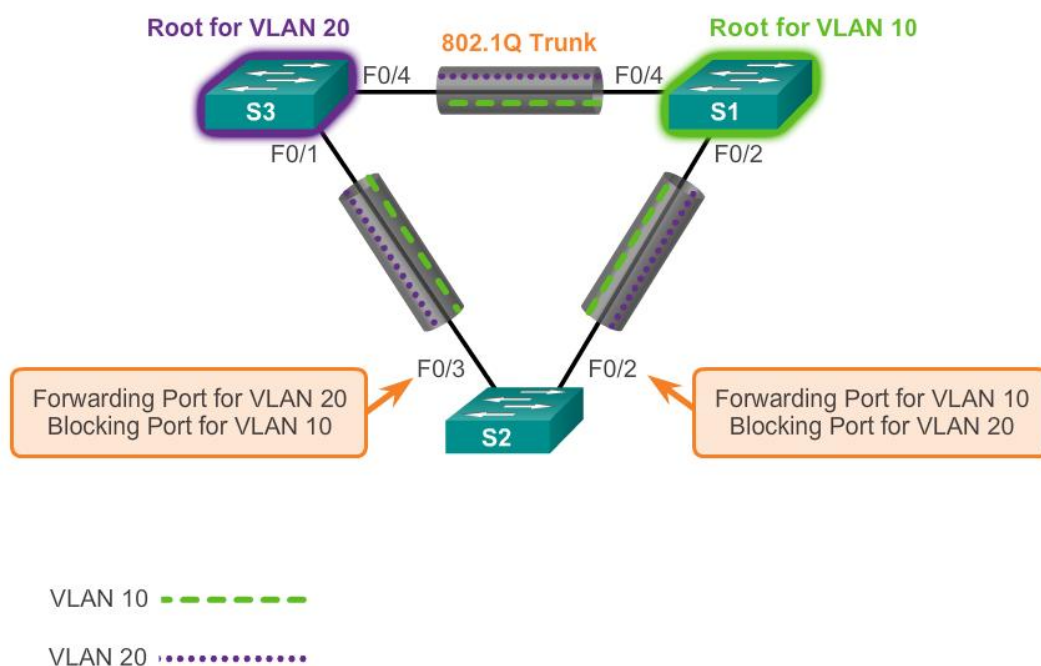


Figure 11 The PVST+ Topology (Cisco, LAN Redundancy, 2013)

PVST+ includes the PortFast enhancement that called BPDU guard and root guard. In PVST+ environment, the extend system ID can ensure that each VLAN has a unique BID in each switch. For instance, the VLAN 20 default BID is 32770, so that means the priority is 32768 and plus the extended system ID of 2. If VLAN has been not configured by priority, each

VLAN have the same default priority and elected the root is based on the MAC address for switch (Cisco, LAN Redundancy, 2013).

The weakness of PVST+ is when there are many VLANs in topology, the computation and resource footprint will increase dramatically when maintaining multiple spanning trees. Especially the port state changes when a VLAN has amount of trunk links, all of the spanning tree states must be recalculated, and the CPU will be overwhelmed. Therefore, Cisco switches to limit the use number of VLANs, at the same time in a port is not recommended to created many VLAN trunks (Cisco, LAN Redundancy, 2013).

3.3 Rapid Spanning Tree Protocol

In 2001, the IEEE introduced Rapid Spanning Tree Protocol (RSTP) as 802.1W. RSTP is an evolution from the 802.1D standard. The RSTP use the primarily terminology of the original IEEE 802.1D STP. Most parameters are same as STP, so user can easily configure the RSTP with the knowledge of STP (Cisco, Understanding Rapid Spanning Tree Protocol (802.1w), 2006).

As shown in Figure 12, the topology of the RSTP. S1 is the root bridge and the F0/4 and F0/2 are designated ports in forwarding state. On S2, F0/3 is an alternate port in discarding state which is the new port type. In Figure 13, it is the comparison of STP and RSTP Port State. There is no blocking port exist in RSTP port state. In 802.1W, the disable port, listening port, and blocking port from 802.1D are unified into discarding port state.

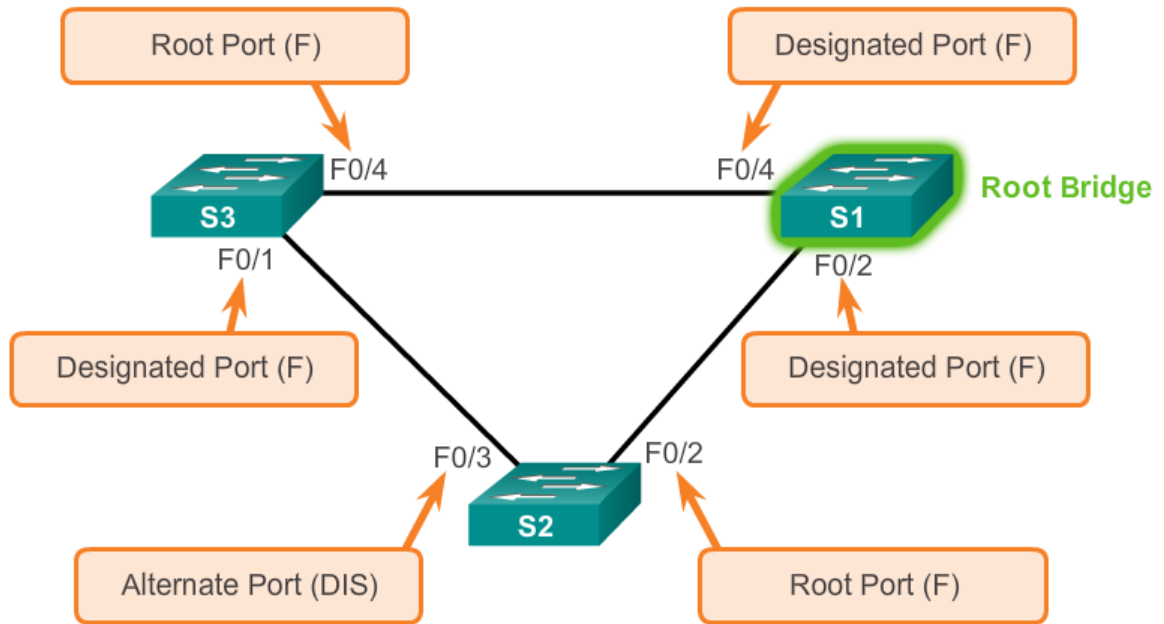


Figure 12 RSTP Topology (Cisco, LAN Redundancy, 2013)

STP (802.1D) Port State	RSTP (802.1w) Port State	Is Port Included in Active Topology?	Is Port Learning MAC Addresses?
Disabled	Discarding	No	No
Blocking	Discarding	No	No
Listening	Discarding	Yes	No
Learning	Learning	Yes	Yes
Forwarding	Forwarding	Yes	Yes

Figure 13 STP and RSTP Port State Comparison(Cisco, 2006)

As shown in Figure 14, RSTP BPDU is type 2, version 2. The original 802.1D BPDU is type 0, version 0. In 802.1D BPDU only defines the Topology Change and Topology Change Acknowledgment two flags bits. The RSTP uses all six bits of the flag byte to achieve encode the role and state of the port that originates the BPDU and manage the proposal and agreement mechanism. For the legacy will drop this RSTP BPDU, so this property can easily to detect the legacy bridge to connected of RSTP bridge (Cisco, Understanding Rapid Spanning Tree Protocol (802.1w), 2006).

RSTP Version 2 BPDU	
Field	Byte Length
Protocol ID=0x0000	2
Protocol Version ID=0x02	1
BPDU Type=0X02	1
Flags	1
Root ID	8
Root Path Cost	4
Bridge ID	8
Port ID	2
Message Age	2
Max Age	2
Hello Time	2
Forward Delay	2

Flag Field	
Field Bit	Bit
Topology Change	0
Proposal	1
Port Role	2-3
Unknown Port	00
Alternate or Backup Port	01
Root Port	10
Designated Port	11
Learning	4
Forwarding	5
Agreement	6
Topology Change Acknowledgment	7

Figure 14 RSTP BPDU (Cisco, LAN Redundancy, 2013)

There are four edge ports as shown in the Figure 15 and the rest of ports on switches are non-edge ports. The RSTP edge port also is a switch port that is will never connected to another switch. It directly connected to the end stations, so that it cannot occurs the bridge loops in the networks. The edge port immediately transitions to the forwarding state, the advantage is skip the listening and learning port states, the functions is similarly to the PortFast in PVST+ (Cisco, LAN Redundancy, 2013).

