Jack Holland, Ryan Stearns

Lab 1

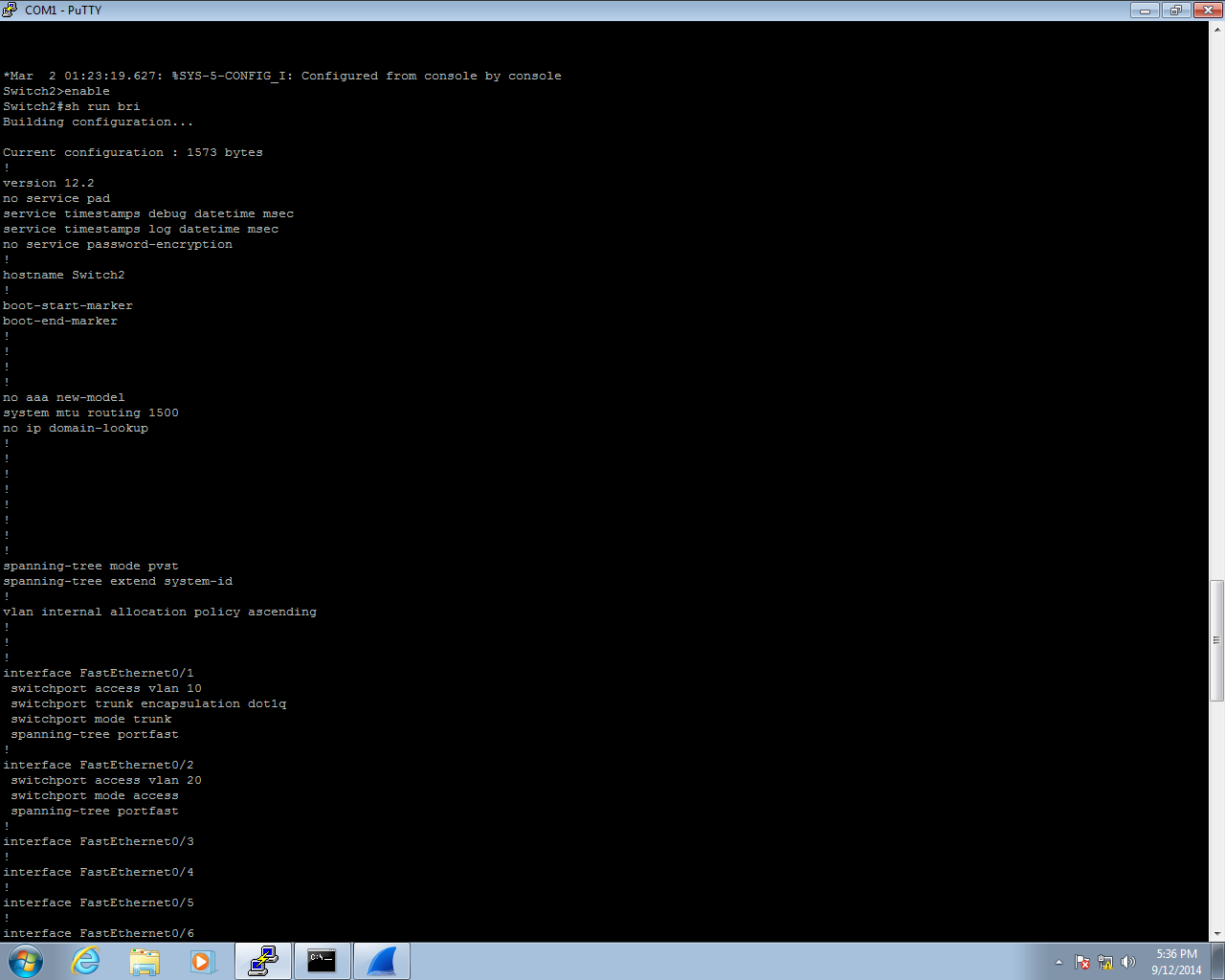
Due: 9/15/2014

VLAN Hopping Using Double Encapsulation

For our lab we configured two Cisco Catalyst 3560 Series Switches using the application Putty. We connected a Windows 7 Computer to each of the switches via a console cable for the configuration. Accordingly, we created a Virtual Local Area Network on each switch. For the first switch, which we named Switch1, we created VLAN 10 and configured port Fa0/1 as a trunk port and Fa0/2 as an access port for our Backtrack 5 Linux machine. For the second switch, labeled Switch2, we created VLAN 20 and configured the ports in the same way as was performed on the first switch, except for our Windows machine.

