Capitolo 4

Test su Port Mirroring e Policy Classification

Questo capitolo mostra alcuni test svolti per analizzare i pacchetti, spesso utilizzando la funzionalità di Port Mirroring. Riporta inoltre un test sulla funzione di Policy Classification.

4.1 Test Port Mirroring

Test	Test Port Mirroring
Object	Enterasys SecureStack B2H124-48P (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24; Enterasys SecureStack C2G124-24 (24 ports) Chassis Serial Number: 04400122900B Chassis Firmware Revision: 04.00.24
Target	To verify the functionality called Port Mirroring on switch Enterasys entry level.
Designer	Giulia Papini
Executor	Giulia Papini (01/02/2007)
Result	Positive

Object of the experiment

The object of the experiment is composed of two stackable switch Enterasys:

- SW1: Enterasys SecureStack B2

  B2H124-48P (48 ports)

  Chassis Serial Number: 05320111900B

  Chassis Firmware Revision: 03.01.24

- SW2: Enterasys SecureStack C2

  C2G124-24 (24 ports)

  Chassis Serial Number: 04400122900B

  Chassis Firmware Revision: 04.00.24

Target of the experiment

To verify the functionality called Port Mirroring on switch Enterasys entry level.

Expected results

The expected results is the possibility to address on a chosen port the whole traffic between two switches, in order to be able to observe and study it.

Rationales

It is chosen to use a linear configuration with two switch in order to show that the whole traffic between them can be addressed towards one port of one switch for being analyzed easily.

Tools

The test platform is composed of a personal computer:

- PC1: PC Notebook HP-DV2172EA 1.66 GHz/980 MHz with 1 Gb of RAM;

Moreover it's used the program 'Wireshark' that 'sniffs' the packages, it captures and analyzes them, showing their information and fields.

Configuration

The switches have one setting priority. A bridge priority of default, correspondent to 32768, is assigned to both switches.

All the subnet 192.168.0.0/24 is assigned to VLAN 1 (default VLAN); on all the network device is configured the traditional Spanning Tree Protocol and the level two protocols, GVRP and LACP, are disabled.

The level three configuration assigned to both switches is the follow: IP address:

- SW1: 192.168.1.1/24;
- SW2: 192.168.1.2/24;

The level three configuration assigned to personal computer is the follow: IP address:

- PC1: 192.168.1.11;

Topology

In figure 4.1 is represented the topology used on physical level for the test.

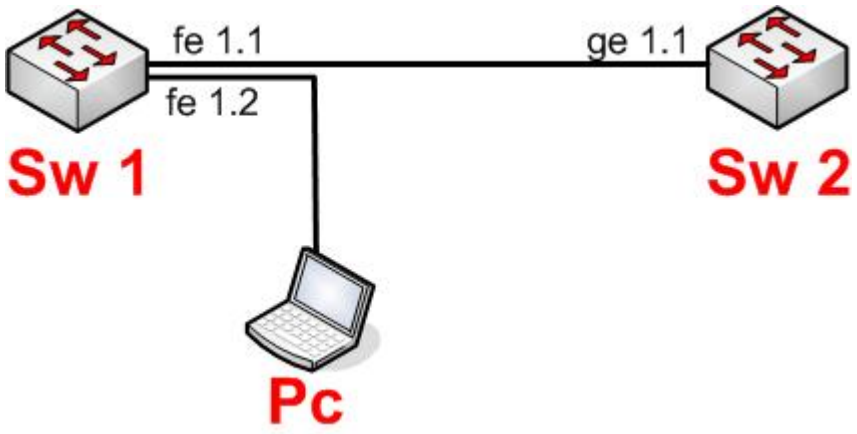

Figura 4.1: Topology of Port Mirroring

Operations

1. On both switches execute the following command to delete the pre-existing settings:

```
SW 1,2:> clear config all
```

2. On SW1 execute the following command to set the IP address:

```
SW1:> set ip address 192.168.1.1 mask 255.255.255.0
```

3. On SW2 execute the following command to set the IP address:

```
SW2:> set ip address 192.168.1.2 mask 255.255.255.0
```

4. On both switches execute the following command to set the Spanning Tree Protocol:

```
SW 1,2:> set spantree version stpcompatible
```

5. On both switches execute the following command to disable the GVRP Protocol:

```
SW 1,2:> set gvrp disable
```

6. On both switches execute the following command to disable the LACP Protocol:

```
SW 1,2:> set lacp disable
```

7. On SW1 execute the following command to set the port linked with the second switch (fe.1.1) as 'source port' and the port linked with PC1 (fe.1.2) as 'destination port':

```
SW1:> set port mirroring create fe.1.1 fe.1.2
```

8. On SW1 execute the following command to enable the port linked with the second switch (fe.1.1) as 'source port' and the port linked with PC1 (fe.1.2) as 'destination port':

```
SW1:> set port mirroring enable fe.1.1 fe.1.2
```

9. On SW1 execute the following command to verify from PC1, with program Wireshark, if effectively the traffic between the two switches is addressed on 'destination port':

```
SW1:> ping 192.168.1.2
```

Execution

1. The execution of the operations 7 and 8 has produced the following result:

```
      Port Mirroring
==============
Source Port     = fe.1.1
Target Port     = fe.1.2
Frames Mirrored = Rx and Tx
Port Mirroring status enabled
```

2. The execution of the operation 9 has produced a positive result: using the program Wireshark is effectively possible to observe and analyze all the packages between the two switches.

Test Result

Seen the results obtained in point 9 of execution of the test is possible to assert that the functionality of port mirroring can be used on the switches Enterasys.

The test result is positive.

4.2 Test Configuration BPDU

Test	Test Configuration BPDU
Object	Enterasys SecureStack B2H124-48P (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24; Enterasys SecureStack C2G124-24 (24 ports) Chassis Serial Number: 04400122900B Chassis Firmware Revision: 04.00.24
Target	To use the port mirroring functionality in order to analyze the packages of type configuration BPDU excanges between switches Enterasys entry level.
Designer	Giulia Papini
Executor	Giulia Papini (12/02/2007)
Result	Positive

Object of the experiment

The object of the experiment is composed of two stackable switch Enterasys:

- SW1: Enterasys SecureStack B2

  B2H124-48P (48 ports)

  Chassis Serial Number: 05320111900B

  Chassis Firmware Revision: 03.01.24

- SW2: Enterasys SecureStack C2

  C2G124-24 (24 ports)

  Chassis Serial Number: 04400122900B

  Chassis Firmware Revision: 04.00.24

Prerequisites

Prerequisites regarding the verification of the Port Mirroring functionality are necessary to the development of the test. Before executing the test it must be assessed that Port Mirroring functions as related in detailed list. (see Test 4.1 'Port Mirroring').

Target of the experiment

To use the port mirroring functionality in order to analyze the packages of type configuration BPDU excanges between switches Enterasys entry level.

Expected results

The expected result is the possibility to address on one chosen port the traffic between two switches to come to terms about Spanning Tree, in order to observe and study (in particular it will be studied one package of type configuration BPDU). With the used configuration the port ge.1.3 state will be 'discarding'.

Rationales

It is chosen to use a linear configuration with two switches with a cycle in order to allow the two to generate packages of agreement for the Spanning Tree (bpdu IEEE 802.3 Ethernet) to analize their informations and field.

Tools

The test platform is composed of a personal computer:

- PC1: PC Notebook HP-DV2172EA 1.66 GHz/980 MHz with 1 Gb of RAM;

Moreover it's used the program 'Wireshark' that 'sniff' the packages, it capture and analyzes them, showing theirs information and fields.

Configuration

The switches have one setting priority. The highest is assigned to SW1, to SW2 the lowest one. The priority adopts a position with increments of 4096, the highest one is indicated with the number 0, the lowest one with number 61440.

The bridge priority of both switches it's set up so:

- SW1: 0;
- SW2: 4096;

All subnet 192.168.0.0 /24 is associated to the VLAN 1 (default VLAN); on all the network devices are shaped traditional protocol STP and are disabled the level-two protocols GVRP and LACP.

The level-three configuration assigned to every switch is the follow: IP address:

- SW1: 192.168.1.1/24;
- SW2: 192.168.1.2/24;

The level-three configuration assigned personal computer is the follow: IP address:

- PC1: 192.168.1.11;

Topology

In figure 4.2 is represented the topology used on physical level for the test.

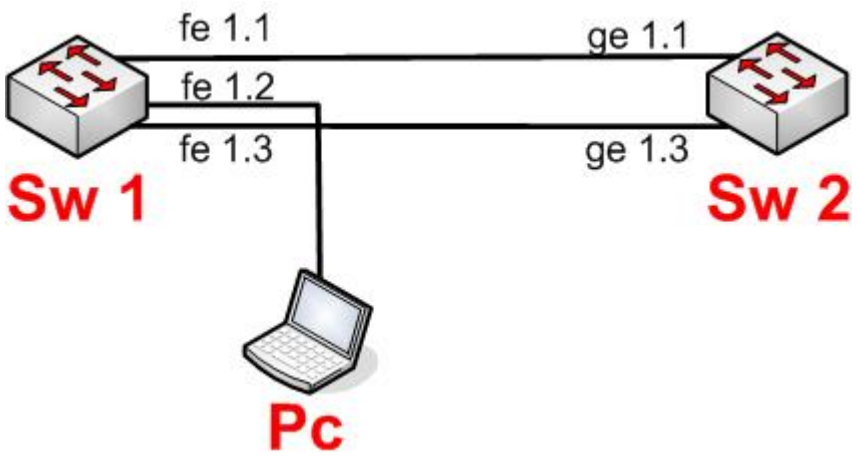

Figura 4.2: Topology Configuration BPDU

Operation

1. On both switches execute the following command to delete the pre-existing settings:

```
SW 1,2:> clear config all
```

2. On SW1 execute the following command to set the IP address:

```
SW1:> set ip address 192.168.1.1 mask 255.255.255.0
```

3. On SW2 execute the following command to set the IP address:

```
SW2:> set ip address 192.168.1.2 mask 255.255.255.0
```

4. On both switches execute the following command to set the Spanning Tree Protocol:

```
SW 1,2:> set spantree version stpcompatible
```

5. On both switches execute the following command to disable the GVRP Protocol:

```
SW 1,2:> set gvrp disable
```

6. On both switches execute the following command to disable the LACP Protocol:

```
SW 1,2:> set lacp disable
```

7. On SW1 execute the following command to set the priority:

```
SW1:> set spantree priority 0
```

8. On SW2 execute the following command to set the priority:

```
SW2:> set spantree priority 4096
```

9. On SW1 execute the following command to set the port linked with the second switch (fe.1.3) as 'source port' and the port linked with PC1 (fe.1.2) as 'destination port':

```
SW1:> set port mirroring create fe.1.3 fe.1.2
```

10. On SW1 execute the following command to enable the port linked with the second switch (fe.1.3) as 'source port' and the port linked with PC1 (fe.1.2) as 'destination port':

```
SW1:> set port mirroring enable fe.1.3 fe.1.2
```

11. On SW1 execute the following command to verify the port fe.1.1 state:

```
SW1:> show spantree stats port fe.1.1
```

12. On SW1 execute the following command to verify the port fe.1.3 state:

```
SW1:> show spantree stats port fe.1.3
```

13. On SW2 execute the following command to verify the port ge.1.1 state:

```
SW2:> show spantree stats port ge.1.1
```

14. On SW2 execute the following command to verify the port ge.1.3 state:

```
SW2:> show spantree stats port ge.1.3
```

15. On Sw1 execute the following command to analize from PC1, with program Wireshark, traffic between the two switch:

```
SW1:> ping 192.168.1.2
```

Execution

1. The execution of the operations 11 and 12 has produced the following result:

```
SID   Port         State              Role          Cost        Priority
---   ----------   ----------------   -----------   --------    --------
0     fe.1.1       Forwarding         Designated    200000      128

SID   Port         State              Role          Cost        Priority
---   ----------   ----------------   -----------   --------    --------
0     fe.1.3       Forwarding         Designated    200000      128
```

2. The execution of the operations 13 and 14 has produced the following result:

```
SID   Port         State              Role          Cost        Priority
---   ----------   ----------------   -----------   --------    --------
0     ge.1.1       Forwarding         Root          200000      128

SID   Port         State              Role          Cost        Priority
---   ----------   ----------------   -----------   --------    --------
0     ge.1.3       Discarding         Alternate     200000      128
```

The execution of the operation 15 has produced a positive result: using the program Wireshark is effectively possible to observe and analyze all the packages between the two switches.

Test Result

Seen the results obtained in point 15 of execution of the test it is possible to study information and fields of a configuration BPDU. The test result is positive.

Comments

Information and fields of one of the sniffed packages:

```
Frame 1 (60 bytes on wire, 60 bytes captured)
Arrival Time: Feb 14, 2007 11:47:39.395938000
Time delta from previous packet: 0.000000000 seconds
Time since reference or first frame: 0.000000000 seconds
Frame Number: 1
Packet Length: 60 bytes
Capture Length: 60 bytes
Frame is marked: False
Protocols in frame: eth:llc:stp
Coloring Rule Name: Broadcast
Coloring Rule String: eth[0] & 1

0000   01 80 c2 00 00 00 00 11 88 1a 19 e2 00 26 42 42  .............&BB
0010   03 00 00 00 00 00 00 00 00 11 88 1a 19 e1 00 00  ................
0020   00 00 00 00 00 11 88 1a 19 e1 80 01 00 00 14 00  ................
0030   02 00 0f 00 00 00 00 00 00 00 00 00              ............

IEEE 802.3 Ethernet
Destination: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
Address: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
.... ...1 .... .... .... .... = IG bit: Group address (multicast/broadcast)
.... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
Source: Enterasy_1a:19:e2 (00:11:88:1a:19:e2)
Address: Enterasy_1a:19:e2 (00:11:88:1a:19:e2)
.... ...0 .... .... .... .... = IG bit: Individual address (unicast)
.... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
Length: 38
Trailer: 0000000000000000


Logical-Link Control
DSAP: Spanning Tree BPDU (0x42)
IG Bit: Individual
SSAP: Spanning Tree BPDU (0x42)
CR Bit: Command
Control field: U, func=UI (0x03)
00. 00.. = Command: Unnumbered Information (0x00)
.... ..11 = Frame type: Unnumbered frame (0x03)
```

```
Spanning Tree Protocol
Protocol Identifier: Spanning Tree Protocol (0x0000)
Protocol Version Identifier: Spanning Tree (0)
BPDU Type: Configuration (0x00)
BPDU flags: 0x00
0... .... = Topology Change Acknowledgment: No
.... ...0 = Topology Change: No
Root Identifier: 0 / 00:11:88:1a:19:e1
Root Path Cost: 0
Bridge Identifier: 0 / 00:11:88:1a:19:e1
Port identifier: 0x8001
Message Age: 0
Max Age: 20
Hello Time: 2
Forward Delay: 15
```

4.3 Test TopologyChange BPDU

Test	Test TopologyChange BPDU
Object	Enterasys SecureStack B2H124-48P (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24; Enterasys SecureStack C2G124-24 (24 ports) Chassis Serial Number: 04400122900B Chassis Firmware Revision: 04.00.24
Target	To use the port mirroring functionality in order to analyze the packages of type Topology Change Notification excanges between switches Enterasys entry level.
Designer	Giulia Papini
Executor	Giulia Papini (20/02/2007)
Result	Positive

Object of the experiment

The object of the experiment is composed of two stackable switches Enterasys:

- SW1: Enterasys SecureStack B2

  B2H124-48P (48 ports)

  Chassis Serial Number: 05320111900B

  Chassis Firmware Revision: 03.01.24

- SW2: Enterasys SecureStack C2

  C2G124-24 (24 ports)

  Chassis Serial Number: 04400122900B

  Chassis Firmware Revision: 04.00.24

Prerequisites

Prerequisites regarding the verification of the Port Mirroring and Spanning Tree functionality are necessary to the development of the test.