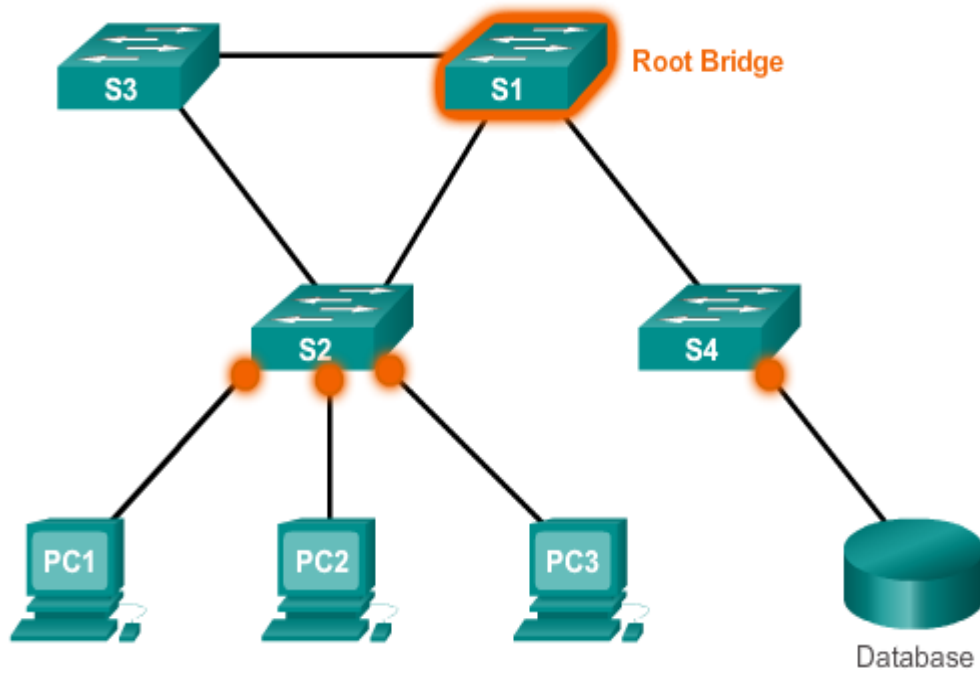


Figure 15 Edge Ports (Cisco, LAN Redundancy, 2013)

RSTP is still the single spanning tree structure, it cannot do load balancing, and cannot carrying any traffic after the link has been blocked that caused the wasted of bandwidth. For the large network, the convergence speed is still not enough to cope with it.

3.4 Multiple Spanning Tree Protocol

Mentioned STP, RSTP protocols and Cisco proprietary protocol PVST, PVST + belong to a single spanning tree (SST) protocol, which is the support for multi-VLANs devices can only run a single spanning tree. Multiple Spanning Tree Protocol (MSTP) is new agreement that defines in IEEE 802.1S standard which proposed combination STP and VLAN, it inherits the advantages of rapid migration of RSTP port, and also solve the problem that different VLANs must run on the same spanning tree of RSTP, to avoid the VLAN load balancing cannot be achieved and waste of link bandwidth (ZhangCun, 13).

3.4.1 Instance and Region

To put it simply, STP and RSTP are based on the port, PVST+ is based on VLAN, and MSTP is based on the instance. Compared with STP, RSTP and PVST+, MSTP introduced “instance” and “region” concept (ZhangCun, 13).

The instance is a collection of multiple VLANs, this through multiple VLANs bundle into an instance method can save the communication overhead and resource utilization. In MSTP, each instance topology calculating is independent, it can be realized the load balancing in these instances. The same VLANs topologies can be mapped to a particular instance, these VLANs port forwarding state will depend on corresponding instance forwarding state of MSTP (MSTP--H3C, 2009).

The region consist by Configuration Name, Revision Level, Configuration Identifier Format Selector, mapping of VLAN ID (VID) to spanning trees. In which the configuration name, revision level, configuration identifier format selector in the BPDU packets have an associated field.

Mapping of VIDs to spanning trees in BPDU performance as Configuration Digest. According to Mapping of VIDs to spanning trees, the configuration digest calculated a 16-byte signature. These four parameters are same and with the interconnected switches to be treats as in the same region (MSTP--H3C, 2009).

MSTP instance 0 has a special role, and called the Common Internal Spanning Tree (CIST), the other instance 2 called Multiple Spanning Tree Instance (MSTI). CIST consist of calculate the single spanning tree by STP and RSTP and region calculation of MSTP, it is to ensure that all bridged LANs are simple and full connectivity. CST is STP and RSTP, it also is a single spanning tree which calculates by MSTP for connecting the Multiple Spanning Tree (MST) region. Internal Spanning Tree (IST) in a given MST region provided by the CIST connectivity. In Figure 16 MSTP basic concepts shown, assume each MST region as a "switch", CST is a spanning tree which through these "switches" generated by STP and RSTP or MSTP. IST is a fragment of CIST in an MST region, it is a special Multiple Spanning Tree Instance (MSTI) (MSTP--H3C, 2009).

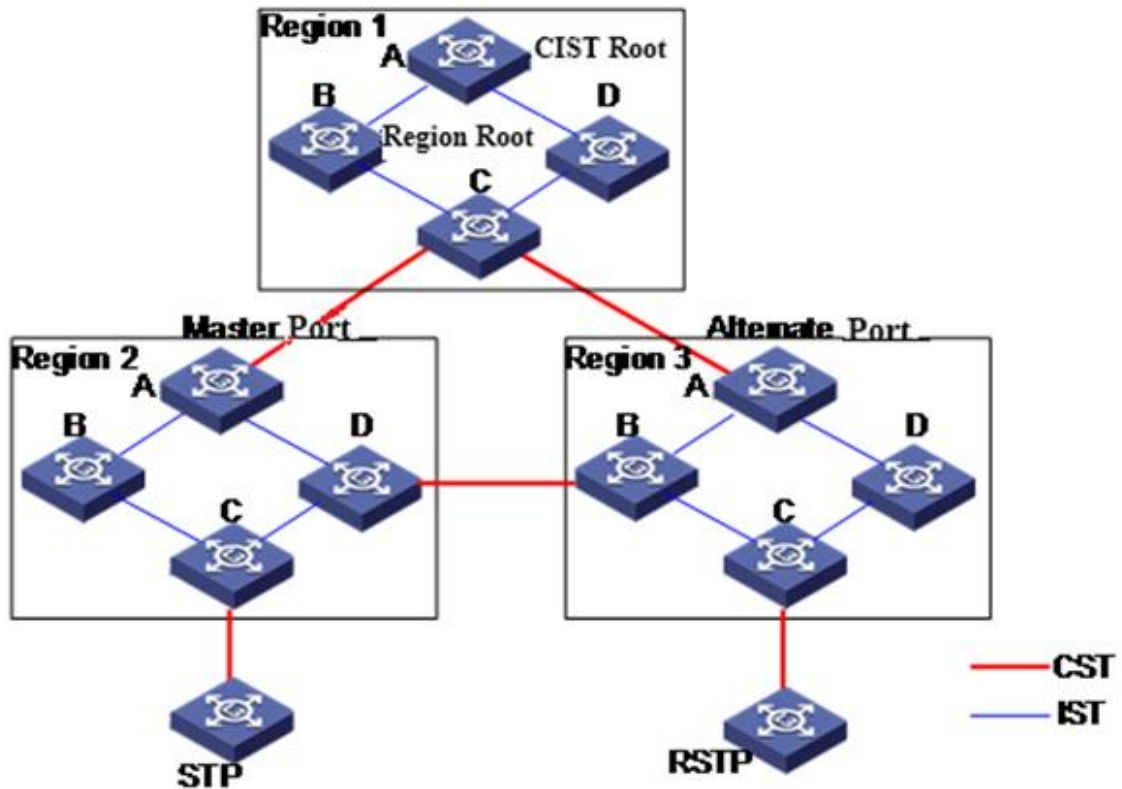


Figure 16 MSTP Basic Concepts Topology (MSTP--H3C, 2009)

3.4.2 CIST Root and Region Root

Compared with STP and RSTP, MSTP introduced the concept of CIST root and region root. The CIST root is a global concept, all interconnected for running STP, RSTP and MSTP switches only have one CIST root. While the region root is a local concept is relative to a certain instance of region. As shown in Figure 16, all connected devices only have one CIST root which is switch A in region 1, and the number of region root for each region associated with the number of instances.

3.4.3 Path Cost and Ports

MSTP introduced the concept of external and internal path cost. External path cost is relative to the CIST, the same region external path costs are the same. Internal path cost is relative to an instance in a region, the same port for different instances correspond to different internal path cost (MSTP--H3C, 2009).

MSTP introduced the concept of Boundary port and Master port. Boundary port is connected to a different of the MST regions, MST region to STP operation area, and MST region to port of RSTP area which located at the edge of the MST region. In a region that does not contain the CIST root, master port is all the boundary ports which reach the CIST root port with minimal cost, and it is connected the MST region to the CIST root port. Alternate port is the backup port for master port, if master port is blocked, alternate port will become the new master port. In Figure 16, the CIST root in region 1. In region 2 and region 3, the port on device C is the boundary port. In region 2, the port on device A which is connected to region 1 is the master port. In region 3, the port on device A which is connected to region 1 is the alternate port. The device A contains master port is called the master device in region 2(MSTP--H3C, 2009).