Once the VLANs were created, we had to assign IP addresses to each VLAN. For VLAN 10 residing on Switch1, we assigned an IP address of 10.1.1.254 with a subnet mask of 255.255.255.0. For VLAN 20 residing on Switch2, we assigned an IP address of 10.1.2.254 with a subnet mask of 255.255.255.0.

We then configured Fa/01 on each switch to be the trunk port using the Cisco Command :



From the VLAN connected machines we were able to set Static IP Addresses for the computers connected to VLAN 10 and VLAN 20. For the Backtrack Linux Machine connected to port Fa0/2 of Switch1, we set the IP Address to be assigned the IP 10.1.1.10 from the Backtrack machine using the GUI. For the Windows machine connected to port Fa0/2 of Switch2, we set the IP Address of 10.1.2.10 using the windows network configuration tools.

To ensure forwarding of the frame, we turned off Cisco Inter-Switch Link, the default method of encapsulation, and enabled IEEE 802.1Q Encapsulation for both trunk ports. We theorized at this point that if a frame was crafted with double encapsulation, that we should be able to successfully attack the Windows machine residing on VLAN 20 of Switch2, from the Backtrack machine which resides on VLAN 10 of Switch1.

We then crafted a double encapsulated frame in a packet manipulation program called Scapy on the Backtrack machine. We wrote a few different versions of a Python Script which invoked the program Scapy in an attempt to attack the Windows machine on VLAN 20 of Switch2. The script varies from directly setting the destination IP address and MAC address of the Windows machine, to sending the frame to a broadcast address.

We were able to monitor the traffic on the Windows Machine from the network protocol analyzer program called Wireshark. Despite our best efforts, we were unable to capture the ICMP Packet on the Windows machine. A second Windows machine was hooked up to VLAN 10 alongside Backtrack, so we could monitor the packet being sent out.

After a ton of testing and rewriting code, we concluded that it was not possible to attack VLAN 20 from VLAN 10 using the double encapsulation VLAN hopping technique. Further research supported our conclusion, “Its possible to hop from VLAN 1 to other VLANs, but it’s not possible to hop from VLAN 2 or 3 to other VLANs. As VLAN 1 is the native VLAN (default configuration), only VLAN 1 is two times decapsulated” (SANS Institute, Pg. 16) “In order to attack from one VLAN to another, attackers need to meet some specific conditions, but this is the set up by default. In order to avoid the possibility of VLAN hopping and double tagged 802.1q attacks, the administrator should dedicate VLAN other than VLAN 1 for trunking and assign specific VLAN.” (SANS Institute, Pg. 24) Since we were dealing with VLAN 10 and VLAN 20, and not VLAN 1, this eliminated the vulnerability that is evident when using VLAN 1 as a trunk port.

In conclusion we spent too much time on an ass-backwards impossible lab. The end.