Target of the experiment

To use the port mirroring functionality in order to analyze the packages of type Topology Change Notification excanges between switches Enterasys entry level.

Expected results

The expected result is the possibility to address on one chosen port the whole traffic between two switches in order to observe it and study it. In particular it will come to fall a link between two switches so that the devices will generate a package of type Topology Change BPDU. With the used configuration the port ge.1.3 state will be 'discarding'.

Rationales

It is chosen to use a linear configuration with two switches with a cycle and in a given istant it will come to fall a link in order to generate the wanted package.

Tools

The test platform is composed of a personal computer:

- PC1: PC Notebook HP-DV2172EA 1.66 GHz/980 MHz with 1 Gb of RAM;

Moreover it's used the program 'Wireshark' that 'sniffs' the packages, it capture and analyzes them, showing their information and fields.

Configuration

The switches has one setting priority. The highest is assigned to SW1, to switch 2 the lowest one. The priority adopts a position with increments of 4096, the highest one is indicated with the number 0, the lowest one with number 61440. The bridge priority of everyone switch it's set up so:

- SW1: 0;
- SW2: 4096;

All subnet 192.168.0.0 /24 is associated to the VLAN 1 (default VLAN); on all the network devices are shaped traditional protocol STP and are disabled the level-two protocols GVRP and LACP.

The level-three configuration assigned every switch is the follow: IP address:

- SW1: 192.168.1.1/24;
- SW2: 192.168.1.2/24;

The level-three configuration assigned to the personal computer is the follow: IP address:

- PC1: 192.168.1.11;

Topology

In figure 4.3 is represented the topology used on physical level for the test.

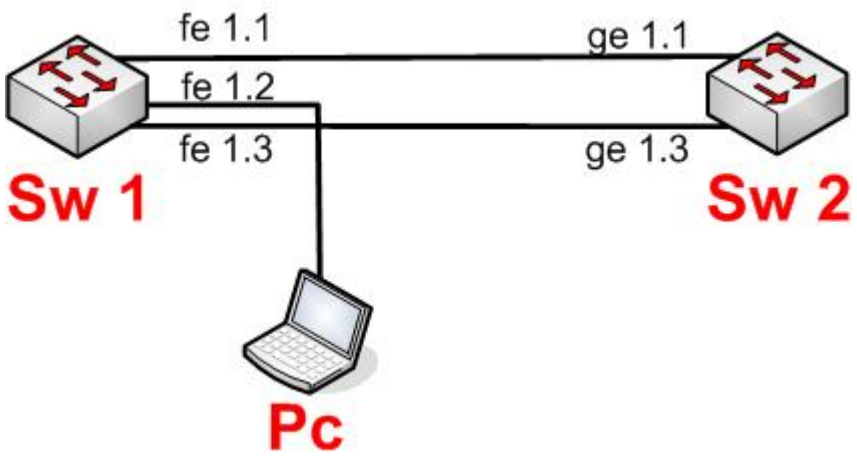

Figura 4.3: Topology TopologyChange BPDU

Operation

1. On both switches execute the following command to delete the pre-existing settings:

```
SW 1,2:> clear config all
```

2. On SW1 execute the following command to set the IP address:

```
SW1:> set ip address 192.168.1.1 mask 255.255.255.0
```

3. On SW2 execute the following command to set the IP address:

```
SW2:> set ip address 192.168.1.2 mask 255.255.255.0
```

4. On both switches execute the following command to set the Spanning Tree Protocol:

```
SW 1,2:> set spantree version stpcompatible
```

5. On both switches execute the following command to disable the GVRP Protocol:

```
SW 1,2:> set gvrp disable
```

6. On both switches execute the following command to disable the LACP Protocol:

```
SW 1,2:> set lacp disable
```

7. On SW1 execute the following command to set the priority:

```
SW1:> set spantree priority 0
```

8. On SW2 execute the following command to set the priority:

```
SW2:> set spantree priority 4096
```

9. On SW1 execute the following command to set the port linked with the second switch (fe.1.3) as 'source port' and the port linked with PC1 (fe.1.2) as 'destination port':

```
SW1:> set port mirroring create fe.1.3 fe.1.2
```

10. On SW1 execute the following command to enable the port linked with the second switch (fe.1.3) as 'source port' and the port linked with PC1 (fe.1.2) as 'destination port':

```
SW1:> set port mirroring enable fe.1.3 fe.1.2
```

11. On SW1 execute the following command to verify the port fe.1.1 state:

```
SW1:> show spantree stats port fe.1.1
```

12. On SW1 execute the following command to verify the port fe.1.3 state:

```
SW1:> show spantree stats port fe.1.3
```

13. On SW2 execute the following command to verify the port ge.1.1 state:

```
SW2:> show spantree stats port ge.1.1
```

14. On SW2 execute the following command to verify the port ge.1.3 state:

```
SW2:> show spantree stats port ge.1.3
```

15. On Sw1 execute the following command to analize from PC1 the traffic between the two switches with program Wireshark:

```
SW1:> ping 192.168.1.2
```

16. While the previous command is still executing, it comes physically disconnected the link of the port ge.1.1 of switch 2 and it is analyzed from the PC1, with the program Wireshark, the traffic between the two switches.

Execution

1. The execution of operations 11 and 12 has produced the following result:

```
SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  0     fe.1.1       Forwarding         Designated    200000      128

  SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  0     fe.1.3       Forwarding         Designated    200000      128
```

2. The execution of operations 13 and 14 has produced the following result:

```
  SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  0     ge.1.1       Forwarding         Root          200000      128

  SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  0     ge.1.3       Discarding         Alternate     200000      128
```

The execution of the operation 15 has produced a positive result: using the program Wireshark is effectively possible to observe and analyze all the packages between the two switches.

Test Result

Seen the results obtained in point 15 of execution of the test is possible to study the information and fields of a TopologyChange BPDU. The test result is positive.

Comments

Information and fields of one of the sniffed packages:

```
Frame 7 (60 bytes on wire, 60 bytes captured)
Arrival Time: Feb 14, 2007 12:37:24.625175000
Time delta from previous packet: 0.150556000 seconds
Time since reference or first frame: 4.298507000 seconds
Frame Number: 7
Packet Length: 60 bytes
Capture Length: 60 bytes
Frame is marked: False
Protocols in frame: eth:llc:stp
Coloring Rule Name: Spanning Tree Topology  Change
Coloring Rule String: stp.type == 0x80

0000   01 80 c2 00 00 00 00 01 f4 5c 3f 23 00 07 42 42  .........\?#..BB
0010   03 00 00 00 80 00 00 00 00 00 00 00 00 00 00 00  ................
0020   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
0030   00 00 00 00 00 00 00 00 00 00 00 00              ............


IEEE 802.3 Ethernet
Destination: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
Address: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
.... ...1 .... .... .... .... = IG bit: Group address (multicast/broadcast)
.... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
Source: Enterasy_5c:3f:23 (00:01:f4:5c:3f:23)
Address: Enterasy_5c:3f:23 (00:01:f4:5c:3f:23)
.... ...0 .... .... .... .... = IG bit: Individual address (unicast)
.... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
Length: 7
Trailer: 000000000000000000000000000000000000000000000000...


Logical-Link Control
```

```
DSAP: Spanning Tree BPDU (0x42)
IG Bit: Individual
SSAP: Spanning Tree BPDU (0x42)
CR Bit: Command
Control field: U, func=UI (0x03)
00. 00.. = Command: Unnumbered Information (0x00)
.... ..11 = Frame type: Unnumbered frame (0x03)


Spanning Tree Protocol
Protocol Identifier: Spanning Tree Protocol (0x0000)
Protocol Version Identifier: Spanning Tree (0)
BPDU Type: Topology Change Notification (0x80)
```

4.4 Test Tagged BPDU

Test	Test Tagged BPDU
Object	Enterasys SecureStack B2H124-48P (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24; Enterasys SecureStack C2H124-48 (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24
Target	To use an hub connected to a PC and two switches in order to analyze the packages of IEEE type 802.1Q, exchanges between switches Enterasys entry level.
Designer	Giulia Papini
Executor	Giulia Papini (12/03/2007)
Result	Positive

Object of the experiment

The object of the experiment is composed of two stackable switches Enterasys:

- SW1: Enterasys SecureStack B2

  B2H124-48P (48 ports)

  Chassis Serial Number: 05320111900B

  Chassis Firmware Revision: 03.01.24

- SW2: Enterasys SecureStack C2

  C2H124-48 (24 ports)

  Chassis Serial Number: 05320111900B

  Chassis Firmware Revision: 03.01.24

Target of the experiment

To use an hub connected to a PC and two switch in order to analyze the packages of IEEE type 802.1Q, exchanges between switches Enterasys entry level.

Expected results

The expected result is the possibility to observe on a PC connected to an hub the traffic between two switches on which is setted the IEEE 802.1Q protocol.

Rationales

It is chosen to use a linear configuration with two switch whose ports is setting on the IEEE 802.1Q protocol. It is so generated one trunk 1Q link, from which is possible to sniff the packages that are tagged.

Tools

The test platform is composed of three personal computer:

- PC1: PC Notebook HP-DV2172EA 1.66 GHz/980 MHz with 1 Gb of RAM;
- PC2: PC Desktop INTEL PENTIUM II 500 MHz with 256 Mb of RAM;
- PC3: PC Desktop INTEL PENTIUM II 350 MHz with 256 Mb of RAM;

Moreover it's used the program 'Wireshark' that 'sniffs' the packages, it capture and analyzes them, showing their information and fields.

Configuration

The switches have one setting priority. A bridge priority of default, correspondent to 32768, is assigned to both switches.

On all the network devices are shaped traditional protocol STP and the level-two protocols, GVRP and LACP, are disabled.

The level-three configuration assigned to every switch is the follow:

IP address:

- SW1: 192.168.1.1/24;
- SW2: 192.168.1.2/24;

The level-three configuration assigned to the personal computers is the follow:

IP address:

- PC1: 192.168.1.11;
- PC2: 192.168.1.22;
- PC3: 192.168.1.33;

Topology

In figure 4.4 is represented the topology used on physical level for the test.

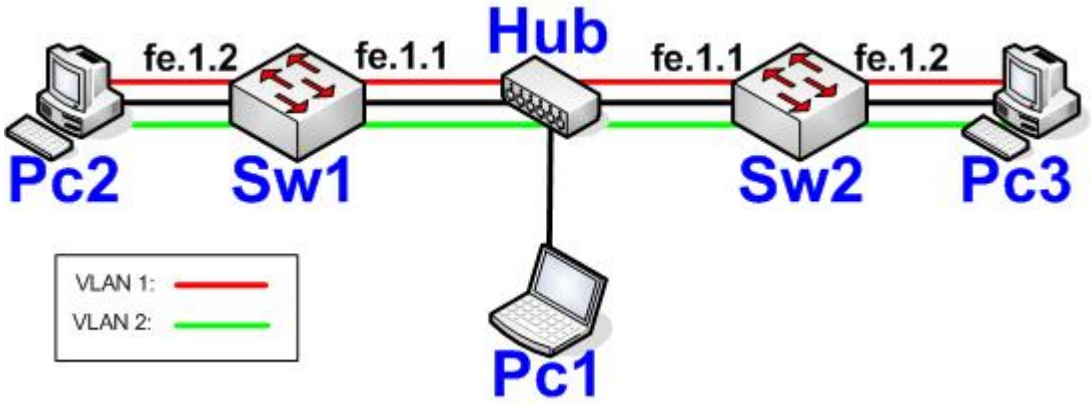

Figura 4.4: Topology Tagged BPDU

Operations

1. On both switches execute the following command to delete the pre-existing settings:

```
SW 1,2:> clear config all
```

2. On SW1 execute the following command to set the IP address:

```
SW1:> set ip address 192.168.1.1 mask 255.255.255.0
```

3. On SW2 execute the following command to set the IP address:

```
SW2:> set ip address 192.168.1.2 mask 255.255.255.0
```

4. On both switches execute the following command to set the Spanning Tree Protocol:

```
SW 1,2:> set spantree version stpcompatible
```

5. On both switches execute the following command to disable the GVRP Protocol:

```
SW 1,2:> set gvrp disable
```

6. On both switches execute the following command to disable the LACP Protocol:

```
SW 1,2:> set lacp disable
```

7. On both switches execute the following command to create the VLAN with PVID 2:

```
SW 1,2:> set vlan create 2
```

8. On both switches execute the following command to create the VLAN with PVID 3:

```
SW 1,2:> set vlan create 3
```

9. On SW1 execute the following command to assign the port fe.1.1 to VLAN 2:

```
SW1:> set port vlan fe.1.1 2
```

10. On SW1 execute the following command to assign the port fe.1.2 to VLAN 2:

```
SW1:> set port vlan fe.1.2 2
```

11. On SW2 execute the following command to assign the port fe.1.1 to VLAN 2:

```
SW2:> set port vlan fe.1.1 2
```

12. On SW2 execute the following command to assign the port fe.1.2 to VLAN 2:

```
SW2:> set port vlan fe.1.2 2
```

13. On SW1 execute the following command to allow to port fe.1.1 to receive packages pertaining to VLAN 2:

```
SW1:> set vlan egress 2 fe.1.1 tagged
```

14. On SW1 execute the following command to allow to port fe.1.2 to receive packages pertaining to VLAN 2 (it is important to set the port as untagged because the PCs can't assign tag to packages, so SW1 would not accept them):

```
SW1:> set vlan egress 2 fe.1.2 untagged
```

15. On SW1 execute the following command to allow to port fe.1.1 to receive packages pertaining to VLAN 3:

```
SW1:> set vlan egress 3 fe.1.1 tagged
```

16. On SW1 execute the following command to allow to port fe.1.2 to receive packages pertaining to VLAN 3 (it is important to set the port as untagged because the PCs can't assign tag to packages, so SW1 would not accept them):

```
SW1:> set vlan egress 3 fe.1.2 untagged
```

17. On SW2 execute the following command to allow to port fe.1.1 to receive packages pertaining to VLAN 2:

```
SW2:> set vlan egress 2 fe.1.1 tagged
```

18. On SW2 execute the following command to allow to port fe.1.2 to receive packages pertaining to VLAN 2 (it is important to set the port as untagged because the PCs can't assign tag to packages, so SW1 would not accept them):

```
SW2:> set vlan egress 2 fe.1.2 untagged
```

19. On SW2 execute the following command to allow to port fe.1.1 to receive packages pertaining to VLAN 3:

```
SW2:> set vlan egress 3 fe.1.1 tagged
```

20. On SW2 execute the following command to allow to port fe.1.2 to receive packages pertaining to VLAN 3 (is important set the port as untagged because the PCs can't assign tag to packages, so SW1 would not accept them):

```
SW2:> set vlan egress 3 fe.1.2 untagged
```

21. On PC2 execute the following command to verify if it shares with PC3 and to analyze from PC1, with program Wireshark, traffic between the two switches:

```
PC2:> ping 192.168.1.33
```

Execution

The execution of operation 21 has produced a positive result: using the program Wireshark is effectively possible to observe and analyze all the packages between the two switches, that effectively are tagged packages.