4 MEASUREMENT DESIGN

After discussing the theory of the STP, in this chapter, we will discuss the testing environment which I use in my performance measurement. I will test convergence performance of the default STP which is PVST+ implement on Cisco Catalyst 2960 series. After that I will test the RSTP and MSTP.

Before the configuration of the switches, I need to check does Cisco Catalyst 2960 series support all three protocols which I want to implement and test. This is the quite important work need to be done. The result is Cisco Catalyst 2960 series support all three protocols.

The topology diagram is shown in Figure 17. This is basic the topology for my PVST+, RSTP and MSTP tests.

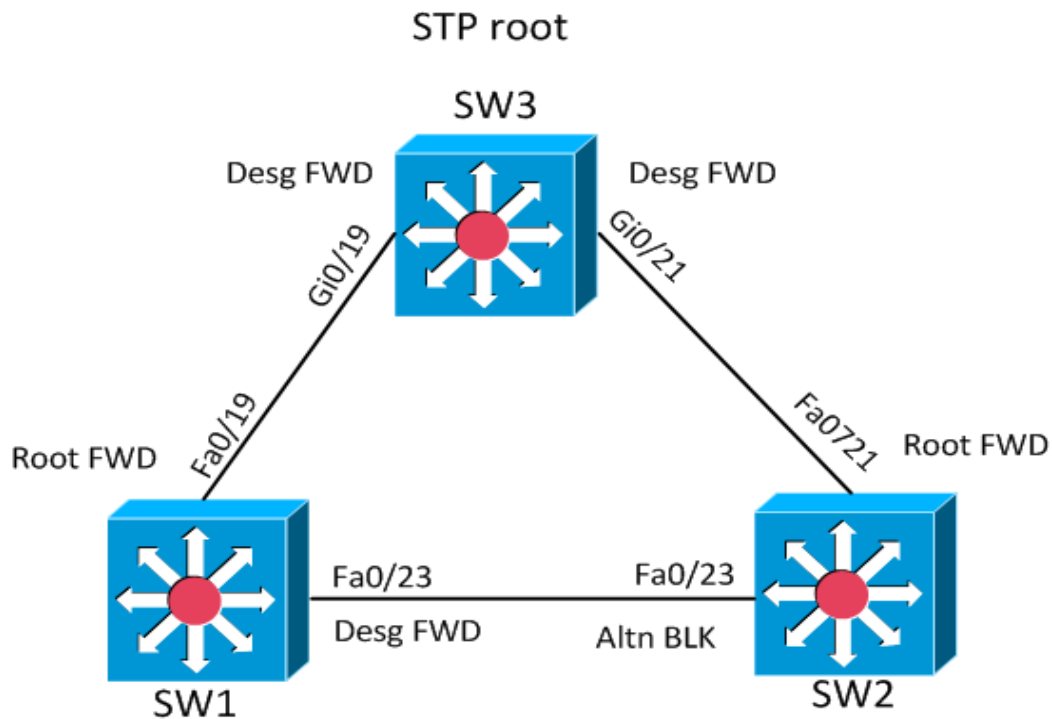


Figure 17 Basic Topology Diagram

The purpose of the measurement is to measure the convergence performance, so I choose to send the continuous ping from one PC to another PC. Then I will shutdown one interface of the switch, and start to calculate the time by my mobile phone. The ping test will show "Request timed out.", now the topology will choose a new link to send the information. When the new line is set up, the ping test will show like "Reply from 172.31.10.22: bytes =32 time<1ms TTL =128", and now, I will stop the clock and check the time. This time is the convergence time for the spanning tree, because I measure the result by hand, so I will measure five times to reduce errors compare with just test one time.

5 DESIGN IMPLEMENTATION AND MEASUREMENT

In this chapter, I will show the process of the implementation of three protocols. Explain the method and process of test. Give the summary for the each test result.

5.1 Performance Measurement of PVST+

The topology diagram is shown in Figure 18, this is basic the topology for the PVST+. I will add more PCs on each VLAN, not like shown in Figure that just have one PC in each VLAN.

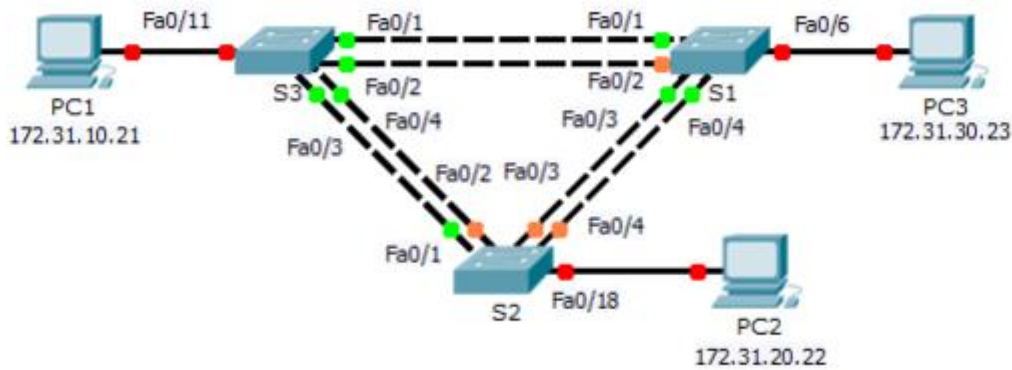


Figure 18 PVST+ Topology

5.1.1 Implementing PVST+

I create the network as shown in Figure 18, and implement the PVST+ on this topology. The physical work is to connect all the devices, so I need to take care of all the wires and all devices.

The implement action steps of PVST+ are shown below.

Step 1: Create VLANs on all switches. I created VLANs 10, 20, 30, 40, 50, 60, 70, 80 and 99. The VLAN 99 is the native VLAN. Configure VLAN 99 with the IP address and subnet mask.

Step 2: Enable the user ports on S1, S2 and S3 in access mode and enabled them. Assign each VLAN on right port in all switches. Configure ports Fa0/1 to Fa0/4 as trunk ports and assign them to the native VLAN 99.

Step 3: Configure the IP addresses, subnet masks and default gateways on all PCs.

Step 4: Configure the STP mode which is PVST on all switches.

Step 5: Configure spanning tree PVST+ load balancing on all switches. Configure S1 as the primary root for VLAN 1, 10, 30, 50 and 70. Configure S3 as the primary root for VLAN 20, 40, 60, 80 and 99. S2 is set as the secondary root for all VLANs.

Step 6: Configure PortFast and BPDU guard on all switches which the port connected to the PC.

5.1.2 Test and Result

Now I test the devices in VLAN 10, the S1 is the root for VLAN 10. I added a new PC4 on S2 which IP address is 172.31.10.22 and it belongs to VLAN 10. This PC was not shown in Figure 18, so I explain it here.

I use the CMD from PC1 to send continuous ping to PC4. The link should go through from S3 to S1 to S2 and arrived at PC4. The ping test is shown in Figure 19, and the link is up based on first part of result which is shown as “Replay from 172.31.10.22: bytes =32 time<1ms TTL =128”.

“”

After that, I disconnect the links between S1 to S2, the original path stops the work in the network of VLAN 10. At the same time, I start to measure the time by my clock. The changes are shown in CMD. We can find the second part which indicates “Request timed out.” During this period, the link states between PC1 to PC4 is disconnected, the devices in the topology will find the new path transmit the message. We also can find the message from the Command-Line Interface (CLI) as shown in Figure 20. It is shows on S1, the interface fa0/1 and fa0/2 states changed to down.

On third part as shown in Figure 19, the new link now is set up. When the first line of the third part come out, I stop the clock and check the time. This time as shown in Figure 21 which 31.4 seconds is the convergence time for the network.

On fourth and fifth parts as shown in Figure 20, I reconnected the links between S1 to S2 and start to calculate the time, the switches communicate with others and decide S1 to be the root bridge again. The link between PC1 to PC4 change down and after election of the root bridge finished, the link state changes to up. When the first line of the fifth part comes out, I stop the clock to write down the time. This time also is the convergence time for the network topology.

```

Microsoft Windows [Version 6.2.9200]
(c) 2012 Microsoft Corporation. All rights reserved.
C:\Users\Student>ping 172.31.10.22 -t

Pinging 172.31.10.22 with 32 bytes of data:
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128

```

Figure 19 CMD Result

```

Tera Term Web 3.1 - COM1 VT
File Edit Setup Web Control Window Help

*Mar 1 01:24:07.651: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan99, changed state
to down
*Mar 1 01:24:08.582: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, chan
ged state to down
*Mar 1 01:24:08.624: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, chan
ged state to down
*Mar 1 01:24:09.597: %LINK-3-UPDOWN: Interface FastEthernet0/2, changed state to down
*Mar 1 01:24:09.648: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to down
*Mar 1 01:24:37.657: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan99, changed state
to up
*Mar 1 01:25:10.759: %LINK-3-UPDOWN: Interface FastEthernet0/1, changed state to up
*Mar 1 01:25:12.755: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan99, changed state
to down
*Mar 1 01:25:12.789: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, chan
ged state to up
*Mar 1 01:25:41.813: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan99, changed state
to up

```

Figure 20 CLI Result



Figure 21 Time Result for One Test

The test result as shown in Table 1, we can find that the convergence time of PVST+ is around 30 seconds but no less than 30 seconds. It is the lowest protocol compare with the RSTP and MSTP of convergence time.