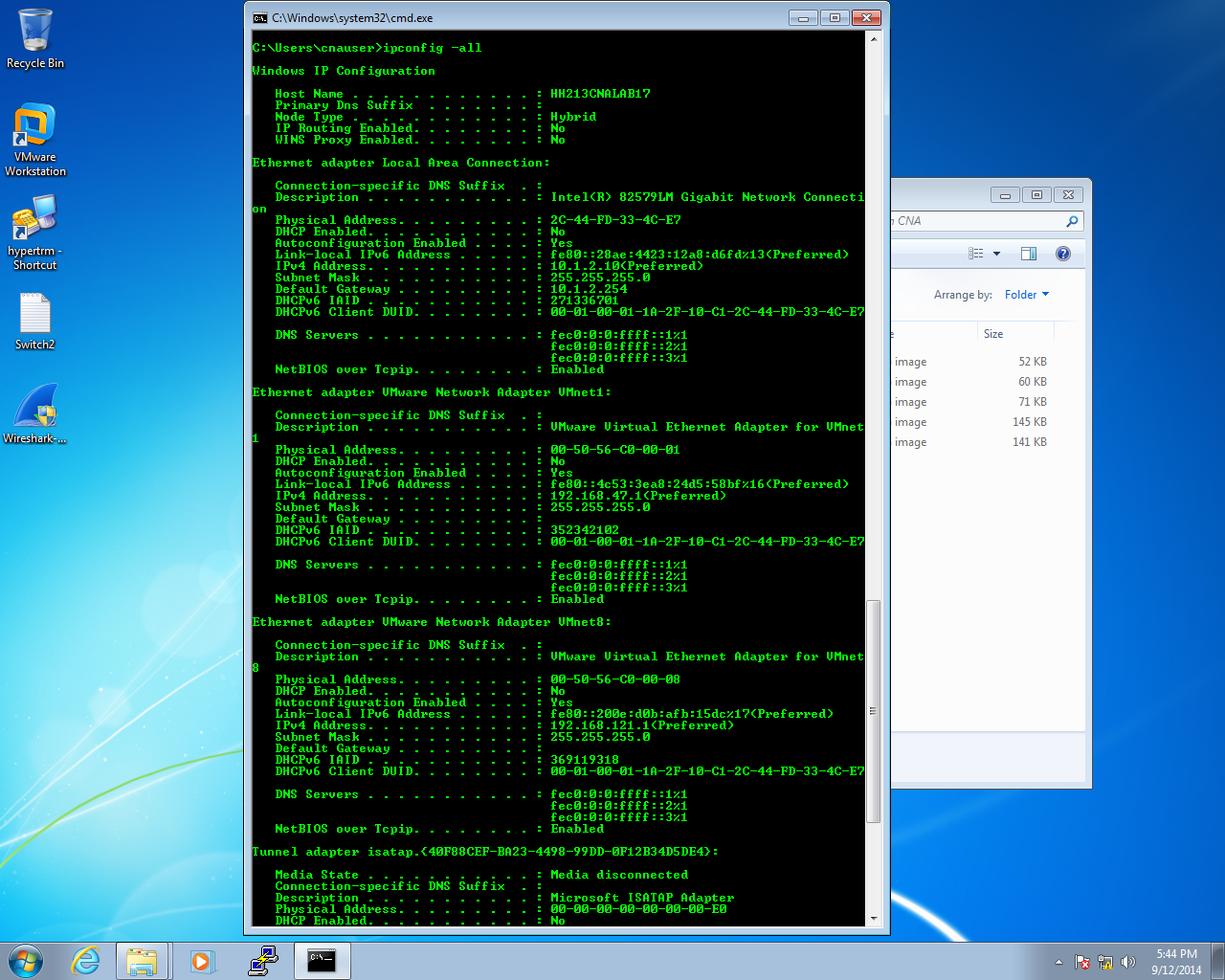
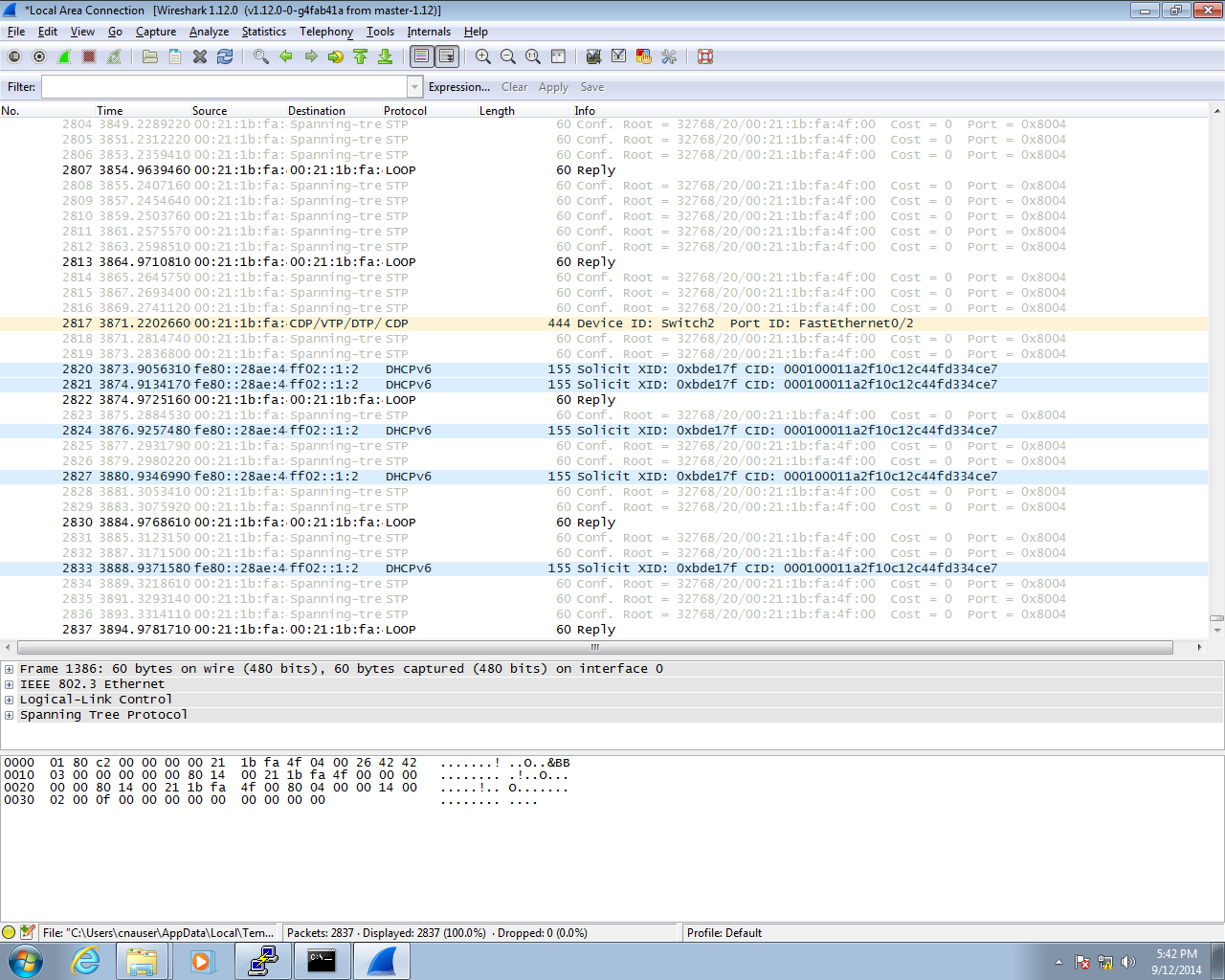
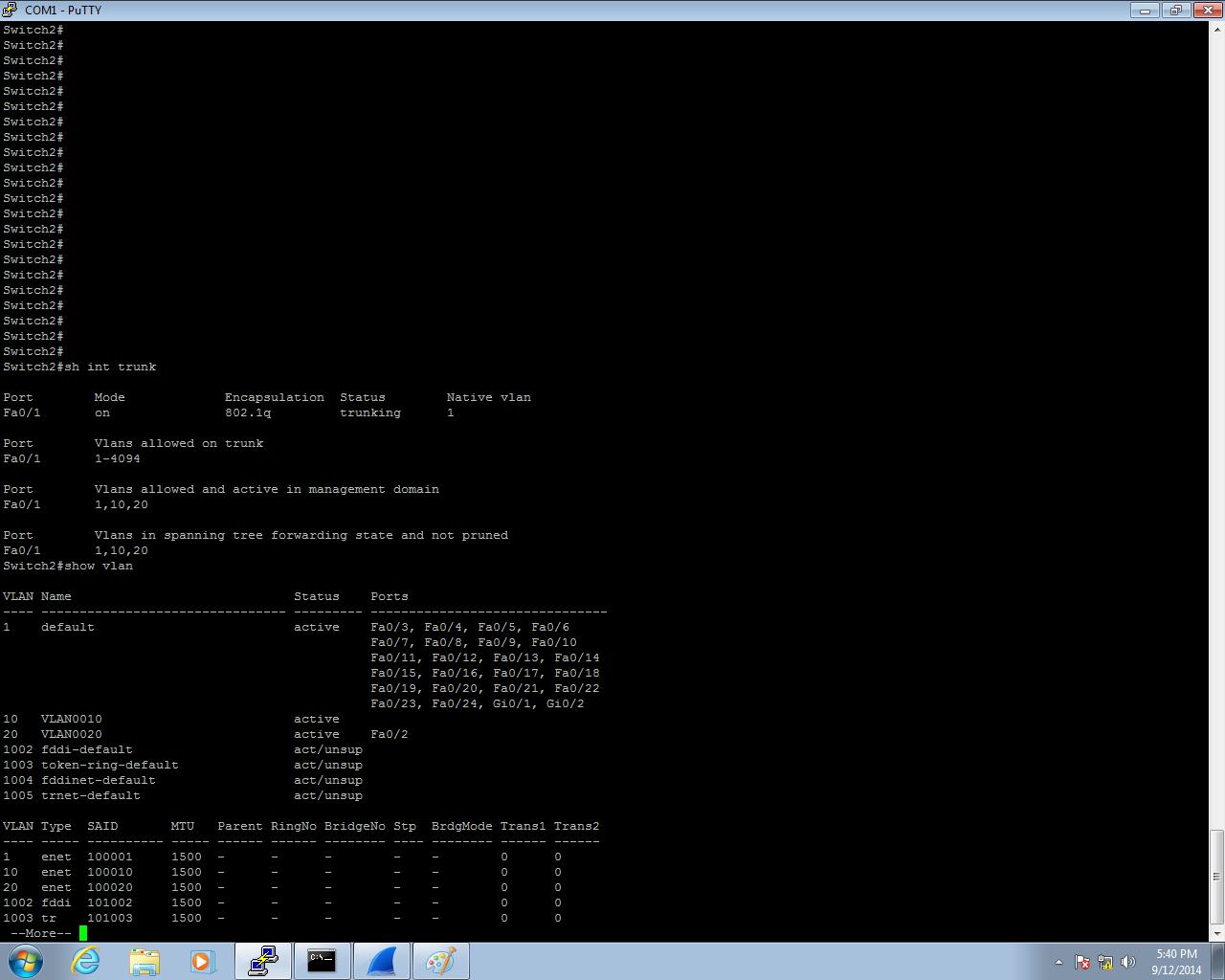
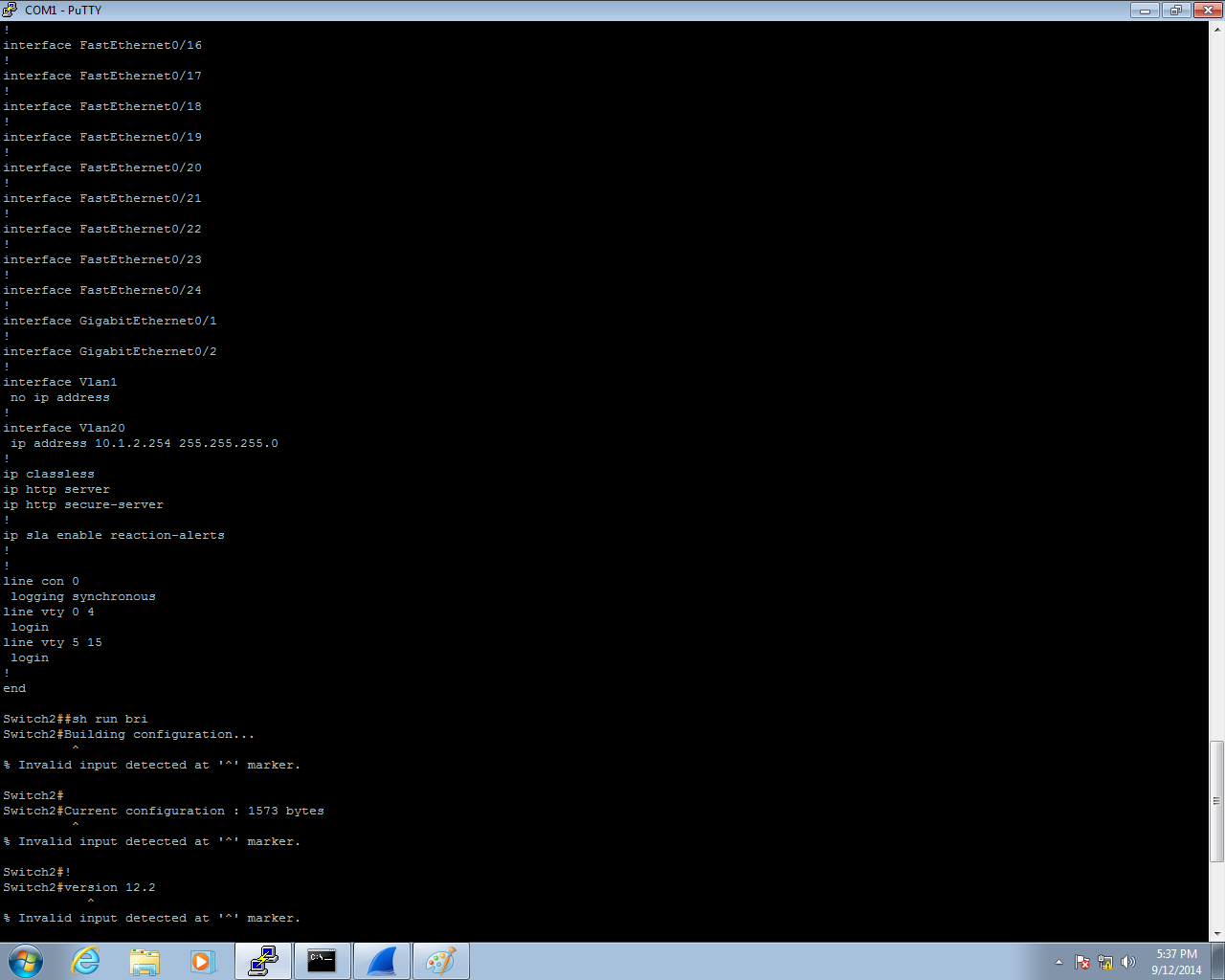
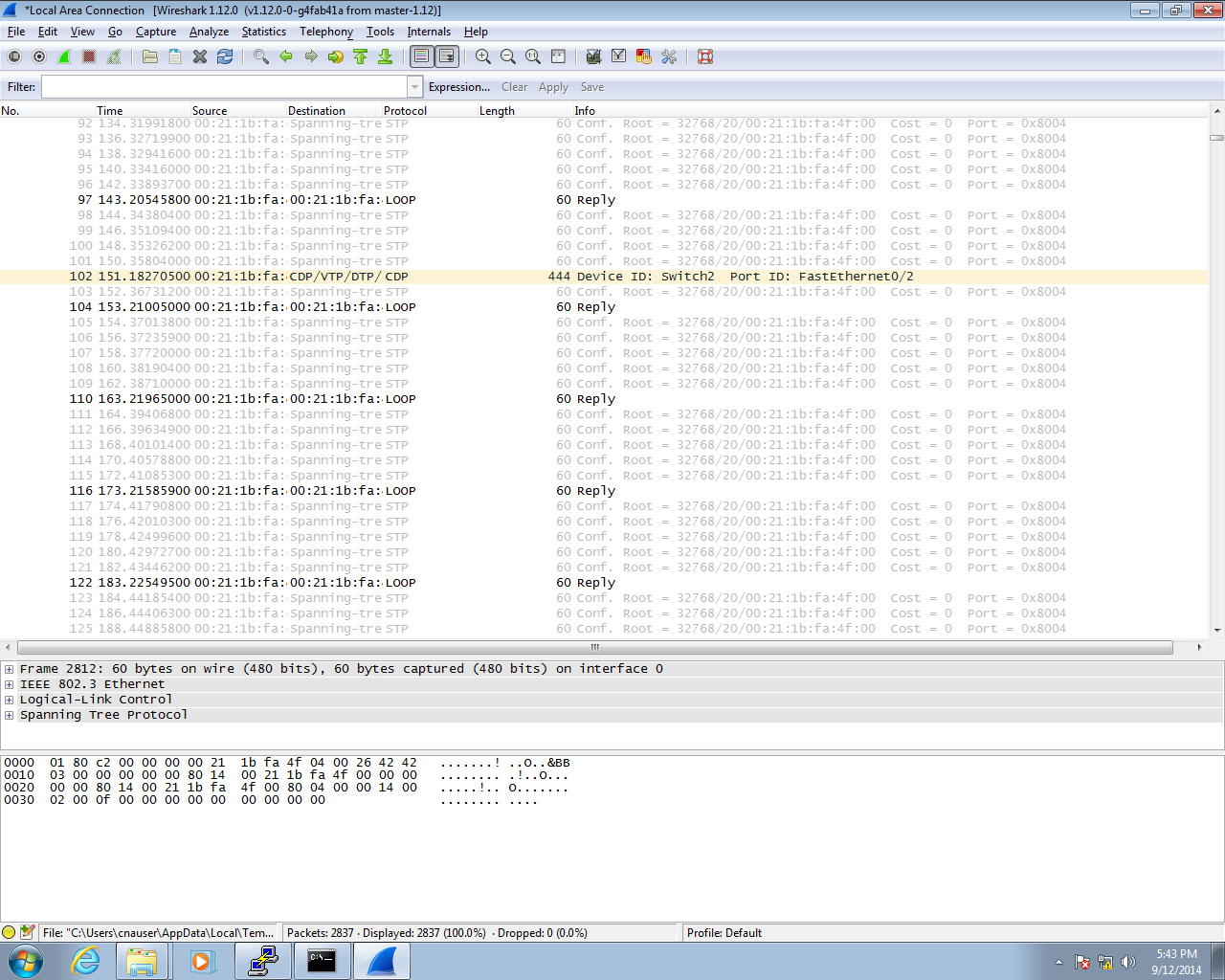


Figure 4.5.1 from Reading Page 16

“Table 5 shows that its possible to hop from VLAN 1 to other VLANs, but it’s not possible to hop from VLAN 2 or 3 to other VLANs. As VLAN 1 is the native VLAN (default configuration), only VLAN 1 is two times decapsulated.”



4.5.3  
“It was concluded that the traffic from VLAN 1 was allowed to hop to other VLANs because the trunk port was also set (implicitly) to native VLAN 1.”

4.5.4  
“It is possible to get 802.1q frames to hop from one VLAN to another if the framers are injected into a switch port belonging to the native VLAN of the trunk port. It is also necessary for the source and destination Ethernet devices to be on different switches.”  


5 Conclusion  
“In order to attack from one VLAN to another, attackers need to meet some specific conditions, but this is the set up by default. In order to avoid the possibility of VLAN hopping and double tagged 802.1q attacks, the administrator should dedicate VLAN other than VLAN 1 for trunking and assign specific VLAN.”