Test Result

Seen the results obtained in point 21 of execution of the test is possible to study the informations and field of a tagged BPDU. The test result is positive.

Comments

Information and fields of one of the sniffed packages:

```
Frame 20 (64 bytes on wire, 64 bytes captured)
Arrival Time: Mar 29, 2007 13:06:26.834722000
Time delta from previous packet: 1.067952000 seconds
Time since reference or first frame: 33.066399000 seconds
Frame Number: 20
Packet Length: 64 bytes
Capture Length: 64 bytes
Frame is marked: False
Protocols in frame: eth:vlan:arp
Coloring Rule Name: ARP
Coloring Rule String: arp

0000   ff ff ff ff ff ff 00 50 da bf cc e5 81 00 00 02  .......P........
0010   08 06 00 01 08 00 06 04 00 01 00 50 da bf cc e5  ...........P....
0020   c0 a8 01 16 00 00 00 00 00 00 c0 a8 01 21 00 00  .............!..
0030   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................


Ethernet II, Src: 3com_bf:cc:e5 (00:50:da:bf:cc:e5), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Destination: Broadcast (ff:ff:ff:ff:ff:ff)
Address: Broadcast (ff:ff:ff:ff:ff:ff)
.... ...1 .... .... .... .... = IG bit: Group address (multicast/broadcast)
.... ..1. .... .... .... .... = LG bit: Locally administered address (this is NOT the factory default)
Source: 3com_bf:cc:e5 (00:50:da:bf:cc:e5)
Address: 3com_bf:cc:e5 (00:50:da:bf:cc:e5)
.... ...0 .... .... .... .... = IG bit: Individual address (unicast)
.... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
Type: 802.1Q Virtual LAN (0x8100)


802.1Q Virtual LAN
000. .... .... .... = Priority: 0
...0 .... .... .... = CFI: 0
.... 0000 0000 0010 = ID: 2
Type: ARP (0x0806)
Trailer: 000000000000000000000000000000000000


Address Resolution Protocol (request)
```

```
Hardware type: Ethernet (0x0001)
Protocol type: IP (0x0800)
Hardware size: 6
Protocol size: 4
Opcode: request (0x0001)
Sender MAC address: 3com_bf:cc:e5 (00:50:da:bf:cc:e5)
Sender IP address: 192.168.1.22 (192.168.1.22)
Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
Target IP address: 192.168.1.33 (192.168.1.33)
```

4.5 Test Policy Classification

Test	Test Policy Classification
Object	Enterasys SecureStack B2H124-48P (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24; Enterasys SecureStack C2G124-24 (24 ports) Chassis Serial Number: 04400122900B Chassis Firmware Revision: 04.00.24
Target	To verify the functionality called Policy Classification on switches Enterasys entry level.
Designer	Giulia Papini
Executor	Giulia Papini (15/02/2007)
Result	Positive

Object of the experiment

The object of the experiment is composed of two stackable switches Enterasys:

- SW1: Enterasys SecureStack B2

  B2H124-48P (48 ports)

  Chassis Serial Number: 05320111900B

  Chassis Firmware Revision: 03.01.24

- SW2: Enterasys SecureStack C2

  C2G124-24 (24 ports)

  Chassis Serial Number: 04400122900B

  Chassis Firmware Revision: 04.00.24

Target of the experiment

To verify the functionality called Policy Classification on switches Enterasys entry level.

Expected results

The expected results is the possibility to manage the traffic flow of the SW1, tagging the frames coming from PC2 as pertaining to VLAN 2.

Rationales

It is chosen to use a linear configuration with two switches whose ports is set on IEEE 802.1Q protocol. Is so generated one trunk 1Q link, from which is possible to sniffs the packages. On the port of SW1 is set the Policy Classification.

Tools

The test platform is composed of three personal computer:

- PC1: PC Notebook HP-DV2172EA 1.66 GHz/980 MHz with 1 Gb of RAM;
- PC2: PC Desktop INTEL PENTIUM II 500 MHz with 256 Mb of RAM;
- PC3: PC Desktop INTEL PENTIUM II 350 MHz with 256 Mb of RAM;

Moreover it's used the program 'Wireshark' that 'sniffs' the packages, it capture and analyzes them, showing their information and fields.

Configuration

The switches has one setting priority. A bridge priority of default, correspondent to 32768 is assigned to both switches.

On all the network devices are shaped traditional protocol STP and the level-two protocols, GVRP and LACP, are disabled.

The level-three configuration assigned everyone switch is the follow:

IP address:

- SW1: 192.168.1.1/24;
- SW2: 192.168.1.2/24;

The level-three configuration assigned to the personal computers is the follow:

IP address:

- PC1: 192.168.1.11;
- PC2: 192.168.1.22;
- PC3: 192.168.1.33;

Topology

In figure 4.5 is represented the topology used on physical level for the test.

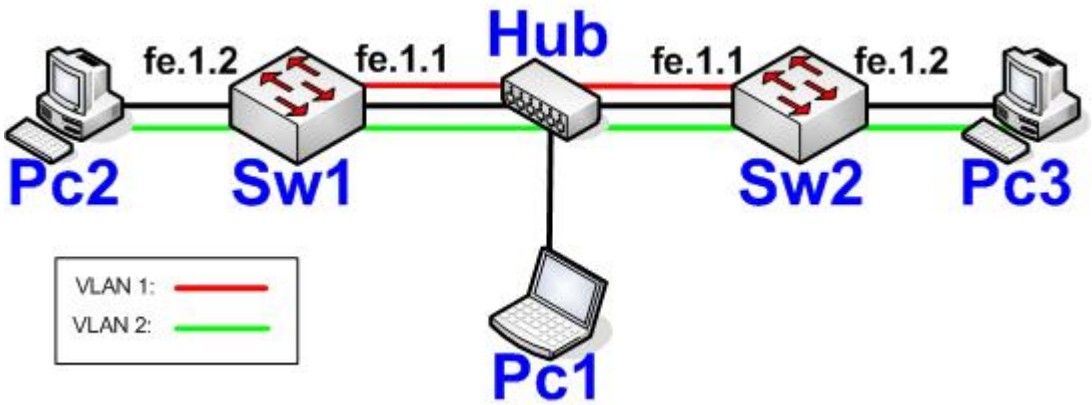

Figura 4.5: Topology of Policy Classification

Operations

1. On both switches execute the following command to delete the pre-existing settings:

```
SW 1,2:> clear config all
```

2. On SW1 execute the following command to set the IP address:

```
SW1:> set ip address 192.168.1.1 mask 255.255.255.0
```

3. On SW2 execute the following command to set the IP address:

```
SW2:> set ip address 192.168.1.2 mask 255.255.255.0
```

4. On both switches execute the following command to disable the GVRP Protocol:

```
SW 1,2:> set gvrp disable
```

5. On both switches execute the following command to disable the LACP Protocol:

```
SW 1,2:> set lacp disable
```

6. On both switches execute the following command to create the VLAN with PVID 2:

```
SW 1,2:> set vlan create 2
```

7. On SW1 execute the following command to assign the port fe.1.1 to VLAN 2:

```
SW1:> set port vlan fe.1.1 2
```

8. On SW1 execute the following command to assign the port fe.1.2 to VLAN 2:

```
SW1:> set port vlan fe.1.2 2
```

9. On SW2 execute the following command to assign the port fe.1.1 to VLAN 2:

```
SW2:> set port vlan fe.1.1 2
```

10. On SW2 execute the following command to assign the port fe.1.2 to VLAN 2:

```
SW2:> set port vlan fe.1.2 2
```

11. On SW1 execute the following command to allow to port fe.1.1 to receive packages pertaining to VLAN 2:

```
SW1:> set vlan egress 2 fe.1.1 tagged
```

12. On SW1 execute the following command to allow to port fe.1.2 to receive packages pertaining to VLAN 2 (it is important to set the port as untagged because the PCs can't assign tag to packages, so SW1 would not accept them):

```
SW1:> set vlan egress 2 fe.1.2 untagged
```

13. On SW2 execute the following command to allow to port fe.1.1 to receive packages pertaining to VLAN 2:

```
SW2:> set vlan egress 2 fe.1.1 tagged
```

14. On SW2 execute the following command to allow to port fe.1.2 to receive packages pertaining to VLAN 2 (it is important to set the port as untagged because the PCs can't assign tag to packages, so SW1 would not accept them):

```
SW2:> set vlan egress 2 fe.1.2 untagged
```

15. On SW1 execute the following command to enable the policy profile that tags the packages pertainting to VLAN 1:

```
SW1:> set policy profile 1 pvid-status enable pvid 1
```

16. On SW1 execute the following command to set the policy rule that applies the profile to the packages coming from PC2:

```
SW1:> set policy rule 1 ipsourcesocket 192.168.1.22
```

17. On SW1 execute the following command to apply the policy profile on the port fe.1.1:

```
SW1:> set policy port fe.1.1 1
```

18. On PC2 execute the following command to verify if it share with PC3 and to analyze from PC1, with program Wireshark, traffic between the two switch:

```
PC2:> ping 192.168.1.33
```

Execution

The execution of the operation 18 has produced a positive result: using the program Wireshark is effectively possible to observe and analyze all the packages that the two switch are exchanged and the packages coming from PC2 are tagged as pertaining to VLAN 1.

Test Result

Analyzing the frames coming from PC2 is possible to assert that the functionality of Policy Classification is usable on the switches Enterasys.

The test result is positive.

Capitolo 5
Test su Multiple Spanning Tree

Questo capitolo ha come scopo quello di mostrare la metodologia di test applicata a scenari in cui entri in gioco l'algoritmo di Multiple Spanning Tree tra switch Enterasys. I concetti teorici per una migliore comprensione del capitolo sono raccolti nel capitolo 2.

5.1 Test MST istances

Test	Test MST istances
Object	Enterasys SecureStack B2H124-48P (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24; Enterasys SecureStack C2G124-24 (24 ports) Chassis Serial Number: 04400122900B Chassis Firmware Revision: 04.00.24; Enterasys SecureStack C2H124-48 (48 ports) Chassis Serial Number: 05210107900D Chassis Firmware Revision: 04.00.31
Target	To verify the functioning of the creation of MST istance on switches Enterasys entry level.
Designer	Giulia Papini
Executor	Giulia Papini (05/03/2007)
Result	Positive

Object of the experiment

The object of the experiment is composed of three stackable switches Enterasys:

- SW1: Enterasys SecureStack B2

  B2H124-48P (48 ports)

  Chassis Serial Number: 05320111900B

  Chassis Firmware Revision: 03.01.24

- SW2: Enterasys SecureStack C2

  C2G124-24 (24 ports)

  Chassis Serial Number: 04400122900B

  Chassis Firmware Revision: 04.00.24

- SW3: Enterasys SecureStack C2

  C2H124-48 (48 ports)

  Chassis Serial Number: 05210107900D

  Chassis Firmware Revision: 04.00.31

Target of the experiment

To verify the functioning of creation of MST istance on switches Enterasys entry level.

Expected results

The expected results is the creation of two different paths in two different istances blocking the link between SW2 and SW3 for the Spanning Tree ID 22 and blocking the link between SW3 and SW1 for the Spanning Tree ID 33.

Rationales

It is chosen to use a triangular configuration with three switches because it allows to show in a better way how shape a cycle and how the Multiple Spanning Tree resolves it with different paths based on the different priority assigned to the switch in the different instances.

Configuration

The switches have one setting priority for each istance of MST.

For the istance 22 the priority of every switches is the follow:

- SW1: 0;
- SW2: 4096;
- SW3: 8192;

For the istance 33 the priority of every switches is the follow:

- SW1: 8192;
- SW2: 0;
- SW3: 4096;

The istance 22 is mapped to VLAN 2 and istance 33 is mapped to VLAN 3.

All the links are Trunk 1Q.

On all the network devices are configured the Multiple Spanning Tree Protocol and are disabled the level two protocols, GVRP and LACP.

The level three configuration assigned to every switch is the follow: IP address:

- SW1: 192.168.1.1/24;

- SW2: 192.168.1.2/24;
- SW3: 192.168.1.3/24;

Topology

In figure 5.1 is represented the topology used on physical level for the test.