Table 1 PVST+ Convergence Time Result

Times of Test	Convergence Time(s)	Reconnected Convergence Time(s)
1	31.4	33.4
2	30.8	34.1
3	30.6	32.9
4	31.6	32.7
5	30.9	34.1
Average Time	31.06	33.44

5.2 Performance Measurement of RSTP

The topology diagram for RSTP is shown in Figure 22, there are three switches I will use and two PCs in the same VLAN. With this topology, we can test the performance of convergence time of the network which uses the RSTP.

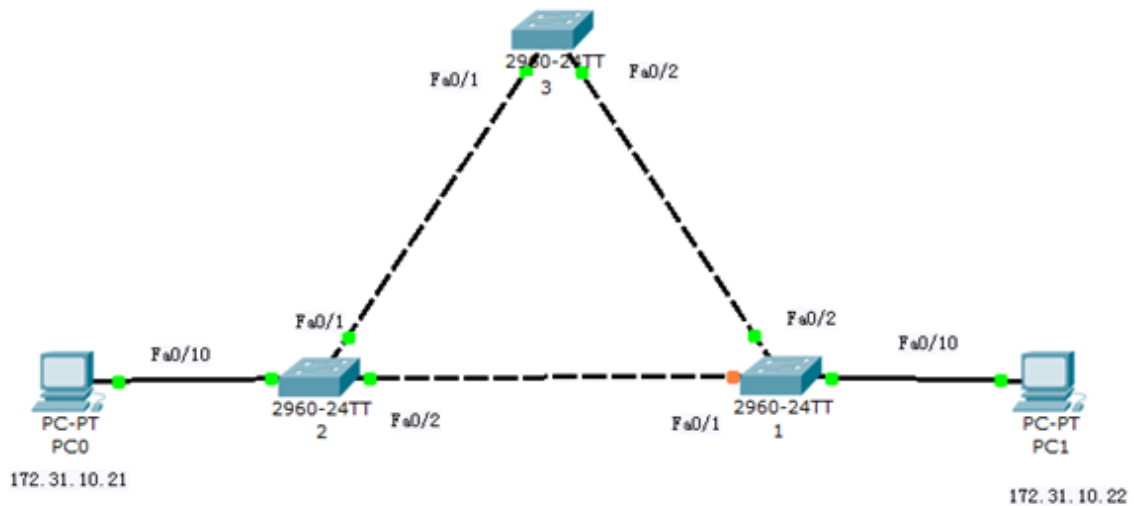


Figure 22 RSTP Topology

5.2.1 Implementing RSTP

After the first measurement of PVST+, the implement action of RSTP is easy. I use the short and clean commands to let RSTP work. The implement action steps of the RSTP are shown below.

Step 1: Connected all the devices following the topology in Figure 22.

Step 2: Enable the user port Fa0/10 on each switch in access mode and PortFast. I use the VLAN 1, so by default, all ports can access VLAN 1.

Step 3: Configure the IP addresses, subnet masks and default gateways on all PCs.

Step 4: Configure the STP mode which is RSTP on all switches.

Step 5: Configure the S3 as the primary root for VLAN 1, S2 as the secondary root for VLAN 1.

5.2.2 Test and Result

I ping from PC0 to PC1. The S3 is the root switch for VLAN 1, so the process of the ping is from S2 to S3 to S1 and arrive at PC1. From Figure 23, we can find the link is up.

After that, I disconnect the Fa0/2 and start to record the time by clock. However, there is nothing change from the CMD ping test as shown in Figure 23. And I check the interface state, I find the change already happened as shown in Figure 24. The Fa0/1 port role change to root port and the state change to forwarding. This means the network choose a new link to transmit the information that is from S2 to S1 and reach the PC1.

```

Command Prompt - ping 172.31.10.22 -t
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128
Reply from 172.31.10.22: bytes=32 time<1ms TTL=128

```

Figure 23 Ping Test

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/1	Altn	BLK	19	128.1	P2p
Fa0/2	Root	FWD	19	128.2	P2p
Fa0/10	Desg	FWD	19	128.10	P2p

↓

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/1	Root	FWD	19	128.1	P2p
Fa0/10	Desg	FWD	19	128.10	P2p

Figure 24 S1 Port

Because I need to measure the convergence time, the basic on measurement method for PVST+ is not working now. So I use the “debug spanning-tree event”. When the topology is changed, the change record is shown on CLI. But the drawback is by this method I just can test when reconnect the link between S1 and S3. I cannot test the convergence time when disconnect the link between S1 and S3. As shown in Figure 25, these commands are recorded in the communication process. When I reconnect the link, we can find based on the communication results, that the interface Fa0/2 changed from designated port to root port, and

Fa0/1 change to alternate port. The convergence time is from the moment I reconnect the interface Fa0/2 physically to the last command displayed in CLI which shows line protocol on Fa0/2 changed state to up to stop the clock.

```

*Mar 1 00:25:17.062: RSTP(1): initializing port Fa0/2
*Mar 1 00:25:17.062: RSTP(1): Fa0/2 is now designated
*Mar 1 00:25:17.071: RSTP(1): transmitting a proposal on Fa0/2
*Mar 1 00:25:17.104: RSTP(1): updt roles, received superior bpdu on Fa0/2
*Mar 1 00:25:17.104: RSTP(1): Fa0/2 is now root port
*Mar 1 00:25:17.104: RSTP(1): Fa0/1 blocked by re-root
*Mar 1 00:25:17.104: RSTP(1): synced Fa0/2
*Mar 1 00:25:17.104: RSTP(1): Fa0/1 is now alternate
*Mar 1 00:25:17.104: STP[1]: Generating TC trap for port FastEthernet0/2
*Mar 1 00:25:17.113: RSTP(1): transmitting an agreement on Fa0/2 as a response
to a proposal
*Mar 1 00:25:19.059: %LINK-3-UPDOWN: Interface FastEthernet0/2, changed state
to up
*Mar 1 00:25:20.066: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEther
net0/2, changed state to up

```

Figure 25 S1 Debug Events

Table 2 RSTP Convergence Time Result

Times of Test	1	2	3	4	5	Average Time
Convergence Time(s)	4.3	3.4	4.1	4.3	3.9	4.0

From Table 2, we can find the convergence time of RSTP is around the 4 seconds in my lab environment. It is the fastest protocol compare with PVST+ and MSTP of convergence time.

5.3 Performance Measurement of MSTP

The topology diagram for MSTP is shown in Figure 22. There are three switches and four PCs. Two PCs are in the same VLAN. With this topology, we can test the performance of convergence time of the network which implement the MSTP.

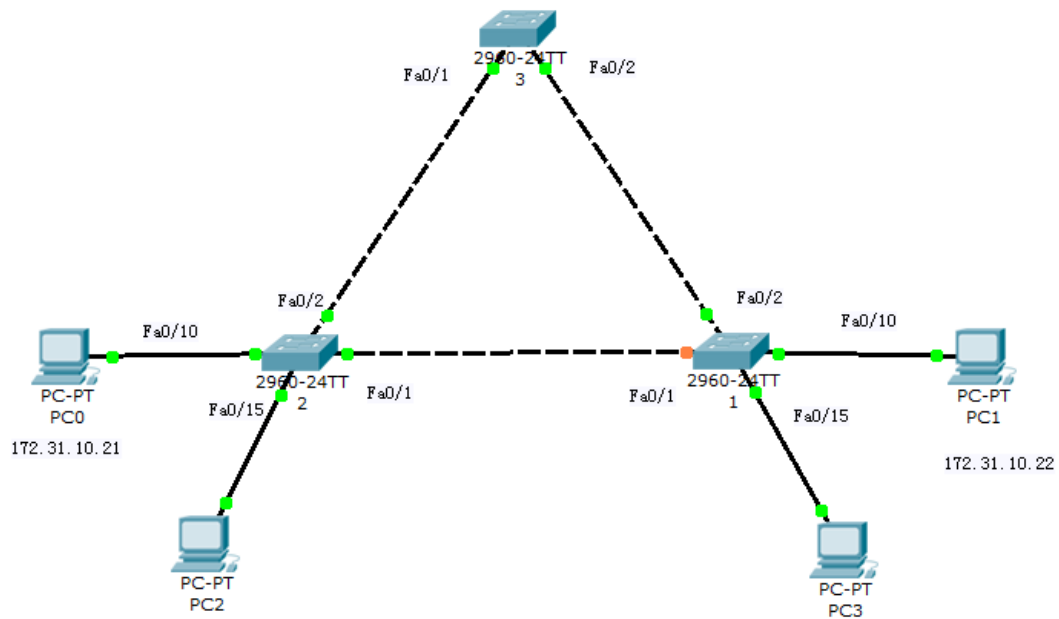


Figure 26 MSTP Topology

5.3.1 Implementing MSTP

I connect the devices following the topology as shown in Figure 26. The implement action steps of the MSTP are shown below.

Step 1: Create VLAN 10, 20, 40 on all switches. Enable the user port on S1 and S2 in access mode. Assign PC0 and PC1 to VLAN 10, PC2 to VLAN 40, PC3 to VLAN 20.

Step 2: Configure the IP addresses, subnet masks and default gateways on all PCs.

Step 3: Configure S3 as root for VLAN 10, S2 as the secondary root for VLAN 10.

Step 4: Configure Fa0/1 and Fa0/2 on all switches as trunk link. Enable all user ports with PortFast.

Step 5: Configure the STP mode as “mst”. Create an instance, set the name and revision number on all switches. Let PC0 and PC2 in instance1, PC1 and PC3 in instance2.

5.3.2 Test and Result

Now I test the devices in VLAN 10 in two instances. The S3 is the root for VLAN 10. Continuous ping from PC0 to s PC1 id done again. I test the link between them and the link is up. I shutdown the interface Fa0/2 in S1, so the topology will change to set up a new link. But the result from the CMD shows like nothing has changed. The situation is similar to the test of RSTP. As shown in Figure 27, I reconnect the Fa0/2 on S1, the Fa0/1 changes back to the alternate port role and state change to blocking. The difference between RSTP and MSTP is that MSTP has a higher cost.

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/1	Root	FWD	200000	128.1	P2p
Fa0/10	Desg	FWD	200000	128.10	P2p Edge

↓

Interface	Role	Sts	Cost	Prio.Nbr	Type
Fa0/1	Altn	BLK	200000	128.1	P2p
Fa0/2	Root	FWD	200000	128.2	P2p
Fa0/10	Desg	FWD	200000	128.10	P2p Edge

Figure 27 S1 Port

So I have to use the method I used with RSTP which use debug command to track the process. I start to record the time when I reconnect the Fa0/2 on S1 and pay attention on debug events, until the line protocol on Fa0/2 change state to up as shown in Figure 28. Then I stop the clock. This time is the convergence time for MSTP.

```

Tera Term Web 3.1 - COM1 VT
File Edit Setup Web Control Window Help
*Mar 1 02:54:21.499: %LINK-3-UPDOWN: Interface FastEthernet0/2, changed stat
e to up
*Mar 1 02:54:22.531: STP[1]: Generating TC trap for port FastEthernet0/2
*Mar 1 02:54:22.540: STP[0]: Generating TC trap for port FastEthernet0/2
*Mar 1 02:54:22.540: STP[2]: Generating TC trap for port FastEthernet0/2
*Mar 1 02:54:23.521: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEth
ernet0/2, changed state to up
*Mar 1 02:54:47.563: STP[1]: Generating TC trap for port FastEthernet0/1
*Mar 1 02:54:48.553: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEth
ernet0/2, changed state to down
*Mar 1 02:54:49.559: %LINK-3-UPDOWN: Interface FastEthernet0/2, changed stat
e to down
*Mar 1 02:55:01.261: %LINK-3-UPDOWN: Interface FastEthernet0/2, changed stat
e to up
*Mar 1 02:55:02.310: STP[1]: Generating TC trap for port FastEthernet0/2
*Mar 1 02:55:02.327: STP[0]: Generating TC trap for port FastEthernet0/2
*Mar 1 02:55:02.327: STP[2]: Generating TC trap for port FastEthernet0/2
*Mar 1 02:55:03.283: %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEth
ernet0/2, changed state to up

```

Figure 28 S1 Debug Events

Table 3 MSTP Convergence Time Result

Times of Test	1	2	3	4	5	Average Time
Convergence Time(s)	6.2	5.7	5.8	6.1	6.1	5.98

From Table 3, we can find the convergence time of MSTP is around the 6 seconds in my lab environment. It is slower compare with the RSTP.

6 CONCLUSION

Through the theoretical part, I deeply learn the knowledge of the STP and enhanced STPs. I have mastered more details about how they operate and what are the difference between them. Enhanced STP protocols such as the RSTP and MSTP were new knowledge for me. After my new experiences I can say that they are more useful and better than the traditional STP.

After theoretical part, we know the basic concepts of the STP and enhanced STPs, and then I test the performance of original STP which is PVST+ as default on Cisco Catalyst 2960 series. After that I test the enhanced STPs of RSTP and MSTP. The testing process is divided into two part, one part is configuration of the protocols in network environment and another part is test and results. The implementation part performed relatively smooth, just some small

problems were met during the RSTP. I spent more time on research how to configure these protocols, check so many versions of configuration and choose a better and clearer one to implement in my test network. I used the Packet Tracer to test the configuration at home, and implemented it then in the real machines, because the test should in the real machines. The virtual environment is always idealized. I can met more unexpected problems and get more experience to solve them with the real switches and computers. The test and result part is more interesting, I need to know how to test, and what to test. There was very little information on internet about how to test the convergence time. Also when testing PVST+, I use one kind of method, but this method did not work in RSTP and MSTP environment. So I found another method to deal with the RSTP and MSTP. The test results are similar to theoretical part. The PVST+ is quite slow which the average time is more than 30 seconds compare with the RSTP and MSTP on the convergence time. The RSTP is the fastest one in these three protocols which the average time is 4 seconds, but the most suitable solution is use the MSTP in the real work situation. It has the rapid convergent performance which the average time is 5.98 seconds and compatibility.

The development of any technology will not stop because of the emergence of an "ideal" technology. STP development process itself illustrates this point. With the application of in-depth, there will be new demands of STP for users and service providers. Perhaps in the near future, the new protocols will appear in our networks.

BIBLIOGRAPHY

(2013, 8 13). Retrieved from House of Technology :

http://mars.tekkom.dk/mediawiki/index.php/CCNA_Explorer_3_STP

(2013, 12 27). Retrieved from Baidubaiké:

http://baike.baidu.com/link?url=2LJa_pGW_rCSMD8SG9eL3HmT-uIxQ56qAJ4EVEPIFBnXk7PURcB-Ha6KMivB-5cH

Academy, U. C. (2014). *Varieties of Spanning Tree Protocols-pvst+*. Retrieved from Academy, UPC Cisco Networking:

<http://cna.upc.edu.cn/rs/03/course/module2/2.2.2.2/2.2.2.2.html>

Chapter 5 - Spanning Tree Protocol. (2011, 6 15). Retrieved from <https://ls->

[a.org/doku.php?id=school:2a_chapter_5_notes](https://ls-a.org/doku.php?id=school:2a_chapter_5_notes)

- Cisco. (2006, OCT 24). *Understanding Rapid Spanning Tree Protocol (802.1w)*. Retrieved from Cisco: <http://www.cisco.com/c/en/us/support/docs/lan-switching/spanning-tree-protocol/24062-146.html>
- Cisco. (2012, january). Retrieved from Common Spanning Tree: http://www.cisco.com/en/US/tech/tk389/tk621/tk868/tsd_technology_support_sub-protocol_home.html
- Cisco. (2013). *LAN Redundancy*. Retrieved from Cisco Networking Academy : <https://static-course-assets.s3.amazonaws.com/ScaN50ENU/module2/index.html#2.0.1.1>
- Da, W. (2011). Cisco PVST、PVST+Rapid-PVST+.
- Hurgh.org. (2014). *About Spanning Tree*. Retrieved from Hurgh.org: http://www.hurgh.org/articles.php?article=spanning_tree_about
- MSTP--H3C. (2009, 11 16). Retrieved from H3C VIP Club: http://www.h3c.com.cn/MiniSite/H3care_Club/Data_Center/Net_Reptile/The_One/Home/Catalog/200911/655244_97665_0.htm
- Radia Perlman. (2014, 4 14). Retrieved from Wikipedia: http://en.wikipedia.org/wiki/Radia_Perlman
- ZhangCun. (13, 7 22). *STP PVST CST RSTP MSTP*. Retrieved from <http://network.51cto.com/art/201307/404013.htm>

APPENDIX

Chapter 5.1 PVST+ configuration:

S1 configuration:

Current configuration : 3890 bytes

!