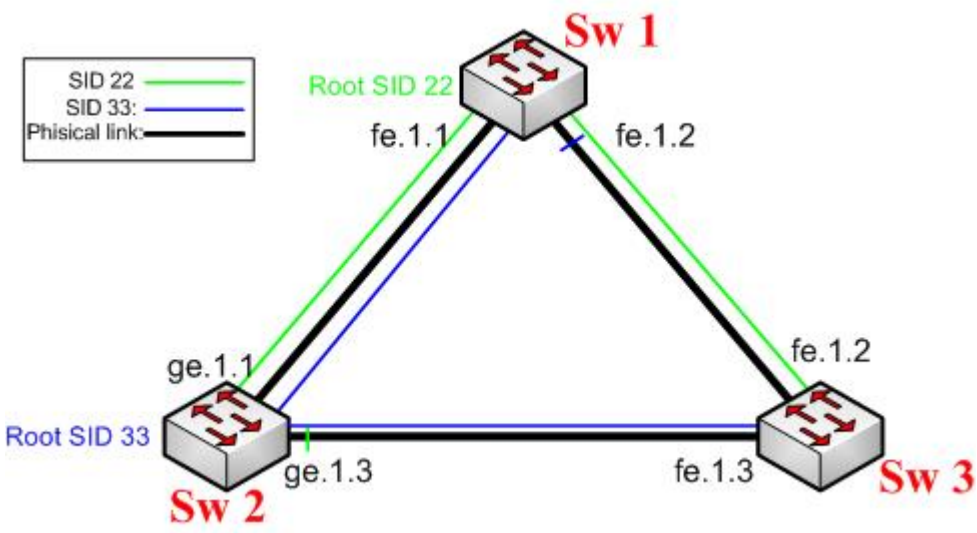

Figura 5.1: Topology of Port Mirroring

Operations

1. On every switch execute the following command to delete the pre-existing settings:

```
SW 1,2,3:> clear config all
```

2. On SW1 execute the following command to set the IP address:

```
SW1:> set ip address 192.168.1.1 mask 255.255.255.0
```

3. On SW2 execute the following command to set the IP address:

```
SW2:> set ip address 192.168.1.2 mask 255.255.255.0
```

4. On SW3 execute the following command to set the IP address:

```
SW3:> set ip address 192.168.1.3 mask 255.255.255.0
```

5. On all switches execute the following command to set the Multiple Spanning Tree Protocol:

```
SW 1,2,3:> set spantree version mstp
```

6. On all switches execute the following command to disable the GVRP Protocol:

```
SW 1,2,3:> set gvrp disable
```

7. On all switches execute the following command to disable the LACP Protocol:

```
SW 1,2,3:> set lacp disable
```

8. On all switches execute the following command to create the VLAN 2:

```
SW 1,2,3:> set vlan create 2
```

9. On all switches execute the following command to create the VLAN 3:

```
SW 1,2,3:> set vlan create 3
```

10. On all switches execute the following command to create the istance 22 of MST:

```
SW 1,2,3:> set spantree msti sid 22 create
```

11. On all switches execute the following command to create the istance 33 of MST:

```
SW 1,2,3:> set spantree msti sid 33 create
```

12. On all switches execute the following command to associate the VLAN 2 to istance 22 of MST:

```
SW 1,2,3:> set spantree mstmap 2 sid 22
```

13. On all switches execute the following command to associate the VLAN 3 to istance 33 of MST:

```
SW 1,2,3:> set spantree mstmap 3 sid 33
```

14. On SW1 execute the following command to assign it the spanning tree priority for the SID 22:

```
SW1:> set spantree priority 0 22
```

15. On SW1 execute the following command to assign it the spanning tree priority for the SID 33:

```
SW1:> set spantree priority 8192 33
```

16. On SW2 execute the following command to assign it the spanning tree priority for the SID 22:

```
SW2:> set spantree priority 4096 22
```

17. On SW2 execute the following command to assign it the spanning tree priority for the SID 33:

```
SW2:> set spantree priority 0 33
```

18. On SW3 execute the following command to assign it the spanning tree priority for the SID 22:

```
SW3:> set spantree priority 8192 22
```

19. On SW3 execute the following command to assign it the spanning tree priority for the SID 33:

```
SW3:> set spantree priority 4096 33
```

20. On SW1 execute the following command to verify the state of the port fe.1.1 for SID 22:

```
SW1:> show spantree stats port fe.1.1 sid 22
```

21. On SW1 execute the following command to verify the state of the port fe.1.1 for SID 33:

```
SW1:> show spantree stats port fe.1.1 sid 33
```

22. On SW1 execute the following command to verify the state of the port fe.1.2 for SID 22:

```
SW1:> show spantree stats port fe.1.2 sid 22
```

23. On SW1 execute the following command to verify the state of the port fe.1.2 for SID 33:

```
SW1:> show spantree stats port fe.1.2 sid 33
```

24. On SW2 execute the following command to verify the state of the port ge.1.1 for SID 22:

```
SW2:> show spantree stats port ge.1.1 sid 22
```

25. On SW2 execute the following command to verify the port ge.1.1 state for SID 33:

```
SW2:> show spantree stats port ge.1.1 sid 33
```

26. On SW2 execute the following command to verify the state of the port ge.1.3 for SID 22:

```
SW2:> show spantree stats port ge.1.3 sid 22
```

27. On SW2 execute the following command to verify the state of the port ge.1.3 for SID 33:

```
SW2:> show spantree stats port ge.1.3 sid 33
```

28. On SW3 execute the following command to verify the state of the port fe.1.2 for SID 22:

```
SW3:> show spantree stats port fe.1.2 sid 22
```

29. On SW3 execute the following command to verify the state of the port fe.1.2 for SID 33:

```
SW3:> show spantree stats port fe.1.2 sid 33
```

30. On SW3 execute the following command to verify the state of the port fe.1.3 for SID 22:

```
SW3:> show spantree stats port fe.1.3 sid 22
```

31. On SW3 execute the following command to verify the state of the port fe.1.3 for SID 33:

```
SW3:> show spantree stats port fe.1.3 sid 33
```

Execution

1. The execution of operations 20 and 21 has produced the following result:

```
SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  22    fe.1.1       Forwarding         Designated    200000      128

  SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  33    fe.1.1       Forwarding         Root          200000      128
```

2. The execution of operations 22 and 23 has produced the following result:

```
SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  22    fe.1.2       Forwarding         Designated    200000      128

  SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  33    fe.1.2       Discarding         Alternate     200000      128
```

3. The execution of operations 24 and 25 has produced the following result:

```
SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  22    ge.1.1       Forwarding         Root          200000      128

  SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  33    ge.1.1       Forwarding         Designated    200000      128
```

4. The execution of operations 26 and 27 has produced the following result:

```
SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  22    ge.1.3       Forwarding         Designated    200000      128

  SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  33    ge.1.3       Forwarding         Designated    200000      128
```

5. The execution of operations 28 and 29 has produced the following result:

```
  SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  22    fe.1.2       Forwarding         Root          200000      128

  SID   Port         State              Role          Cost        Priority
  ---   ----------   ----------------   -----------   --------    --------
  33    fe.1.2       Forwarding         Designated    200000      128
```

6. The execution of operations 28 and 29 has produced the following result:

```
SID   Port         State              Role          Cost        Priority
---   ----------   ----------------   -----------   --------    --------
22    fe.1.3       Forwarding         Root          200000      128

SID   Port         State              Role          Cost        Priority
---   ----------   ----------------   -----------   --------    --------
33    fe.1.3       Discarding         Alternate     200000      128
```

Test Result

Seen the results obtained in point 20-31 of execution of the test is possible to assert that the functioning of creation of 2 MST istance with different priority on switch Enterasys entry level generates two different paths.

The result is positive.

5.2 Test Root Cost in MST Region

Test	Test Root Cost in MST Region
Object	Enterasys SecureStack B2H124-48P (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24; Enterasys SecureStack C2G124-24 (24 ports) Chassis Serial Number: 04400122900B Chassis Firmware Revision: 04.00.24; Enterasys SecureStack C2H124-48 (48 ports) Chassis Serial Number: 05210107900D Chassis Firmware Revision: 04.00.31 Enterasys Matrix N3 serie Platinum (24 ports) Chassis Serial Number: 0001f45c06e9 Chassis Firmware Revision: 05.11.29 Enterasys Matrix N3 serie Platinum (48 ports) Chassis Serial Number: 0001f47f0575 Chassis Firmware Revision: 05.32.06 Enterasys Matrix N3 serie Platinum (24 ports) Chassis Serial Number: 0001f45c06e9 Chassis Firmware Revision: 05.11.29
Target	To analyze the root cost in MST Region on switches Enterasys entry level.
Designer	Giulia Papini
Executor	Giulia Papini (30/03/2007)
Result	Positive

Object of the experiment

The object of the experiment is composed of six stackable switches Enterasys:

- SW1: Enterasys SecureStack B2

  B2H124-48P (48 ports)

  Chassis Serial Number: 05320111900B

Chassis Firmware Revision: 03.01.24

- SW2: Enterasys SecureStack C2

  C2G124-24 (24 ports)

  Chassis Serial Number: 04400122900B

  Chassis Firmware Revision: 04.00.24

- SW3: Enterasys SecureStack C2

  C2H124-48 (48 ports)

  Chassis Serial Number: 05210107900D

  Chassis Firmware Revision: 04.00.31

- SW4: Enterasys Matrix N3 serie Platinum (24 ports)

  Chassis Serial Number: 0001f45c06e9

  Chassis Firmware Revision: 05.11.29

- SW5: Enterasys Matrix N3 serie Platinum (48 ports)

  Chassis Serial Number: 0001f47f0575

  Chassis Firmware Revision: 05.32.06

- SW6: Enterasys Matrix N3 serie Platinum (24 ports)

  Chassis Serial Number: 0001f45c06e9

  Chassis Firmware Revision: 05.11.29

Target of the experiment

To analyze the root cost in MST Region on switches Enterasys entry level.

Expected results

After analizyng the root cost in the topology of this test, the expected result is that, from the switches of the others region, the region appear like one only switch. Moreover the SW 5, though had the same Region Configuration with Region A, result another Region.

Rationales

It is chosen to use a triangular configuration of switches belonging to the same region connected with a switch STP that is connected with two switches of two different regions because it allows to show as the external switch of one region seen it and to verify that a switch with the same regional configuration not directly connected with a region appears like another region.

Configuration

The switches have one setting priority. The default priority is 32768.

The priority of the switch is the follow:

- SW1: 32768;
- SW2: 32768;
- SW3: 4096;
- SW4: 32768;
- SW5: 0;
- SW6: 32768;

Multiple Spanning Tree Protocol is configurated on switches 1, 2, 3, 5, 6, on switch 4 is configured the traditional Spanning Tree Protocol. On all devices are disabled the level two protocols, GVRP and LACP.

The level three configuration assigned to every switch is the follow: IP address:

- SW1: 192.168.1.1/24;
- SW2: 192.168.1.2/24;
- SW3: 192.168.1.3/24;
- SW4: 192.168.1.4/24;
- SW5: 192.168.1.5/24;
- SW6: 192.168.1.6/24;

Topology

In figure 5.2 is represented the topology used on physical level for the test.

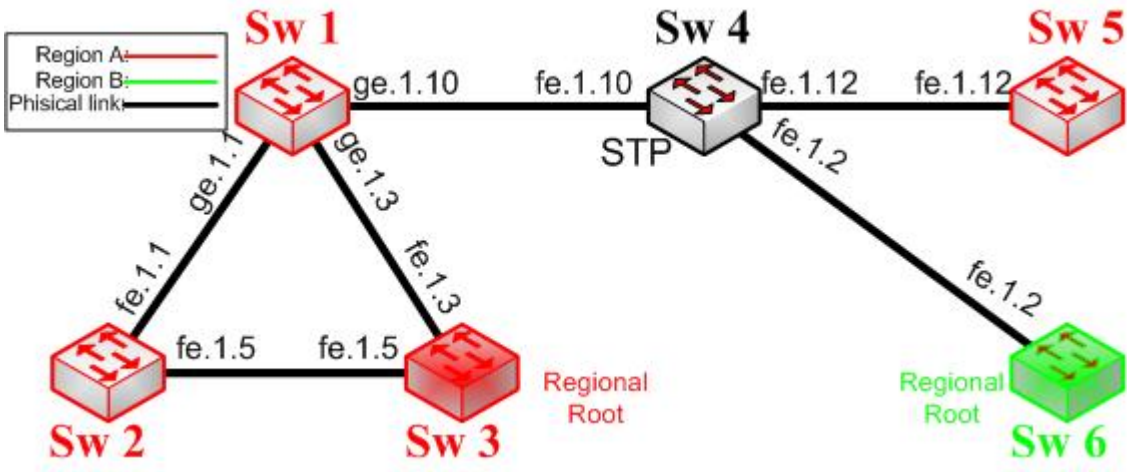

Figura 5.2: Topology of Root Cost in MST Region

Operations

1. On all switches execute the following command to delete the pre-existing settings:

```
SW 1,2,3,4,5,6:> clear config all
```

2. On SW1 execute the following command to set the IP address:

```
SW1:> set ip address 192.168.1.1 mask 255.255.255.0
```

3. On SW2 execute the following command to set the IP address:

```
SW2:> set ip address 192.168.1.2 mask 255.255.255.0
```

4. On SW3 execute the following command to set the IP address:

```
SW3:> set ip address 192.168.1.3 mask 255.255.255.0
```

5. On SW4 execute the following command to set the IP address:

```
SW4:> set ip address 192.168.1.4 mask 255.255.255.0
```

6. On SW5 execute the following command to set the IP address:

```
SW5:> set ip address 192.168.1.5 mask 255.255.255.0
```

7. On SW6 execute the following command to set the IP address:

```
SW6:> set ip address 192.168.1.6 mask 255.255.255.0
```

8. On switches 1, 2, 3, 5, 6 execute the following command to set the Multiple Spanning Tree Protocol:

```
SW 1,2,3,5,6:> set spantree version mstp
```

9. On switch 4 execute the following command to set the Spanning Tree Protocol:

```
SW 4:> set spantree version stpcompatible
```

10. On all switches execute the following command to disable the GVRP Protocol:

```
SW 1,2,3,4,5,6:> set gvrp disable
```

11. On all switches execute the following command to disable the LACP Protocol:

```
SW 1,2,3,4,5,6:> set lacp disable
```

12. On all switches execute the following command to create the VLAN 2:

```
SW 1,2,3,4,5,6:> set vlan create 2
```

13. On switches 1, 2, 3, 5 execute the following command to configure the name of the region:

```
SW 1,2,3,5:> set spantree mstcfgid cfgname regA
```

14. On switch 6 execute the following command to configure the name of the region:

```
SW 6:> set spantree mstcfgid cfgname regB
```

15. On SW3 execute the following command to assign it the spanning tree priority:

```
SW3:> set spantree priority 4096
```

16. On SW5 execute the following command to assign it the spanning tree priority:

```
SW5:> set spantree priority 0
```

17. On SW1 execute the following command to verify the state of the port ge.1.1:

```
SW1:> show spantree stats port ge.1.1
```

18. On SW1 execute the following command to verify the state of the port ge.1.3:

```
SW1:> show spantree stats port ge.1.3
```

19. On SW1 execute the following command to verify the state of the port ge.1.10:

```
SW1:> show spantree stats port ge.1.10
```

20. On SW2 execute the following command to verify the state of the port fe.1.1:

```
SW2:> show spantree stats port fe.1.1
```

21. On SW2 execute the following command to verify the state of the port fe.1.5:

```
SW2:> show spantree stats port fe.1.5
```

22. On SW3 execute the following command to verify the state of the port fe.1.3:

```
SW3:> show spantree stats port fe.1.3
```

23. On SW3 execute the following command to verify the state of the port fe.1.5:

```
SW3:> show spantree stats port fe.1.5
```

24. On SW4 execute the following command to verify the state of the port fe.1.10:

```
SW4:> show spantree stats port fe.1.10
```

25. On SW4 execute the following command to verify the state of the port fe.1.12:

```
SW4:> show spantree stats port fe.1.12
```

26. On SW4 execute the following command to verify the state of the port fe.1.2:

```
SW4:> show spantree stats port fe.1.2
```

27. On SW5 execute the following command to verify the state of the port fe.1.12:

```
SW5:> show spantree stats port fe.1.12
```

28. On SW6 execute the following command to verify the state of the port fe.1.2:

```
SW6:> show spantree stats port fe.1.2
```

29. On all switches execute the following command to see the Designated Root Cost in the result table:

```
SW 1,2,3,4,5,6:> show spantree stats
```