! Last configuration change at 00:34:47 UTC Mon Mar 1 1993

!

version 15.0

no service pad

service timestamps debug datetime msec

service timestamps log datetime msec

service password-encryption

```
!  
hostname S1  
!  
boot-start-marker  
boot-end-marker  
!  
enable secret 5 $1$Vps3$QejY1tgIfhGavMYFaj5cy.  
!  
no aaa new-model  
system mtu routing 1500  
!  
!  
no ip domain-lookup  
!  
!  
crypto pki trustpoint TP-self-signed-2205612288  
  enrollment selfsigned  
  subject-name cn=IOS-Self-Signed-Certificate-2205612288  
  revocation-check none  
  rsakeypair TP-self-signed-2205612288  
!  
!  
crypto pki certificate chain TP-self-signed-2205612288  
  certificate self-signed 01  
    3082023A 308201A3 A0030201 02020101 300D0609 2A864886 F70D0101 04050030  
    31312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274  
    69666963 6174652D 32323035 36313232 3838301E 170D3933 30333031 30303030  
    35355A17 0D323030 31303130 30303030 305A3031 312F302D 06035504 03132649  
    4F532D53 656C662D 5369676E 65642D43 65727469 66696361 74652D32 32303536  
    31323238 3830819F 300D0609 2A864886 F70D0101 01050003 818D0030 81890281  
    8100B632 56444F7E 87DA1186 8FCCD780 36D54FC8 F46666F8 72E05D4B 2981EBC6  
    549DFC22 2A7ACF8A 9D33802E A05E8A51 E61D72A4 C41A2FC7 C901E97F  
12488FB5
```

F3E90662 515CBCEC 843FBC91 6EB69E5D E6E72C94 73ADFF4D 4EB0FA10
2F079543

255C8EBA 33E1426D 7E644298 DD9142F7 E2D23A68 363F7E90 A01FAD90
48D1A006

A8350203 010001A3 62306030 0F060355 1D130101 FF040530 030101FF 300D0603
551D1104 06300482 02533130 1F060355 1D230418 30168014 0F641648 35EEC9B6
E7690DC8 9FC0ABC9 D08CB220 301D0603 551D0E04 1604140F 64164835
EEC9B6E7

690DC89F C0ABC9D0 8CB22030 0D06092A 864886F7 0D010104 05000381 81002169
6797F573 83196F2D 137C879C 49DD5839 133F1D99 E4834FB8 C28189AF F94A139B
28B457AA 0DDFEDBF 674E4A5F 54E96028 9DF3913B 1B288CB8 60250E70
B281F199

89726C78 55478A8D D9CB6DDC 048252C2 7F1AEDF3 9B1EB824 7C6C7E05
41B4CAA6

067F8A3C 045DE2C8 37AB3D22 2418A7DF 0C98F51F 30954713 709350EB 9CD9

quit

!

!

!

!

spanning-tree mode pvst

spanning-tree extend system-id

spanning-tree vlan 1,10,30,50,70 priority 24576

!

vlan internal allocation policy ascending

!

!

!

!

!

!

interface FastEthernet0/1

switchport trunk native vlan 99

switchport mode trunk


```
!  
interface FastEthernet0/2  
    switchport trunk native vlan 99  
    switchport mode trunk  
!  
interface FastEthernet0/3  
    switchport trunk native vlan 99  
    switchport mode trunk  
!  
interface FastEthernet0/4  
    switchport trunk native vlan 99  
    switchport mode trunk  
!  
interface FastEthernet0/5  
    shutdown  
!  
interface FastEthernet0/6  
    switchport access vlan 30  
    switchport mode access  
    spanning-tree portfast  
    spanning-tree bpduguard enable  
!  
interface FastEthernet0/7  
    switchport access vlan 10  
    switchport mode access  
    spanning-tree portfast  
    spanning-tree bpduguard enable  
!  
interface FastEthernet0/8  
    shutdown  
!  
interface FastEthernet0/9  
    shutdown  
!
```

```
interface FastEthernet0/10
  shutdown
!
interface FastEthernet0/11
  shutdown
!
interface FastEthernet0/12
  shutdown
!
interface FastEthernet0/13
  shutdown
!
interface FastEthernet0/14
  shutdown
!
interface FastEthernet0/15
  shutdown
!
interface FastEthernet0/16
  shutdown
!
interface FastEthernet0/17
  shutdown
!
interface FastEthernet0/18
  shutdown
!
interface FastEthernet0/19
  shutdown
!
interface FastEthernet0/20
  shutdown
!
interface FastEthernet0/21
```

```
shutdown
!
interface FastEthernet0/22
shutdown
!
interface FastEthernet0/23
switchport access vlan 20
switchport mode access
spanning-tree portfast
spanning-tree bpduguard enable
!
interface FastEthernet0/24
shutdown
!
interface GigabitEthernet0/1
shutdown
!
interface GigabitEthernet0/2
shutdown
!
interface Vlan1
no ip address
shutdown
!
interface Vlan99
ip address 172.31.99.1 255.255.255.0
!
ip http server
ip http secure-server
logging esm config
banner motd ^C
Final Thesis
^C
!
```

```
line con 0
line vty 0 4
  login
line vty 5 15
  login
!
end
```

S2 configuration:

```
Current configuration : 3832 bytes
!
! Last configuration change at 00:53:48 UTC Mon Mar 1 1993
!
version 15.0
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
service password-encryption
!
hostname S2
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$xvDw$HQaGbaUOhSPM/4z2cEy5D0
!
no aaa new-model
system mtu routing 1500
!
!
no ip domain-lookup
!
!
```

crypto pki trustpoint TP-self-signed-2205612544

enrollment selfsigned

subject-name cn=IOS-Self-Signed-Certificate-2205612544

revocation-check none

rsakeypair TP-self-signed-2205612544

!

!

crypto pki certificate chain TP-self-signed-2205612544

certificate self-signed 01

3082023A 308201A3 A0030201 02020101 300D0609 2A864886 F70D0101 04050030

31312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274

69666963 6174652D 32323035 36313235 3434301E 170D3933 30333031 30303030

35335A17 0D323030 31303130 30303030 305A3031 312F302D 06035504 03132649

4F532D53 656C662D 5369676E 65642D43 65727469 66696361 74652D32 32303536

31323534 3430819F 300D0609 2A864886 F70D0101 01050003 818D0030 81890281

8100A49F 15E8D510 F0A1A1E0 DDB8CC02 C268B9BB A305067E 7D86EF55

ABB19CFC

7BCE0A20 EBFA577D B6D1C979 7D1E2ACE D8A87AD8 1007D8BE 31FD267C

AEBB8775

9D0C4C00 3073135F FD065AE5 3543F71D A44BC42E FBF5DA88 397496EA

BF37FB17

18E7E1E9 71CC70DB 37FCC505 D1622EFF 7174F756 9A36D647 F718E9E5 93708FAE

AA8B0203 010001A3 62306030 0F060355 1D130101 FF040530 030101FF 300D0603

551D1104 06300482 02533230 1F060355 1D230418 30168014 FFF7079B E21EE91C

D6E008DE C9A2539A FE5723DA 301D0603 551D0E04 160414FF F7079BE2

1EE91CD6

E008DEC9 A2539AFE 5723DA30 0D06092A 864886F7 0D010104 05000381 81006AEB

EC8E181A 74EB1151 F0D7792F 0F2EF349 2BF1FCE9 A05C4313 132750B1 2CC76B93

B6AE5970 C7F9045E 9C1E61E4 28D5CA6C F87B3079 CCA5E768 DE3CE499

F1A150E0

39B2678C 70505852 849CE44D 83ABE6A7 AD3B01C2 2E81423A A5F61B55

BF0DF6CC

5E3C5FE5 62C9AE7C 772A9B20 D98E0622 2947C2ED 5967ABE3 F7FF5A9A A7BE

quit

```
!  
!  
!  
!  
spanning-tree mode pvst  
spanning-tree extend system-id  
spanning-tree vlan 1,10,20,30,40,50,60,70,80,99 priority 28672  
!  
vlan internal allocation policy ascending  
!  
!  
!  
!  
!  
!  
!  
interface FastEthernet0/1  
    switchport trunk native vlan 99  
    switchport mode trunk  
!  
interface FastEthernet0/2  
    switchport trunk native vlan 99  
    switchport mode trunk  
!  
interface FastEthernet0/3  
    switchport trunk native vlan 99  
    switchport mode trunk  
!  
interface FastEthernet0/4  
    switchport trunk native vlan 99  
    switchport mode trunk  
!  
interface FastEthernet0/5  
    shutdown  
!
```

```
interface FastEthernet0/6
  shutdown
!
interface FastEthernet0/7
  shutdown
!
interface FastEthernet0/8
  shutdown
!
interface FastEthernet0/9
  shutdown
!
interface FastEthernet0/10
  shutdown
!
interface FastEthernet0/11
  shutdown
!
interface FastEthernet0/12
  shutdown
!
interface FastEthernet0/13
  shutdown
!
interface FastEthernet0/14
  shutdown
!
interface FastEthernet0/15
  shutdown
!
interface FastEthernet0/16
  shutdown
!
interface FastEthernet0/17
```

```
shutdown
!
interface FastEthernet0/18
  switchport access vlan 20
  switchport mode access
  spanning-tree portfast
  spanning-tree bpduguard enable
!
interface FastEthernet0/19
  shutdown
!
interface FastEthernet0/20
  switchport access vlan 10
  switchport mode access
  spanning-tree portfast
  spanning-tree bpduguard enable
!
interface FastEthernet0/21
  shutdown
!
interface FastEthernet0/22
  shutdown
!
interface FastEthernet0/23
  shutdown
!
interface FastEthernet0/24
  shutdown
!
interface GigabitEthernet0/1
  shutdown
!
interface GigabitEthernet0/2
  shutdown
```



```
!  
interface Vlan1  
  no ip address  
!  
interface Vlan20  
  no ip address  
!  
interface Vlan99  
  ip address 172.31.99.2 255.255.255.0  
!  
ip http server  
ip http secure-server  
logging esm config  
banner motd ^C  
Final Thesis  
^C  
!  
line con 0  
line vty 0 4  
  login  
line vty 5 15  
  login  
!  
end
```

S3 configuration:

Current configuration : 3794 bytes

!

! Last configuration change at 00:40:00 UTC Mon Mar 1 1993

!

version 15.0

no service pad

service timestamps debug datetime msec

```
service timestamps log datetime msec
service password-encryption
!
hostname S3
!
boot-start-marker
boot-end-marker
!
enable secret 5 $1$CIYT$xNeKA6yKBwxJ.zrAjOllS.
!
no aaa new-model
system mtu routing 1500
!
!
no ip domain-lookup
!
!
crypto pki trustpoint TP-self-signed-2206785920
  enrollment selfsigned
  subject-name cn=IOS-Self-Signed-Certificate-2206785920
  revocation-check none
  rsakeypair TP-self-signed-2206785920
!
!
crypto pki certificate chain TP-self-signed-2206785920
  certificate self-signed 01
    3082023A 308201A3 A0030201 02020101 300D0609 2A864886 F70D0101 04050030
    31312F30 2D060355 04031326 494F532D 53656C66 2D536967 6E65642D 43657274
    69666963 6174652D 32323036 37383539 3230301E 170D3933 30333031 30303030
    35355A17 0D323030 31303130 30303030 305A3031 312F302D 06035504 03132649
    4F532D53 656C662D 5369676E 65642D43 65727469 66696361 74652D32 32303637
    38353932 3030819F 300D0609 2A864886 F70D0101 01050003 818D0030 81890281
    81009A14 E25D6373 BD9113EB 05A03EAE E3857A65 1E8C8C8D C0021DCD
    2ABBBCA7
```

```

3194B6B2 09ED07BB E1BEC202 9380D5D8 3F56B78F 1A6704FE 2000ECE5
C4EBE5CB
F2B569E8 FA928411 2F128D8E D052C235 64282C94 A32D24E3 498FC765 30D6C23D
4383A328 070BA0F3 5807A2C4 6B8B6B44 D73B141C C21552DC 54275C23
2828CBBC
75FD0203 010001A3 62306030 0F060355 1D130101 FF040530 030101FF 300D0603
551D1104 06300482 02533330 1F060355 1D230418 30168014 03140BB9 6C740F7D
5E014D68 9A4BE94A AF392651 301D0603 551D0E04 16041403 140BB96C 740F7D5E
014D689A 4BE94AAF 39265130 0D06092A 864886F7 0D010104 05000381 81009991
724AE15C C9CC10B7 19B1F4F0 1DE7935A 154AFA5E 05DCDDE2 40DCEA82
D606C71F
B07699A5 D3A27FC7 CCF49C47 E3D7DD4B 795D92E9 C9FC2A8C 43CCFD84
E92BFAE4
D3B428AB 536DEDCE B0892B10 2BFD8866 A0F4DC74 08407FE6 51256DAB
3BDA5369
9C7AE5DB 07560CE7 8D3C4CA6 DCDE7946 C8172E6D DB011704 C960564A 4DC4
quit
!
!
!
!
spanning-tree mode pvst
spanning-tree extend system-id
spanning-tree vlan 20,40,60,80,99 priority 24576
!
vlan internal allocation policy ascending
!
!
!
!
!
!
interface FastEthernet0/1
switchport trunk native vlan 99

```

```
switchport mode trunk
!
interface FastEthernet0/2
switchport trunk native vlan 99
switchport mode trunk
!
interface FastEthernet0/3
switchport trunk native vlan 99
switchport mode trunk
!
interface FastEthernet0/4
switchport trunk native vlan 99
switchport mode trunk
!
interface FastEthernet0/5
shutdown
!
interface FastEthernet0/6
shutdown
!
interface FastEthernet0/7
shutdown
!
interface FastEthernet0/8
shutdown
!
interface FastEthernet0/9
shutdown
!
interface FastEthernet0/10
shutdown
!
interface FastEthernet0/11
switchport access vlan 10
```

```
switchport mode access
spanning-tree portfast
spanning-tree bpduguard enable
!
interface FastEthernet0/12
shutdown
!
interface FastEthernet0/13
switchport access vlan 30
switchport mode access
spanning-tree portfast
spanning-tree bpduguard enable
!
interface FastEthernet0/14
shutdown
!
interface FastEthernet0/15
shutdown
!
interface FastEthernet0/16
shutdown
!
interface FastEthernet0/17
shutdown
!
interface FastEthernet0/18
shutdown
!
interface FastEthernet0/19
shutdown
!
interface FastEthernet0/20
shutdown
!
```

```
interface FastEthernet0/21
  shutdown
!
interface FastEthernet0/22
  shutdown
!
interface FastEthernet0/23
  shutdown
!
interface FastEthernet0/24
  shutdown
!
interface GigabitEthernet0/1
  shutdown
!
interface GigabitEthernet0/2
  shutdown
!
interface Vlan1
  no ip address
  shutdown
!
interface Vlan99
  ip address 172.31.99.3 255.255.255.0
!
ip http server
ip http secure-server
logging esm config
banner motd ^C
Final Thesis
^C
!
line con 0
line vty 0 4
```

```
login
line vty 5 15
login
!
end
```

Chapter 5.2 RSTP configuration

S1 configuration:

```
Current configuration : 1348 bytes
!
! Last configuration change at 00:12:47 UTC Mon Mar 1 1993
!
version 15.0
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname S1
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
system mtu routing 1500
!
!
!
!
!
```

```
!  
!  
!  
spanning-tree mode rapid-pvst  
spanning-tree extend system-id  
!  
vlan internal allocation policy ascending  
!  
!  
!  
!  
!  
!  
!  
interface FastEthernet0/1  
!  
interface FastEthernet0/2  
!  
interface FastEthernet0/3  
!  
interface FastEthernet0/4  
!  
interface FastEthernet0/5  
!  
interface FastEthernet0/6  
!  
interface FastEthernet0/7  
!  
interface FastEthernet0/8  
!  
interface FastEthernet0/9  
!  
interface FastEthernet0/10  
    switchport mode access  
    spanning-tree portfast
```



```
!  
interface FastEthernet0/11  
!  
interface FastEthernet0/12  
!  
interface FastEthernet0/13  
!  
interface FastEthernet0/14  
!  
interface FastEthernet0/15  
!  
interface FastEthernet0/16  
!  
interface FastEthernet0/17  
!  
interface FastEthernet0/18  
!  
interface FastEthernet0/19  
!  
interface FastEthernet0/20  
!  
interface FastEthernet0/21  
!  
interface FastEthernet0/22  
!  
interface FastEthernet0/23  
!  
interface FastEthernet0/24  
!  
interface GigabitEthernet0/1  
!  
interface GigabitEthernet0/2  
!  
interface Vlan1
```

```
no ip address
!  
ip http server  
ip http secure-server  
logging esm config  
!  
line con 0  
line vty 5 15  
!  
end
```