Execution

1. The execution of operations 17, 18 and 19 has produced the following result:

```
SID   Port         State              Role          Cost       Priority
---   ----------   ----------------   -----------   --------   --------
0     ge.1.1       Forwarding         Designated    200000     128

SID   Port         State              Role          Cost       Priority
---   ----------   ----------------   -----------   --------   --------
0     ge.1.3       Forwarding         Root          200000     128

SID   Port         State              Role          Cost       Priority
---   ----------   ----------------   -----------   --------   --------
0     ge.1.10      Forwarding         Designated    200000     128
```

2. The execution of operations 20 and 21 has produced the following result:

```
SID   Port         State              Role          Cost       Priority
---   ----------   ----------------   -----------   --------   --------
0     fe.1.1       Discarding         Alternate     200000     128

SID   Port         State              Role          Cost       Priority
---   ----------   ----------------   -----------   --------   --------
0    fe.1.5        Forwarding         Root          200000     128
```

3. The execution of operations 22 and 23 has produced the following result:

```
SID   Port         State              Role          Cost       Priority
---   ----------   ----------------   -----------   --------   --------
0     fe.1.3       Forwarding         Designated    200000     128

SID   Port         State              Role          Cost       Priority
---   ----------   ----------------   -----------   --------   --------
33    fe.1.5       Forwarding         Designated    200000     128
```

4. The execution of operations 24, 25 and 26 has produced the following result:

```
SID   Port         State              Role          Cost       Priority
---   ----------   ----------------   -----------   --------   --------
0     fe.1.2       Forwarding         Designated    200000     128

SID   Port         State              Role          Cost       Priority
---   ----------   ----------------   -----------   --------   --------
0     fe.1.10      Forwarding         Designated    200000     128

SID   Port         State              Role          Cost       Priority
---   ----------   ----------------   -----------   --------   --------
0     fe.1.12      Forwarding         Designated    200000     128
```

5. The execution of operation 27 has produced the following result:

```
SID   Port         State              Role          Cost        Priority
---   ----------   ----------------   -----------   --------    --------
0     fe.1.12      Forwarding         Designated    200000      128
```

6. The execution of operation 28 has produced the following result:

```
SID   Port         State              Role          Cost        Priority
---   ----------   ----------------   -----------   --------    --------
0     fe.1.2       Forwarding         Designated    200000      128
```

7. From the execution of operation 29 is possible to extract the designated root cost of every switch:

```
SW1: Designated Root Cost  200000
SW2: Designated Root Cost  200000
SW3: Designated Root Cost  0
SW4: Designated Root Cost  200000
SW5: Designated Root Cost  400000
SW6: Designated Root Cost  400000
```

Test Result

Seen the results obtained in point 29 of execution of the test is possible to assert that SW4 see the region A as a single switch (if it were not therefore the designated root cost would have to be 400000 and not 200000) and that the SW5 isn't a member of region A.

The result is positive.

5.3 **Test** BPDU MST

Test	Test BPDU MST
Object	Enterasys SecureStack B2H124-48P (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24; Enterasys SecureStack C2G124-24 (24 ports) Chassis Serial Number: 04400122900B Chassis Firmware Revision: 04.00.24; Enterasys SecureStack C2H124-48 (48 ports) Chassis Serial Number: 05210107900D Chassis Firmware Revision: 04.00.31 Enterasys Matrix N3 serie Platinum (48 ports) Chassis Serial Number: 0001f47f0575 Chassis Firmware Revision: 05.32.06
Target	To analyze the BPDU MST on switches Enterasys entry level.
Designer	Giulia Papini
Executor	Giulia Papini (02/04/2007)
Result	Positive

Object of the experiment

The object of the experiment is composed of 4 stackable switches Enterasys:

- SW1: Enterasys SecureStack B2

  B2H124-48P (48 ports)

  Chassis Serial Number: 05320111900B

  Chassis Firmware Revision: 03.01.24

- SW2: Enterasys SecureStack C2

  C2G124-24 (24 ports)

Chassis Serial Number: 04400122900B

Chassis Firmware Revision: 04.00.24

- SW3: Enterasys SecureStack C2

  C2H124-48 (48 ports)

  Chassis Serial Number: 05210107900D

  Chassis Firmware Revision: 04.00.31

- SW4: Enterasys Matrix N3 serie Platinum (48 ports)

  Chassis Serial Number: 0001f47f0575

  Chassis Firmware Revision: 05.32.06

Target of the experiment

To analyze the BPDU MST on switches Enterasys entry level.

Expected results

The expected result is the possibility to analyze the BPDU MST internal to the region and between different region. Moreover the packages sniffed from the observation point A will be of type traditional Spanning Tree, while the packages sniffed from the observation point B will be comprex packages of type Multiple Spanning Tree.

Rationales

It is chosen to use a triangular configuration of switches belonging to the same region connected with a switch STP because it allow to sniff with the program Wireshark installed on a pc connected with an hub before in point A and then in point B.

Tools

The test platform is composed of a personal computer and an Hub:

- PC1: PC Notebook HP-DV2172EA 1.66 GHz/980 MHz with 1 Gb of RAM;
- Hub: OfficeConnect Hub 3com 8/TPO 10BaseT (8)RJ-45 with speed supported: 10 Mbps

Moreover it's used the program 'Wireshark' that 'sniffs' the packages, it captures and analyzes them, showing their information and fields.

Configuration

The switches have one setting priority. The default priority is 32768.

For the istance 0 the priority of every switch it is the follow:

- SW1: 32768;
- SW2: 32768;
- SW3: 32768;
- SW4: 12288;

For the istance 22 the priority of every switch it is the follow:

- SW1: 0;
- SW2: 4096;
- SW3: 8192;

On SW3 is configured the Traditional Spanning Tree Protocol, so it cannot have more istances.

For the istance 33 the priority of every switch it is the follow:

- SW1: 8192;
- SW2: 4096;
- SW3: 0;

On SW3 is configured the Traditional Spanning Tree Protocol, so it cannot have more istances.

The istance 22 is mapped to VLAN 2 and istance 33 is mapped to VLAN 3. All the link are Trunk 1Q. On the switches 1, 2, 3 is configured the Multiple Spanning Tree Protocol, on switch 4 is configured the traditional Spanning Tree Protocol. On all devices the level two protocols, GVRP and LACP, are disabled.

The level three configuration assigned to every switch is the follow: IP address:

- SW1: 192.168.1.1/24;
- SW2: 192.168.1.2/24;
- SW3: 192.168.1.3/24;
- SW4: 192.168.1.4/24;

Topology

In figure 5.3 is represented the topology used on physical level for the test.

Operations

1. On all switches execute the following command to delete the pre-existing settings:

```
SW 1,2,3,4:> clear config all
```

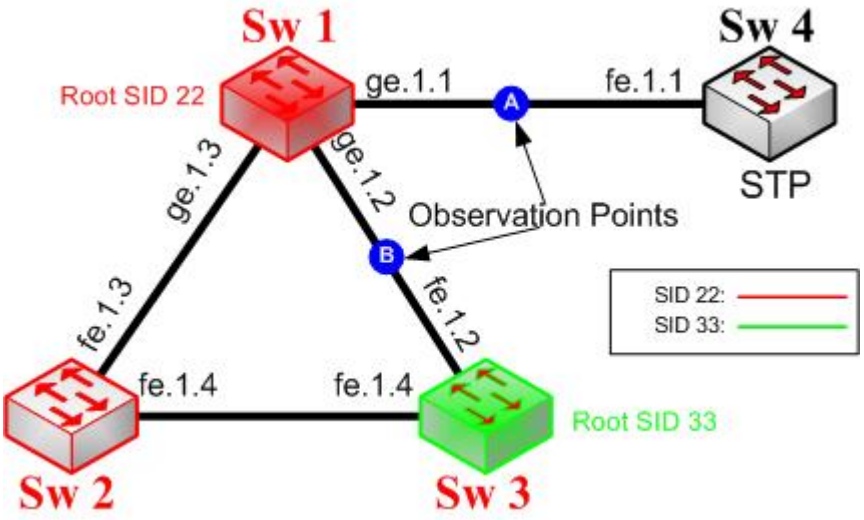

Figura 5.3: Topology BPDU MST

2. On SW1 execute the following command to set the IP address:

```
SW1:> set ip address 192.168.1.1 mask 255.255.255.0
```

3. On SW2 execute the following command to set the IP address:

```
SW2:> set ip address 192.168.1.2 mask 255.255.255.0
```

4. On SW3 execute the following command to set the IP address:

```
SW3:> set ip address 192.168.1.3 mask 255.255.255.0
```

5. On SW4 execute the following command to set the IP address:

```
SW4:> set ip address 192.168.1.4 mask 255.255.255.0
```

6. On switches 1, 2, 3 execute the following command to set the Multiple Spanning Tree Protocol:

```
SW 1,2,3:> set spantree version mstp
```

7. On switch 4 execute the following command to set the Spanning Tree Protocol:

```
SW 4:> set spantree version stpcompatible
```

8. On all switches execute the following command to disable the GVRP Protocol:

```
SW 1,2,3,4:> set gvrp disable
```

9. On all switches execute the following command to disable the LACP Protocol:

```
SW 1,2,3,4:> set lacp disable
```

10. On all switches execute the following command to create the VLAN 2:

```
SW 1,2,3,4:> set vlan create 2
```

11. On all switches execute the following command to create the VLAN 3:

```
SW 1,2,3,4:> set vlan create 3
```

12. On switches 1, 2, 3 execute the following command to configure the name of the region:

```
SW 1,2,3:> set spantree mstcfgid cfgname pippo
```

13. On switches 1, 2, 3 execute the following command to create the istance 22 of MST:

```
SW 1,2,3:> set spantree msti sid 22 create
```

14. On switches 1, 2, 3 execute the following command to create the istance 33 of MST:

```
SW 1,2,3:> set spantree msti sid 33 create
```

15. On switches 1, 2, 3 execute the following command to associate the VLAN 2 to istance 22 of MST:

```
SW 1,2,3:> set spantree mstmap 2 sid 22
```

16. On switches 1, 2, 3 execute the following command to associate the VLAN 3 to istance 33 of MST:

```
SW 1,2,3:> set spantree mstmap 3 sid 33
```

17. On SW1 execute the following command to assign it the spanning tree priority for the SID 22:

```
SW1:> set spantree priority 0 22
```

18. On SW1 execute the following command to assign it the spanning tree priority for the SID 33:

```
SW1:> set spantree priority 8192 33
```

19. On SW2 execute the following command to assign it the spanning tree priority for the SID 22:

```
SW2:> set spantree priority 4096 22
```

20. On SW2 execute the following command to assign it the spanning tree priority for the SID 33:

```
SW2:> set spantree priority 4096 33
```

21. On SW3 execute the following command to assign it the spanning tree priority for the SID 22:

```
SW3:> set spantree priority 8192 22
```

22. On SW3 execute the following command to assign it the spanning tree priority for the SID 33:

```
SW3:> set spantree priority 0 33
```

23. On SW4 execute the following command to assign it the spanning tree priority:

```
SW4:> set spantree priority 12288
```

24. On SW1 execute the following command to assign the port fe.1.1 to VLAN 2:

```
SW1:> set port vlan fe.1.1 2
```

25. On SW1 execute the following command to assign the port fe.1.2 to VLAN 2:

```
SW1:> set port vlan fe.1.2 2
```

26. On SW1 execute the following command to assign the port fe.1.3 to VLAN 2:

```
SW1:> set port vlan fe.1.3 2
```

27. On SW2 execute the following command to assign the port ge.1.3 to VLAN 2:

```
SW2:> set port vlan ge.1.3 2
```

28. On SW2 execute the following command to assign the port ge.1.4 to VLAN 2:

```
SW2:> set port vlan ge.1.4 2
```

29. On SW3 execute the following command to assign the port fe.1.2 to VLAN 2:

```
SW3:> set port vlan fe.1.2 2
```

30. On SW3 execute the following command to assign the port fe.1.4 to VLAN 2:

```
SW3:> set port vlan fe.1.4 2
```

31. On SW4 execute the following command to assign the port fe.1.1 to VLAN 2:

```
SW4:> set port vlan fe.1.1 2
```

32. On SW1 execute the following command to allow to the port fe.1.1 to receive packages from VLAN 2:

```
SW1:> set vlan egress 2 fe.1.1 tagged
```

33. On SW1 execute the following command to allow to the port fe.1.2 to receive packages from VLAN 2:

```
SW1:> set vlan egress 2 fe.1.2 tagged
```

34. On SW1 execute the following command to allow to the port fe.1.3 to receive packages from VLAN 3:

```
SW1:> set vlan egress 2 fe.1.3 untagged
```

35. On SW2 execute the following command to allow to the port ge.1.3 to receive packages from VLAN 2:

```
SW2:> set vlan egress 2 ge.1.3 tagged
```

36. On SW2 execute the following command to allow to the port ge.1.4 to receive packages from VLAN 2:

```
SW2:> set vlan egress 2 ge.1.4 tagged
```

37. On SW3 execute the following command to allow to the port fe.1.2 to receive packages from VLAN 2:

```
SW3:> set vlan egress 2 fe.1.2 tagged
```

38. On SW3 execute the following command to allow to the port fe.1.4 to receive packages from VLAN 2:

```
SW3:> set vlan egress 2 fe.1.4 tagged
```

39. On SW4 execute the following command to allow to the port fe.1.1 to receive packages from VLAN 2:

```
SW4:> set vlan egress 2 fe.1.1 tagged
```

40. Connect the hub with the PC in observation point A and sniff the packages to analyze one of them in detail.

41. Connect the hub with the PC in observation point B and sniff the packages to analyze one of them in detail.

Execution

1. The execution of the operation 40 has produced the possibility of analyze information and fields of one of the sniffed packages:

```
Frame 13 (60 bytes on wire, 60 bytes captured)
    Arrival Time: Mar 28, 2007 12:08:44.354502000
    [Time delta from previous captured frame: 2.000476000 seconds]
    [Time delta from previous displayed frame: 2.000476000 seconds]
    [Time since reference or first frame: 22.004896000 seconds]
    Frame Number: 13
    Frame Length: 60 bytes
    Capture Length: 60 bytes
    [Frame is marked: False]
    [Protocols in frame: eth:llc:stp]
    [Coloring Rule Name: Broadcast]
    [Coloring Rule String: eth[0] & 1]
IEEE 802.3 Ethernet
    Destination: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
        Address: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
        .... ...1 .... .... .... .... = IG bit: Group address (multicast/broadcast)
        .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
    Source: Enterasy_0a:fb:3d (00:11:88:0a:fb:3d)
        Address: Enterasy_0a:fb:3d (00:11:88:0a:fb:3d)
        .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
        .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
    Length: 38
    Trailer: 9224922492249224
Logical-Link Control
    DSAP: Spanning Tree BPDU (0x42)
    IG Bit: Individual
    SSAP: Spanning Tree BPDU (0x42)
    CR Bit: Command
    Control field: U, func=UI (0x03)
        000. 00.. = Command: Unnumbered Information (0x00)
        .... ..11 = Frame type: Unnumbered frame (0x03)
Spanning Tree Protocol
    Protocol Identifier: Spanning Tree Protocol (0x0000)
    Protocol Version Identifier: Spanning Tree (0)
    BPDU Type: Configuration (0x00)
    BPDU flags: 0x00
        0... .... = Topology Change Acknowledgment: No
        .... ...0 = Topology Change: No
    Root Identifier: 12288 / 00:11:88:16:04:2f
```

```
    Root Path Cost: 0
    Bridge Identifier: 12288 / 00:11:88:16:04:2f
    Port identifier: 0x8040
    Message Age: 0
    Max Age: 20
    Hello Time: 2
    Forward Delay: 15
```

2. The execution of the operation 41 has produced the possibility of analyze information and fields of one of the sniffed packages:

```
      Frame 11 (151 bytes on wire, 151 bytes captured)
    Arrival Time: Mar 28, 2007 12:21:18.388503000
    [Time delta from previous captured frame: 1.845332000 seconds]
    [Time delta from previous displayed frame: 1.845332000 seconds]
    [Time since reference or first frame: 7.001571000 seconds]
    Frame Number: 11
    Frame Length: 151 bytes
    Capture Length: 151 bytes
    [Frame is marked: False]
    [Protocols in frame: eth:llc:stp]
    [Coloring Rule Name: Broadcast]
    [Coloring Rule String: eth[0] & 1]
IEEE 802.3 Ethernet
    Destination: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
        Address: Spanning-tree-(for-bridges)_00 (01:80:c2:00:00:00)
        .... ...1 .... .... .... .... = IG bit: Group address (multicast/broadcast)
        .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
    Source: Enterasy_0b:3e:84 (00:11:88:0b:3e:84)
        Address: Enterasy_0b:3e:84 (00:11:88:0b:3e:84)
        .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
        .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
    Length: 137
Logical-Link Control
    DSAP: Spanning Tree BPDU (0x42)
    IG Bit: Individual
    SSAP: Spanning Tree BPDU (0x42)
    CR Bit: Command
    Control field: U, func=UI (0x03)
        000. 00.. = Command: Unnumbered Information (0x00)
        .... ..11 = Frame type: Unnumbered frame (0x03)
Spanning Tree Protocol
```

```
Protocol Identifier: Spanning Tree Protocol (0x0000)
Protocol Version Identifier: Multiple Spanning Tree (3)
BPDU Type: Rapid/Multiple Spanning Tree (0x02)
BPDU flags: 0x04 (Port Role: Alternate or Backup)
    0... .... = Topology Change Acknowledgment: No
    .0.. .... = Agreement: No
    ..0. .... = Forwarding: No
    ...0 .... = Learning: No
    .... 01.. = Port Role: Alternate or Backup (1)
    .... ..0. = Proposal: No
    .... ...0 = Topology Change: No
Root Identifier: 12288 / 00:11:88:16:04:2f
Root Path Cost: 200000
Bridge Identifier: 32768 / 00:01:f4:5c:3f:20
Port identifier: 0x8041
Message Age: 1
Max Age: 20
Hello Time: 2
Forward Delay: 15
Version 1 Length: 0
Version 3 Length: 96
MST Extension
    MST Config ID format selector: 0
    MST Config name: pippo
    MST Config revision: 0
    MST Config digest: D2C0D5E7B542EA9E717F368102FEF3EB
    CIST Internal Root Path Cost: 400000
    CIST Bridge Identifier: 32768 / 00:01:f4:5c:06:eb
    CIST Remaining hops: 18
    MSTID 22, Regional Root Identifier 0 / 00:01:f4:5c:3f:20
        MSTI flags: 0x04 (Port Role: Alternate or Backup)
            0... .... = Topology Change Acknowledgment: No
            .0.. .... = Agreement: No
            ..0. .... = Forwarding: No
            ...0 .... = Learning: No
            .... 01.. = Port Role: Alternate or Backup (1)
            .... ..0. = Proposal: No
            .... ...0 = Topology Change: No
        MSTID 22, priority 0 Root Identifier 00:01:f4:5c:3f:20
        Internal root path cost: 400000
        Bridge Identifier Priority: 2
        Port identifier priority: 8
        Remaining hops: 18
```

```
MSTID 33, Regional Root Identifier 0 / 00:01:f4:5c:06:eb
    MSTI flags: 0xfe (Master, Agreement, Forwarding, Learning, Port Role: Designated, Propos
        1... .... = Topology Change Acknowledgment: Yes
        .1.. .... = Agreement: Yes
        ..1. .... = Forwarding: Yes
        ...1 .... = Learning: Yes
        .... 11.. = Port Role: Designated (3)
        .... ..1. = Proposal: Yes
        .... ...0 = Topology Change: No
    MSTID 33, priority 0 Root Identifier 00:01:f4:5c:06:eb
    Internal root path cost: 0
    Bridge Identifier Priority: 0
    Port identifier priority: 8
    Remaining hops: 20
```

Test Result

Seen the results obtained in points 40 and 41 of the execution of the test is possible to assert that the packages sniffed from the observation point A are of type Traditional Spanning Tree, while the packages sniffed from the observation point B are comprex packages of type Multiple Spanning Tree.

The test result is positive.

5.4 Test Topology Change in MST 1

Test	Test Topology Change in MST 1
Object	Enterasys SecureStack B2H124-48P (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24; Enterasys SecureStack C2G124-24 (24 ports) Chassis Serial Number: 04400122900B Chassis Firmware Revision: 04.00.24;
Target	To understand the behavior of an MST switch when it's connected to a traditional ST switch
Designer	Giulia Papini
Executor	Giulia Papini (04/04/2007)
Result	Positive

Object of the experiment

The object of the experiment is composed of 2 stackable switches Enterasys:

- SW1: Enterasys SecureStack B2

  B2H124-48P (48 ports)

  Chassis Serial Number: 05320111900B

  Chassis Firmware Revision: 03.01.24

- SW2: Enterasys SecureStack C2

  C2G124-24 (24 ports)

  Chassis Serial Number: 04400122900B

  Chassis Firmware Revision: 04.00.24

Target of the experiment

To understand the behavior of an MST switch when it's connected to a traditional ST switch.

Expected results

The expected result is that the SW1, in regular conditions send MSTP BPDU and when it's connected to SW2, configured with traditional Spanning Tree, it starts to send normal STP BPDU.

Rationales

It is chosen to use a linear configuration of a switch MSTP that will be connected to a switch STP, because it allows to sniff the traffic with the program Wireshark installed on a pc connected with an hub to the link between SW1 and SW2.

Tools

The test platform is composed of a personal computer and an Hub:

- PC1: PC Notebook HP-DV2172EA 1.66 GHz/980 MHz with 1 Gb of RAM;
- Hub: OfficeConnect Hub 3com 8/TPO 10BaseT (8)RJ-45 with speed supported: 10 Mbps

Moreover it's used the program 'Wireshark' that 'sniffs' the packages, it captures and analyzes them, showing their information and fields.

Configuration

The switches have one setting priority. The default priority is 32768.

The priority of switch 1 is the follow:

- Istance 0: 0;
- Istance 22: 0;

The priority of switch 2 is the default one.

The istance 22 is mapped to VLAN 2. On switch 1 it's configured the Multiple Spanning Tree Protocol, on switch 2 it's configured the traditional Spanning Tree Protocol. On both devices the level two protocols, GVRP and LACP, are disabled.

The level three configuration assigned to every switch is the follow: IP address:

- SW1: 192.168.1.1/24;
- SW2: 192.168.1.2/24;

Topology

In figure 5.4 is represented the topology used on physical level for the test.

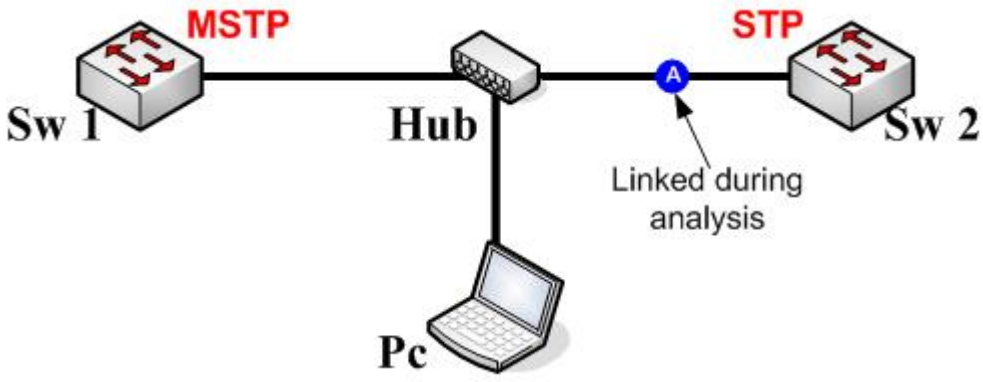

Figura 5.4: Topology Change in MST 1

Operations

1. On both switches execute the following command to delete the pre-existing settings:

```
SW 1,2:> clear config all
```

2. On SW1 execute the following command to set the IP address:

```
SW1:> set ip address 192.168.1.1 mask 255.255.255.0
```

3. On SW2 execute the following command to set the IP address:

```
SW2:> set ip address 192.168.1.2 mask 255.255.255.0
```

4. On SW1 execute the following command to set the Multiple Spanning Tree Protocol:

```
SW 1:> set spantree version mstp
```

5. On SW2 execute the following command to set the Spanning Tree Protocol:

```
SW 2:> set spantree version stpcompatible
```

6. On both switches execute the following command to disable the GVRP Protocol:

```
SW 1,2:> set gvrp disable
```

7. On both switches execute the following command to disable the LACP Protocol:

```
SW 1,2:> set lacp disable
```

8. On both switches execute the following command to create the VLAN 2:

```
SW 1,2:> set vlan create 2
```

9. On SW1 execute the following command to configure the name of the region:

```
SW 1:> set spantree mstcfgid cfgname pippo
```

10. On SW1 execute the following command to create the istance 22 of MST:

```
SW 1:> set spantree msti sid 22 create
```

11. On SW1 execute the following command to associate the VLAN 2 to istance 22 of MST:

```
SW 1:> set spantree mstmap 2 sid 22
```

12. On SW1 execute the following command to assign it the spanning tree priority:

```
SW1:> set spantree priority 0
```

13. On SW1 execute the following command to assign it the spanning tree priority for the SID 22:

```
SW1:> set spantree priority 0 22
```

14. While capturing the packages with the program Wireshark, connect SW2 to the Hub.

Execution

The execution of the operation 14 has produced the following result:

```
10 15.998985  Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP    MST. Root = 0/00:01:f4:5c:3f:20
Cost = 0  Port = 0x8001
11 16.258371  Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP    Conf. Root = 4096/00:11:88:1a:19:e1
  Cost = 0  Port = 0x8001
12 16.280310  Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP    Conf. Root = 4096/00:11:88:1a:19:e1
Cost = 0  Port = 0x8001
13 17.998685  Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP    Conf. Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
14 18.029764  Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP    Topology Change Notification
15 19.998619  Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP    Conf. Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
```

Test Result

Seen the results obtained in point 14 of execution of the test is possible to assert that SW1, in regular conditions send MST BPDU's and when it's connected to SW2, configured with traditional Spanning Tree, it starts to send normal SPT BPDU's.

The test result is positive.

5.5 Test Topology Change in MST 2

Test	Test Topology Change in MST 2
Object	Enterasys SecureStack B2H124-48P (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24; Enterasys SecureStack C2G124-24 (24 ports) Chassis Serial Number: 04400122900B Chassis Firmware Revision: 04.00.24;
Target	To understand the behavior of an MST switch when it's connected to another MST switch of the same region
Designer	Giulia Papini
Executor	Giulia Papini (05/04/2007)
Result	Positive

Object of the experiment

The object of the experiment is composed of 2 stackable switches Enterasys:

- SW1: Enterasys SecureStack B2

  B2H124-48P (48 ports)

  Chassis Serial Number: 05320111900B

  Chassis Firmware Revision: 03.01.24

- SW2: Enterasys SecureStack C2

  C2G124-24 (24 ports)

  Chassis Serial Number: 04400122900B

  Chassis Firmware Revision: 04.00.24

Target of the experiment

To understand the behavior of an MST switch when it's connected to another MST switch of the same region.

Expected results

The expected result is that the SW1, in regular conditions send MST BPDU's. When it's connected to SW2, it send some packages to put themself in agreement and then, since it's the root, it restart to send MST BPDU's.

Rationales

It is chosen to use a linear configuration of a switch MSTP that will be connected with a switch MSTP of the same region because it allow to sniff the traffic with the program Wireshark installed on PC1 connected to the hub to the link between SW1 and SW2.

Tools

The test platform is composed of a personal computer and an Hub:

- PC1: PC Notebook HP-DV2172EA 1.66 GHz/980 MHz with 1 Gb of RAM;
- Hub: OfficeConnect Hub 3com 8/TPO 10BaseT (8)RJ-45 with speed supported: 10 Mbps

Moreover it's used the program 'Wireshark' that 'sniffs' the packages, it captures and analyzes them, showing their information and fields.

Configuration

The switches have one setting priority. The default priority is 32768.

The priority of switch 1 is the follow:

- Istance 0: 0;
- Istance 22: 0;

The priority of switch 2 is the follow:

- Istance 0: 4096;
- Istance 22: 4096;

The istance 22 is mapped to VLAN 2. On both the switches it's configured the Multiple Spanning Tree Protocol and the level two protocols, GVRP and LACP, are disabled.

The level three configuration assigned to every switch is the follow: IP address:

- SW1: 192.168.1.1/24;
- SW2: 192.168.1.2/24;

Topology

In figure 5.5 is represented the topology used on physical level for the test.