S2 configuration:

```
Current configuration : 1384 bytes  
!  
! Last configuration change at 00:07:34 UTC Mon Mar 1 1993  
!  
version 15.0  
no service pad  
service timestamps debug datetime msec  
service timestamps log datetime msec  
no service password-encryption  
!  
hostname S2  
!  
boot-start-marker  
boot-end-marker  
!  
!  
no aaa new-model  
system mtu routing 1500  
!  
!  
!
```

```
!  
!  
!  
!  
!  
!  
spanning-tree mode rapid-pvst  
spanning-tree extend system-id  
spanning-tree vlan 1 priority 28672  
!  
vlan internal allocation policy ascending  
!  
!  
!  
!  
!  
!  
!  
interface FastEthernet0/1  
!  
interface FastEthernet0/2  
!  
interface FastEthernet0/3  
!  
interface FastEthernet0/4  
!  
interface FastEthernet0/5  
!  
interface FastEthernet0/6  
!  
interface FastEthernet0/7  
!  
interface FastEthernet0/8  
!  
interface FastEthernet0/9
```

```
!  
interface FastEthernet0/10  
    switchport mode access  
    spanning-tree portfast  
!  
interface FastEthernet0/11  
!  
interface FastEthernet0/12  
!  
interface FastEthernet0/13  
!  
interface FastEthernet0/14  
!  
interface FastEthernet0/15  
!  
interface FastEthernet0/16  
!  
interface FastEthernet0/17  
!  
interface FastEthernet0/18  
!  
interface FastEthernet0/19  
!  
interface FastEthernet0/20  
!  
interface FastEthernet0/21  
!  
interface FastEthernet0/22  
!  
interface FastEthernet0/23  
!  
interface FastEthernet0/24  
!  
interface GigabitEthernet0/1
```

```
!  
interface GigabitEthernet0/2  
!  
interface Vlan1  
  no ip address  
!  
ip http server  
ip http secure-server  
logging esm config  
!  
line con 0  
line vty 5 15  
!  
end
```

S3 configuration:

Current configuration : 1346 bytes

```
!  
! Last configuration change at 00:08:44 UTC Mon Mar 1 1993  
!  
version 15.0  
no service pad  
service timestamps debug datetime msec  
service timestamps log datetime msec  
no service password-encryption  
!  
hostname S3  
!  
boot-start-marker  
boot-end-marker  
!  
!  
no aaa new-model
```

```
system mtu routing 1500
!
!
!
!
!
!
!
!
!
spanning-tree mode rapid-pvst
spanning-tree extend system-id
spanning-tree vlan 1 priority 24576
!
vlan internal allocation policy ascending
!
!
!
!
!
!
interface FastEthernet0/1
!
interface FastEthernet0/2
!
interface FastEthernet0/3
!
interface FastEthernet0/4
!
interface FastEthernet0/5
!
interface FastEthernet0/6
!
interface FastEthernet0/7
```

```
!  
interface FastEthernet0/8  
!  
interface FastEthernet0/9  
!  
interface FastEthernet0/10  
!  
interface FastEthernet0/11  
!  
interface FastEthernet0/12  
!  
interface FastEthernet0/13  
!  
interface FastEthernet0/14  
!  
interface FastEthernet0/15  
!  
interface FastEthernet0/16  
!  
interface FastEthernet0/17  
!  
interface FastEthernet0/18  
!  
interface FastEthernet0/19  
!  
interface FastEthernet0/20  
!  
interface FastEthernet0/21  
!  
interface FastEthernet0/22  
!  
interface FastEthernet0/23  
!  
interface FastEthernet0/24
```

```
!  
interface GigabitEthernet0/1  
!  
interface GigabitEthernet0/2  
!  
interface Vlan1  
  no ip address  
  shutdown  
!  
ip http server  
ip http secure-server  
logging esm config  
!  
line con 0  
line vty 5 15  
!  
end
```

Chapter 5.3 MSTP configuration

S1 configuration:

Current configuration : 1526 bytes

```
!  
! Last configuration change at 02:50:22 UTC Mon Mar 1 1993  
!  
version 15.0  
no service pad  
service timestamps debug datetime msec  
service timestamps log datetime msec  
no service password-encryption  
!  
hostname S1  
!
```



```
boot-start-marker
boot-end-marker
!
!
no aaa new-model
system mtu routing 1500
!
!
!
!
!
!
!
!
!
!
spanning-tree mode mst
spanning-tree extend system-id
!
spanning-tree mst configuration
  name region1
  revision 1
  instance 1 vlan 1, 10
  instance 2 vlan 20, 40
!
!
vlan internal allocation policy ascending
!
!
!
!
!
!
!
interface FastEthernet0/1
  switchport mode trunk
```

```
!  
interface FastEthernet0/2  
    switchport mode trunk  
!  
interface FastEthernet0/3  
!  
interface FastEthernet0/4  
!  
interface FastEthernet0/5  
!  
interface FastEthernet0/6  
!  
interface FastEthernet0/7  
!  
interface FastEthernet0/8  
!  
interface FastEthernet0/9  
!  
interface FastEthernet0/10  
    switchport access vlan 10  
    spanning-tree portfast  
!  
interface FastEthernet0/11  
!  
interface FastEthernet0/12  
!  
interface FastEthernet0/13  
!  
interface FastEthernet0/14  
!  
interface FastEthernet0/15  
    switchport access vlan 40  
!  
interface FastEthernet0/16
```

```
!  
interface FastEthernet0/17  
!  
interface FastEthernet0/18  
!  
interface FastEthernet0/19  
!  
interface FastEthernet0/20  
!  
interface FastEthernet0/21  
!  
interface FastEthernet0/22  
!  
interface FastEthernet0/23  
!  
interface FastEthernet0/24  
!  
interface GigabitEthernet0/1  
!  
interface GigabitEthernet0/2  
!  
interface Vlan1  
  no ip address  
!  
ip http server  
ip http secure-server  
  
logging esm config  
!  
line con 0  
line vty 5 15  
!  
end
```

S2 configuration:

Current configuration : 1571 bytes

!

! Last configuration change at 02:34:47 UTC Mon Mar 1 1993

!

version 15.0

no service pad

service timestamps debug datetime msec

service timestamps log datetime msec

no service password-encryption

!

hostname S2

!

boot-start-marker

boot-end-marker

!

!

no aaa new-model

system mtu routing 1500

!

!

!

!

!

!

!

!

!

spanning-tree mode mst

spanning-tree extend system-id

!

spanning-tree mst configuration

name region1

```
revision 1
instance 1 vlan 1, 10
instance 2 vlan 20, 40
!
spanning-tree mst 1 priority 28672
!
vlan internal allocation policy ascending
!
!
!
!
!
!
interface FastEthernet0/1
  switchport mode trunk
!
interface FastEthernet0/2
  switchport mode trunk
!
interface FastEthernet0/3
!
interface FastEthernet0/4
!
interface FastEthernet0/5
!
interface FastEthernet0/6
!
interface FastEthernet0/7
!
interface FastEthernet0/8
!
interface FastEthernet0/9
!
interface FastEthernet0/10
```

```
switchport access vlan 10
spanning-tree portfast
!
interface FastEthernet0/11
!
interface FastEthernet0/12
!
interface FastEthernet0/13
!
interface FastEthernet0/14
!
interface FastEthernet0/15
switchport access vlan 20
!
interface FastEthernet0/16
!
interface FastEthernet0/17
!
interface FastEthernet0/18
!
interface FastEthernet0/19
!
interface FastEthernet0/20
!
interface FastEthernet0/21
!
interface FastEthernet0/22
!
interface FastEthernet0/23
!
interface FastEthernet0/24
!
interface GigabitEthernet0/1
!
```

```
interface GigabitEthernet0/2
!
interface Vlan1
  no ip address
  shutdown
!
ip http server
ip http secure-server
logging esm config
!
line con 0
line vty 5 15
!
end
```

S3 configuration

Current configuration : 1492 bytes

```
!
! Last configuration change at 02:43:19 UTC Mon Mar 1 1993
!
version 15.0
no service pad
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname S3
!
boot-start-marker
boot-end-marker
!
!
```

```
no aaa new-model
system mtu routing 1500
!
!
!
!
!
!
!
!
!
spanning-tree mode mst
spanning-tree extend system-id
!
spanning-tree mst configuration
  name region1
  revision 1
  instance 1 vlan 1, 10
  instance 2 vlan 20, 40
!
spanning-tree mst 1 priority 4096
!
vlan internal allocation policy ascending
!
!
!
!
!
!
!
interface FastEthernet0/1
  switchport mode trunk
!
interface FastEthernet0/2
  switchport mode trunk
```



```
!  
interface FastEthernet0/3  
!  
interface FastEthernet0/4  
!  
interface FastEthernet0/5  
!  
interface FastEthernet0/6  
!  
interface FastEthernet0/7  
!  
interface FastEthernet0/8  
!  
interface FastEthernet0/9  
!  
interface FastEthernet0/10  
!  
interface FastEthernet0/11  
!  
interface FastEthernet0/12  
!  
interface FastEthernet0/13  
!  
interface FastEthernet0/14  
!  
interface FastEthernet0/15  
!  
interface FastEthernet0/16  
!  
interface FastEthernet0/17  
!  
interface FastEthernet0/18  
!  
interface FastEthernet0/19
```

```
!  
interface FastEthernet0/20  
!  
interface FastEthernet0/21  
!  
interface FastEthernet0/22  
!  
interface FastEthernet0/23  
!  
interface FastEthernet0/24  
!  
interface GigabitEthernet0/1  
!  
interface GigabitEthernet0/2  
!  
interface Vlan1  
  no ip address  
  shutdown  
!  
ip http server  
ip http secure-server  
logging esm config  
!  
line con 0  
line vty 5 15  
!  
end
```