Operations

1. On both switches execute the following command to delete the pre-existing settings:

```
SW 1,2:> clear config all
```

2. On SW1 execute the following command to set the IP address:

```
SW1:> set ip address 192.168.1.1 mask 255.255.255.0
```

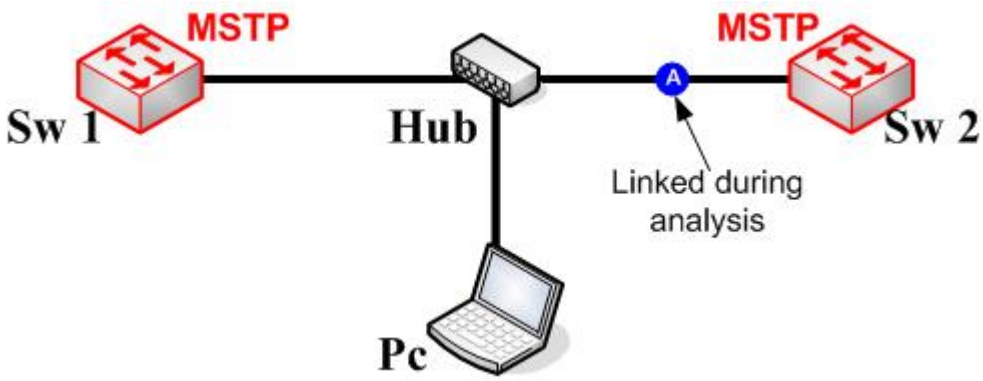

Figura 5.5: Topology Change in MST 2

3. On SW2 execute the following command to set the IP address:

```
SW2:> set ip address 192.168.1.2 mask 255.255.255.0
```

4. On both switches execute the following command to set the Multiple Spanning Tree Protocol:

```
SW 1,2:> set spantree version mstp
```

5. On both switches execute the following command to disable the GVRP Protocol:

```
SW 1,2:> set gvrp disable
```

6. On both switches execute the following command to disable the LACP Protocol:

```
SW 1,2:> set lacp disable
```

7. On both switches execute the following command to create the VLAN 2:

```
SW 1,2:> set vlan create 2
```

8. On both switches execute the following command to configure the name of the region:

```
SW 1,2:> set spantree mstcfgid cfgname pippo
```

9. On both switches execute the following command to create the istance 22 of MST:

```
SW 1,2:> set spantree msti sid 22 create
```

10. On both switches execute the following command to associate the VLAN 2 to istance 22 of MST:

```
SW 1,2:> set spantree mstmap 2 sid 22
```

11. On SW1 execute the following command to assign it the spanning tree priority:

```
SW1:> set spantree priority 0
```

12. On SW1 execute the following command to assign it the spanning tree priority for the SID 22:

```
SW1:> set spantree priority 0 22
```

13. On SW2 execute the following command to assign it the spanning tree priority:

```
SW2:> set spantree priority 4096
```

14. On SW2 execute the following command to assign it the spanning tree priority for the SID 22:

```
SW2:> set spantree priority 4096 22
```

15. While capturing the packages with the program Wireshark, connect SW2 to the Hub.

Execution

The execution of the operation 15 has produced the following result:

```
7 11.999686  Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP   MST. Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
8 12.581985  Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP   MST. Root = 4096/00:11:88:1a:19:e1
  Cost = 0  Port = 0x8001
9 12.583026  Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP   MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
10 12.583612 Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP   MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
11 12.584198 Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP   MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
12 12.585820 Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP   MST. Root = 4096/00:11:88:1a:19:e1
  Cost = 0  Port = 0x8001
13 12.588711 Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP   MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
14 12.590059 Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP   MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
15 12.591386 Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP   MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
16 12.593784 Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP   MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
17 13.999098 Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP   MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
18 14.557517 Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP   MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
19 15.999039 Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP   MST. Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
```

Test Result

Seen the results obtained in point 15 of execution of the test is possible to assert that the SW1, in regular conditions send MST BPDU's. When it's connected to SW2, it send some packages to put themself in agreement and then, since it's the root, it restart to send MST BPDU's.

The test result is positive.

5.6 Test Topology Change in MST 3

Test	Test Topology Change in MST 3
Object	Enterasys SecureStack B2H124-48P (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24; Enterasys SecureStack C2G124-24 (24 ports) Chassis Serial Number: 04400122900B Chassis Firmware Revision: 04.00.24;
Target	To understand the behavior of an MST switch when it's connected to an MST switch of another region
Designer	Giulia Papini
Executor	Giulia Papini (05/04/2007)
Result	Positive

Object of the experiment

The object of the experiment is composed of 2 stackable switches Enterasys:

- SW1: Enterasys SecureStack B2

  B2H124-48P (48 ports)

  Chassis Serial Number: 05320111900B

  Chassis Firmware Revision: 03.01.24

- SW2: Enterasys SecureStack C2

  C2G124-24 (24 ports)

  Chassis Serial Number: 04400122900B

  Chassis Firmware Revision: 04.00.24

Target of the experiment

To understand the behavior of an MST switch when it's connected to an MST switch of another region.

Expected results

The expected result is that the SW1, in regular conditions send MST BPDU's. When it' connected with SW2, it send some packages to put themself in agreement and then, since it's the root of CST, it restart to send MST BPDU's.

Rationales

It is chosen to use a linear configuration of a switch MSTP that will be connected with a switch MSTP of another region because it allow to sniff the traffic with the program Wireshark on PC1 connected to the hub to the link between SW1 and SW2.

Tools

The test platform is composed of a personal computer and an Hub:

- PC1: PC Notebook HP-DV2172EA 1.66 GHz/980 MHz with 1 Gb of RAM;
- Hub: OfficeConnect Hub 3com 8/TPO 10BaseT (8)RJ-45 with speed supported: 10 Mbps

Moreover it's used the program 'Wireshark' that 'sniffs' the packages, it captures and analyzes them, showing their information and fields.

Configuration

The switches have one setting priority. The default priority is 32768.

The priority of switch 1 is the follow:

- Istance 0: 0;
- Istance 22: 0;

The priority of switch 2 is the follow:

- Istance 0: 4096;
- Istance 22: 4096;

The istance 22 is mapped to VLAN 2. On both the switches is configured the Multiple Spanning Tree Protocol, the level two protocols, GVRP and LACP, are disabled.

The level three configuration assigned to every switch is the follow: IP address:

- SW1: 192.168.1.1/24;
- SW2: 192.168.1.2/24;

Topology

In figure 5.6 is represented the topology used on physical level for the test.

Operations

1. On both switches execute the following command to delete the pre-existing settings:

```
SW 1,2:> clear config all
```

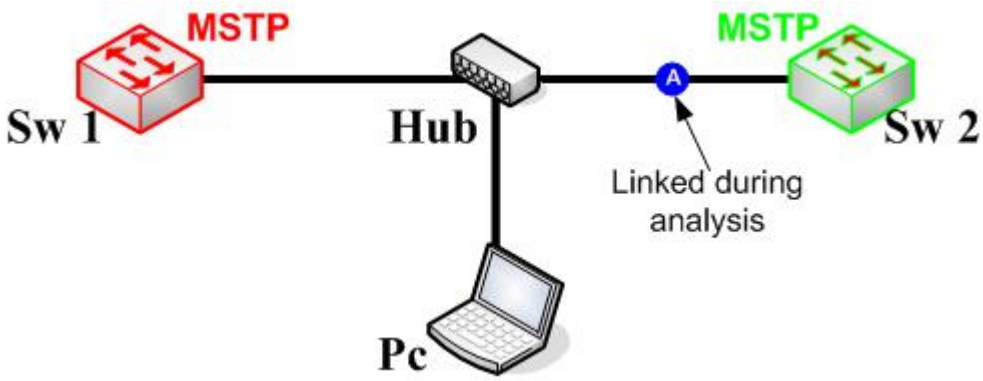

Figura 5.6: Topology Change in MST 3

2. On SW1 execute the following command to set the IP address:

```
SW1:> set ip address 192.168.1.1 mask 255.255.255.0
```

3. On SW2 execute the following command to set the IP address:

```
SW2:> set ip address 192.168.1.2 mask 255.255.255.0
```

4. On both switches execute the following command to set the Multiple Spanning Tree Protocol:

```
SW 1,2:> set spantree version mstp
```

5. On both switches execute the following command to disable the GVRP Protocol:

```
SW 1,2:> set gvrp disable
```

6. On both switches execute the following command to disable the LACP Protocol:

```
SW 1,2:> set lacp disable
```

7. On both switches execute the following command to create the VLAN 2:

```
SW 1,2:> set vlan create 2
```

8. On SW1 execute the following command to configure the name of the region:

```
SW 1:> set spantree mstcfgid cfgname pippo
```

9. On SW2 execute the following command to configure the name of the region:

```
SW 2:> set spantree mstcfgid cfgname pluto
```

10. On both switches execute the following command to create the istance 22 of MST:

```
SW 1,2:> set spantree msti sid 22 create
```

11. On both switches execute the following command to associate the VLAN 2 to istance 22 of MST:

```
SW 1,2:> set spantree mstmap 2 sid 22
```

12. On SW1 execute the following command to assign it the spanning tree priority:

```
SW1:> set spantree priority 0
```

13. On SW1 execute the following command to assign it the spanning tree priority for the SID 22:

```
SW1:> set spantree priority 4096 22
```

14. On SW2 execute the following command to assign it the spanning tree priority:

```
SW2:> set spantree priority 4096
```

15. On SW2 execute the following command to assign it the spanning tree priority for the SID 22:

```
SW2:> set spantree priority 0 22
```

16. While capturing the packages with the program Wireshark, connect SW2 to the Hub.

Execution

The execution of the operation 16 has produced the following result:

```
5 8.000057   Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP  MST. Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
6 8.687610   Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP  MST. Root = 4096/00:11:88:1a:19:e1
  Cost = 0  Port = 0x8001
7 8.688682   Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP  MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
8 8.689268   Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP  MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
9 8.690137   Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP  MST. Root = 4096/00:11:88:1a:19:e1
  Cost = 0  Port = 0x8001
10 8.690711  Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP  MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
11 8.693737  Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP  MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 2000000  Port = 0x8001
12 10.000602 Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP  MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
13 10.558358 Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP  MST. TC + Root = 0/00:01:f4:5c:3f:20
  Cost = 2000000  Port = 0x8001
14 12.000609 Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP  MST. Root = 0/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
```

Test Result

Seen the results obtained in point 16 of execution of the test is possible to assert that the SW1, in regular conditions send MST BPDU's. When it's connected to SW2, it send some packages to put themself in agreement and then, since it's the root of istance 0, it restart to send MST BPDU's.

The test result is positive.

5.7 Test MSTP in heterogeneous environment

Test	Test MSTP in heterogeneous environment
Object	Enterasys SecureStack B2H124-48P (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24; Enterasys SecureStack C2G124-24 (24 ports) Chassis Serial Number: 04400122900B Chassis Firmware Revision: 04.00.24; Enterasys SecureStack C2H124-48 (48 ports) Chassis Serial Number: 05210107900D Chassis Firmware Revision: 04.00.31
Target	To analyze the behavior of MSTP in heterogeneous environment on switches Enterasys entry level.
Designer	Giulia Papini
Executor	Giulia Papini (02/04/2007)
Result	Positive

Object of the experiment

The object of the experiment is composed of 3 stackable switches Enterasys:

- SW1: Enterasys SecureStack B2

  B2H124-48P (48 ports)

  Chassis Serial Number: 05320111900B

  Chassis Firmware Revision: 03.01.24

- SW2: Enterasys SecureStack C2

  C2G124-24 (24 ports)

  Chassis Serial Number: 04400122900B

  Chassis Firmware Revision: 04.00.24

- SW3: Enterasys SecureStack C2

  C2H124-48 (48 ports)

  Chassis Serial Number: 05210107900D

  Chassis Firmware Revision: 04.00.31

Target of the experiment

To analyze the behavior of MSTP in heterogeneous environment on switches Enterasys entry level.

Expected results

The expected result is that SW1 will sends MST BPDU's when it's alone in the network; when it's connected to SW2, it continues sending MST BPDU's. When it's connected with SW3 that is configured with traditional STP, it sends ST BPDU's.

Rationales

It is chosen to use a star configuration of three switch connected by an hub because it allow to sniff the whole traffic with the program Wireshark on PC1 and so is possible analyze the behavior of SW1.

Tools

The test platform is composed of a personal computer and an Hub:

- PC1: PC Notebook HP-DV2172EA 1.66 GHz/980 MHz with 1 Gb of RAM;
- Hub: OfficeConnect Hub 3com 8/TPO 10BaseT (8)RJ-45 with speed supported: 10 Mbps

Moreover it's used the program 'Wireshark' that 'sniffs' the packages, it captures and analyzes them, showing their information and fields.

Configuration

The switches have one setting priority. The default priority is 32768.

For the istance 0 the priority of every switch is the follow:

- SW1: 0;
- SW2: 32768;
- SW3: 32768;

For the istance 22 the priority of every switch is the follow:

- SW1: 0;
- SW2: 32768;

On SW3 is configured the Traditional Spanning Tree Protocol, so it cannot have more istances.

The istance 22 is mapped to VLAN 2. On the switches 1 and 2 is configured the Multiple Spanning Tree Protocol, on switch 3 is configured the traditional Spanning Tree Protocol. On all devices the level two protocols, GVRP and LACP, are disabled.

The level three configuration assigned to every switch is the follow: IP address:

- SW1: 192.168.1.1/24;
- SW2: 192.168.1.2/24;
- SW3: 192.168.1.3/24;

Topology

In figure 5.8 is represented the topology used on physical level for the test.

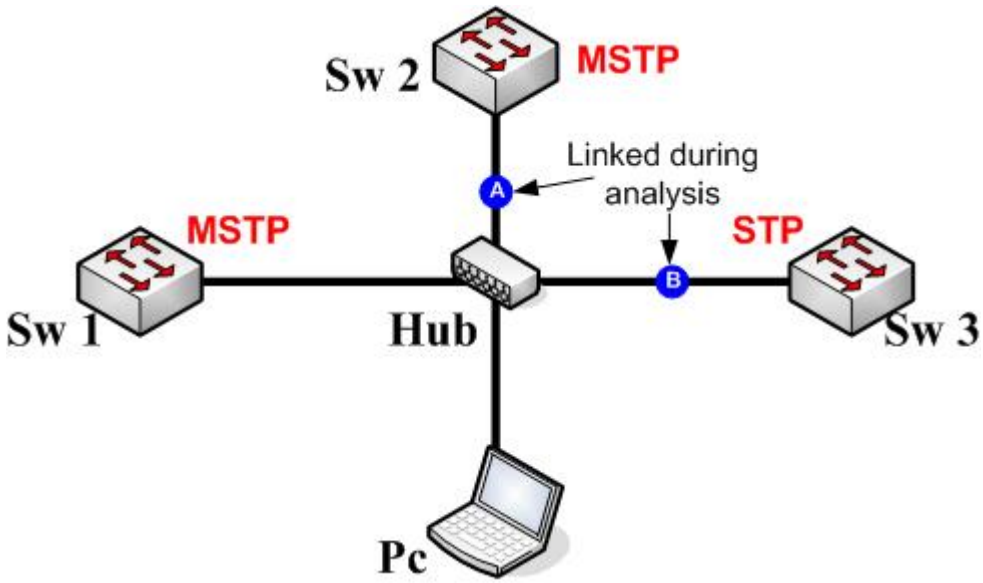

Figura 5.7: Topology MSTP in heterogeneous environment

Operations

1. On all switches execute the following command to delete the pre-existing settings:

```
SW 1,2,3:> clear config all
```

2. On SW1 execute the following command to set the IP address:

```
SW1:> set ip address 192.168.1.1 mask 255.255.255.0
```

3. On SW2 execute the following command to set the IP address:

```
SW2:> set ip address 192.168.1.2 mask 255.255.255.0
```

4. On SW3 execute the following command to set the IP address:

```
SW3:> set ip address 192.168.1.3 mask 255.255.255.0
```

5. On switches 1 and 2 execute the following command to set the Multiple Spanning Tree Protocol:

```
SW 1,2:> set spantree version mstp
```

6. On switch 3 execute the following command to set the Spanning Tree Protocol:

```
SW 3:> set spantree version stpcompatible
```

7. On all switches execute the following command to disable the GVRP Protocol:

```
SW 1,2,3:> set gvrp disable
```

8. On all switches execute the following command to disable the LACP Protocol:

```
SW 1,2,3:> set lacp disable
```

9. On all switches execute the following command to create the VLAN 2:

```
SW 1,2,3:> set vlan create 2
```

10. On switches 1 and 2 execute the following command to configure the name of the region:

```
SW 1,2:> set spantree mstcfgid cfgname pippo
```

11. On switches 1 and 2 execute the following command to create the istance 22 of MST:

```
SW 1,2:> set spantree msti sid 22 create
```

12. On switches 1 and 2 execute the following command to associate the VLAN 2 to istance 22 of MST:

```
SW 1,2:> set spantree mstmap 2 sid 22
```

13. On SW1 execute the following command to assign it the spanning tree priority:

```
SW1:> set spantree priority 0
```

14. On SW1 execute the following command to assign it the spanning tree priority for the SID 22:

```
SW1:> set spantree priority 0 22
```

15. While capturing the packages with the program Wireshark, connect SW2 and then SW3 to the hub.

Execution

The execution of the operation 15 has produced the following result:
Connection with SW2:

```
15 28.000043  Enterasy_1a:19:6a  Spanning-tree-(for-bridges)_00 STP  MST. Root = 0/00:11:88:1a:19:69
  Cost = 0  Port = 0x8001
16 28.194261  Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP  MST. Root = 32768/00:11:88:1a:19:e1
  Cost = 0  Port = 0x8001
17 28.195362  Enterasy_1a:19:6a  Spanning-tree-(for-bridges)_00 STP  MST. TC + Root = 0/00:11:88:1a:19:69
  Cost = 0  Port = 0x8001
18 28.195877  Enterasy_1a:19:6a  Spanning-tree-(for-bridges)_00 STP  MST. TC + Root = 0/00:11:88:1a:19:69
  Cost = 0  Port = 0x8001
19 28.197812  Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP  MST. Root = 32768/00:11:88:1a:19:e1
  Cost = 0  Port = 0x8001
20 28.200583  Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP  MST. TC + Root = 0/00:11:88:1a:19:69
  Cost = 0  Port = 0x8001
21 28.201600  Enterasy_1a:19:6a  Spanning-tree-(for-bridges)_00 STP  MST. TC + Root = 0/00:11:88:1a:19:69
  Cost = 0  Port = 0x8001
22 28.203628  Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP  MST. TC + Root = 0/00:11:88:1a:19:69
  Cost = 0  Port = 0x8001
23 30.000200  Enterasy_1a:19:6a  Spanning-tree-(for-bridges)_00 STP  MST. TC + Root = 0/00:11:88:1a:19:69
  Cost = 0  Port = 0x8001
24 30.123316  Enterasy_1a:19:e2  Spanning-tree-(for-bridges)_00 STP  MST. TC + Root = 0/00:11:88:1a:19:69
  Cost = 0  Port = 0x8001
25 31.999937  Enterasy_1a:19:6a  Spanning-tree-(for-bridges)_00 STP  MST. Root = 0/00:11:88:1a:19:69
  Cost = 0  Port = 0x8001
```

Connection with SW3:

```
32 45.999180  Enterasy_1a:19:6a  Spanning-tree-(for-bridges)_00 STP  MST. Root = 0/00:11:88:1a:19:69
  Cost = 0  Port = 0x8001
33 47.047461  Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP  Conf. Root = 32768/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
34 47.070387  Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP  Conf. Root = 32768/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
35 48.043519  Enterasy_1a:19:6a  Spanning-tree-(for-bridges)_00 STP  Conf. Root = 0/00:11:88:1a:19:69
  Cost = 0  Port = 0x8001
36 48.940330  Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP  Conf. Root = 32768/00:01:f4:5c:3f:20
  Cost = 0  Port = 0x8001
37 49.999509  Enterasy_1a:19:6a  Spanning-tree-(for-bridges)_00 STP  Conf. Root = 0/00:11:88:1a:19:69
  Cost = 0  Port = 0x8001
38 50.069944  Enterasy_5c:3f:21  Spanning-tree-(for-bridges)_00 STP  Topology Change Notification
39 51.999533  Enterasy_1a:19:6a  Spanning-tree-(for-bridges)_00 STP  Conf. Root = 0/00:11:88:1a:19:69
  Cost = 0  Port = 0x8001
```

Test Result

Seen the results obtained in point 15 of execution of the test is possible to assert that SW1 will sends MST BPDU's when it's alone in the network; when it's connected to SW2, it continues sending MST BPDU's. When it's connected with SW3 that is configured with traditional STP, it sends ST BPDU's.

The test result is positive.

5.8 Test IST and CIST

Test	Test Root Cost in MST Region
Object	Enterasys SecureStack B2H124-48P (48 ports) Chassis Serial Number: 05320111900B Chassis Firmware Revision: 03.01.24; Enterasys SecureStack C2G124-24 (24 ports) Chassis Serial Number: 04400122900B Chassis Firmware Revision: 04.00.24; Enterasys SecureStack C2H124-48 (48 ports) Chassis Serial Number: 05210107900D Chassis Firmware Revision: 04.00.31 Enterasys Matrix N3 serie Platinum (24 ports) Chassis Serial Number: 0001f45c06e9 Chassis Firmware Revision: 05.11.29 Enterasys Matrix N3 serie Platinum (48 ports) Chassis Serial Number: 0001f47f0575 Chassis Firmware Revision: 05.32.06 Enterasys Matrix N3 serie Platinum (24 ports) Chassis Serial Number: 0001f45c06e9 Chassis Firmware Revision: 05.11.29
Target	To verify the functioning of creation of IST and CIST on switch Enterasys entry level.
Designer	Giulia Papini
Executor	Giulia Papini (11/04/2007)
Result	Positive

Object of the experiment

The object of the experiment is composed of six stackable switch Enterasys:

- SW1: Enterasys SecureStack B2

  B2H124-48P (48 ports)

  Chassis Serial Number: 05320111900B

Chassis Firmware Revision: 03.01.24

- SW2: Enterasys SecureStack C2

  C2G124-24 (24 ports)

  Chassis Serial Number: 04400122900B

  Chassis Firmware Revision: 04.00.24

- SW3: Enterasys SecureStack C2

  C2H124-48 (48 ports)

  Chassis Serial Number: 05210107900D

  Chassis Firmware Revision: 04.00.31

- SW4: Enterasys Matrix N3 serie Platinum (24 ports)

  Chassis Serial Number: 0001f45c06e9

  Chassis Firmware Revision: 05.11.29

- SW5: Enterasys Matrix N3 serie Platinum (48 ports)

  Chassis Serial Number: 0001f47f0575

  Chassis Firmware Revision: 05.32.06

- SW6: Enterasys Matrix N3 serie Platinum (24 ports)

  Chassis Serial Number: 0001f45c06e9

  Chassis Firmware Revision: 05.11.29

Target of the experiment

To verify the functioning of creation of IST and CIST on switches Enterasys entry level.

Expected results

The expected result is the creation of a CIST that cover all the switches and one IST for each region, by the discarding of some ports for istance 22 and some ports for istance 0.

Rationales

It is chosen to use two connected triangular configuration of switch belonging to two different regions because it allow to show as the MSTP generate the CIST and IST's.

Configuration

The switches have one setting priority. The default priority is 32768.

The priority of every switch for istance 0 is the follow:

- SW1: 32768;
- SW2: 0;
- SW3: 32768;
- SW4: 32768;
- SW5: 32768;
- SW6: 32768;

The priority of every switch for istance 22 is the follow:

- SW1: 0;
- SW2: 32768;
- SW3: 32768;

- SW4: 32768;
- SW5: 32768;
- SW6: 0;

On all the switches is configured the Multiple Spanning Tree Protocol and the level two protocols, GVRP and LACP, are disabled.

The level three configuration assigned to every switch is the follow: IP address:

- SW1: 192.168.1.1/24;
- SW2: 192.168.1.2/24;
- SW3: 192.168.1.3/24;
- SW4: 192.168.1.4/24;
- SW5: 192.168.1.5/24;
- SW6: 192.168.1.6/24;

Topology

In figure 5.8 is represented the topology used on physical level for the test.

Operations

1. On all switches execute the following command to delete the pre-existing settings:

```
SW 1,2,3,4,5,6:> clear config all
```

2. On SW1 execute the following command to set the IP address:

```
SW1:> set ip address 192.168.1.1 mask 255.255.255.0
```

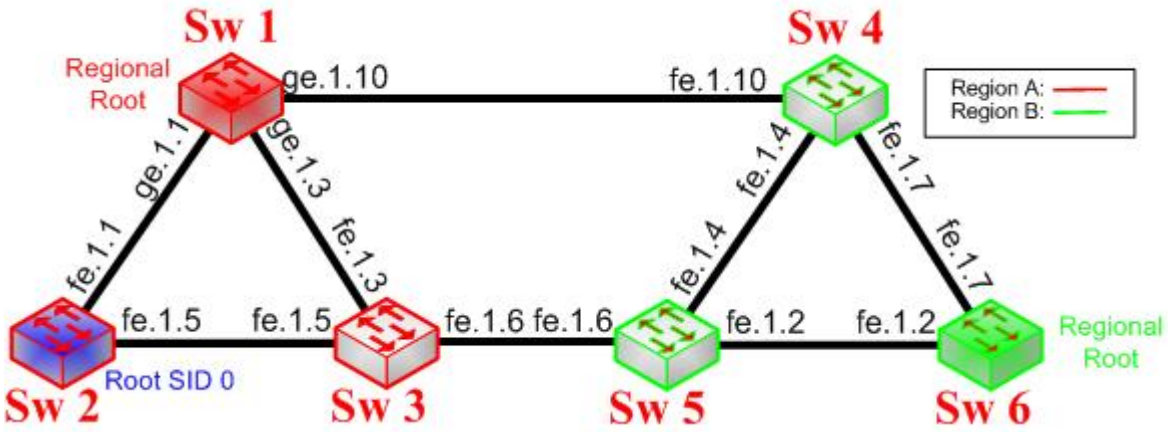

Figura 5.8: Topology CIST and IST

3. On SW2 execute the following command to set the IP address:

```
SW2:> set ip address 192.168.1.2 mask 255.255.255.0
```

4. On SW3 execute the following command to set the IP address:

```
SW3:> set ip address 192.168.1.3 mask 255.255.255.0
```

5. On SW4 execute the following command to set the IP address:

```
SW4:> set ip address 192.168.1.4 mask 255.255.255.0
```

6. On SW5 execute the following command to set the IP address:

```
SW5:> set ip address 192.168.1.5 mask 255.255.255.0
```

7. On SW6 execute the following command to set the IP address:

```
SW6:> set ip address 192.168.1.6 mask 255.255.255.0
```

8. On all switches execute the following command to set the Multiple Spanning Tree Protocol:

```
SW 1,2,3,4,5,6:> set spantree version mstp
```

9. On all switches execute the following command to disable the GVRP Protocol:

```
SW 1,2,3,4,5,6:> set gvrp disable
```

10. On all switches execute the following command to disable the LACP Protocol:

```
SW 1,2,3,4,5,6:> set lacp disable
```

11. On all switches execute the following command to create the VLAN 2:

```
SW 1,2,3,4,5,6:> set vlan create 2
```

12. On switches 1, 2, 3 execute the following command to configure the name of the region:

```
SW 1,2,3:> set spantree mstcfgid cfgname regA
```

13. On switches 4, 5, 6 execute the following command to configure the name of the region:

```
SW 4,5,6:> set spantree mstcfgid cfgname regB
```

14. On SW2 execute the following command to assign it the spanning tree priority for SID 0:

```
SW2:> set spantree priority 0
```

15. On SW1 execute the following command to assign it the spanning tree priority for SID 22:

```
SW1:> set spantree priority 0 22
```

16. On SW6 execute the following command to assign it the spanning tree priority for SID 22:

```
SW6:> set spantree priority 0 22
```

17. On all switches execute the following command to setting the port linked with the others switch as 'source port' and a port linked with PC1 as 'destination port':

```
SW 1,2,3,4,5,6:> set port mirroring create [source] [destination]
```

18. On all switch execute the following command to enable the port linked with the others switch as 'source port' and a port linked with PC1 as 'destination port':

```
SW 1,2,3,4,5,6:> set port mirroring enable [source] [destination]
```

19. On all ports of all switches execute the following command to see the port's status:

```
SW 1,2,3,4,5,6:> show spantree stats port xxx
```

Execution

The execution of operations 17, 18 and 19 has produced the result descripted in figure 5.9.

Test Result

Seen the results obtained in point 17, 18 and 19 of execution of the test is possible to assert that the configuration has generate a CIST that cover all the switch and one IST for each region, by the discarding of some ports for istance 22 and some ports for istance 0.

The test result is positive.

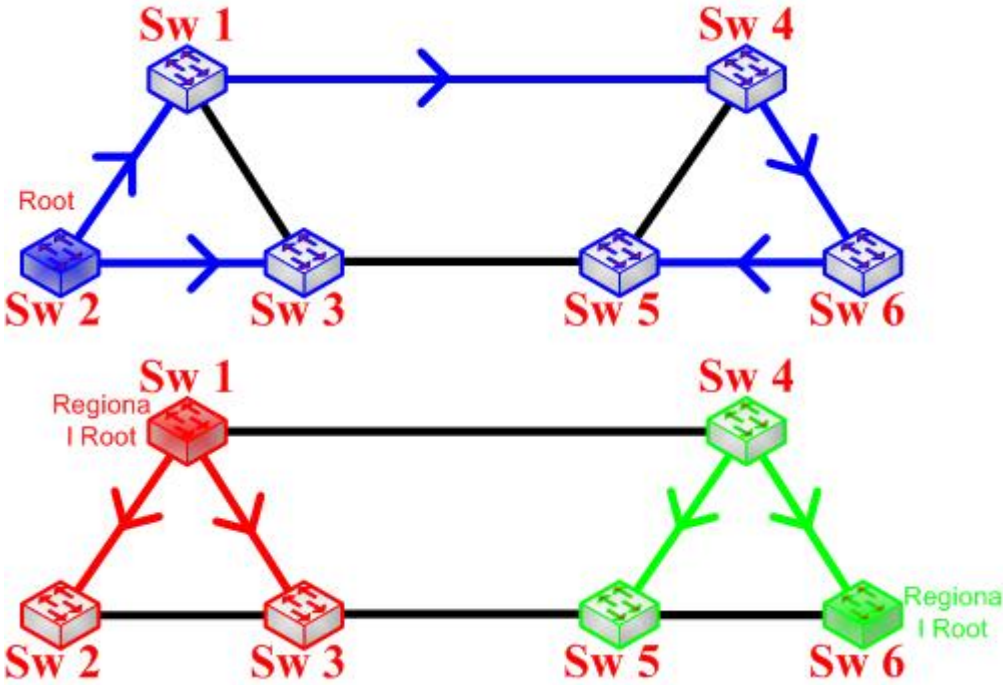

Figura 5.9: Topology result CIST and IST