**Michael Simmons**

**IT4323 – W01**

**Project 35 - Individual**

**Project Wireshark**

Description: This project requires that you work in a group (or as an individual) to do some research regarding the protocols used in the Wireshark captures provided to try and determine what is happening in the capture. Keep in mind that documentation is a big part of what you will do in the real world in your career. You will have to explain to others your findings and propose solutions. “Hands on” means using your brain, too!

Use the capture files in the Project Work Module in D2L (you will have to unzip the zip files to extract the individual capture files) to answer the following questions:

**Part I – HTTP**

1. Review the first capture file **(Project Part I-a)** and determine what is happening with the HTTP traffic.
2. Describe the traffic: what packets are involved and what is happening? (include source, destination, time of capture)

**An HTTP GET request has been made (packet 4) from source 145.254.160.237 to destination 65.208.228.223 on May 13, 2004 at 6:17am. The request took 9ms and is accessing** [**http://www.ethereal.com/development.html\r\n**](http://www.ethereal.com/development.html\r\n) **from a Google search. After acknowledgement on port 80, the source IP contacts the DNS once for the Google ad link for Ethereal. The query acknowledgement is (145.253.2.203, packet 13) and the response is (145.254.160.237, packet 17). The source IP then sends a HTTP GET request (216.239.59.99, packet 18), the link contains HTML code driving traffic to the Ethereal page to download the program. HTTP/1.1 200 OK from 216.239.59.99 back to the source 145.254.160.237 (packet 27)**

1. Take a screenshot of the actual packets within the capture file that you observed this behavior.

Table

Description automatically generated

1. Review the second capture file **(Project Part I-b**) and determine what is happening with the HTTP traffic in this capture.
2. How is the traffic different from the first capture? Describe the traffic: what packets are involved and what is happening? (include source, destination, time of capture)

**On March 1, 2011 at 3:45pm an HTTP GET request (packet 4) was sent from source 192.168.1.140 to destination 174.143.213.184,** [**http://packetlife.net/images/layout/logo.png**](http://packetlife.net/images/layout/logo.png)**. An PNG image was downloaded, File Data 21684 bytes. HTTP/1.1 200 OK from 174.143.213.184 back to the source 192.168.1.140 (packet 36)**

1. Take a screenshot of the actual packets within the capture file that you observed.

A picture containing table

Description automatically generated

**Part II – PPP**

1. Review the third capture file **(Project Part II-a)** and determine what is happening with the PPP traffic in this capture.

a. Research **one** of the protocols relating to PPP and describe it here.

**The PPPoED is attempting to establish a point-to-point connection over ethernet to setup a router. This connection must be confirmed before PPP LCP, PAP, IPCP, or any others can configure, test, or send data.**

1. Describe the traffic: what packets are involved and what is happening? (include source, destination, time of capture)

**The traffic appears to be making a configuration request over the ethernet to setup a new router on Sep 13, 2010 at 9:43am, PPPoED initializes the broadcast router on the ethernet connection (packet 1). The new source router ca:01:0e:88:00:06 makes itself discoverable (packet 2) to the destination broadcast router ca:05:0e:88:00:00. Broadcast ca:05:0e:88:00:00 sends the request to join the network (packet 3). The new router accepts and confirms the session join (packet 4). The broadcast establishes link control with the new router (PPP LCP, packet 5), the two routers begin configuration requests and acknowledgement (though packet 10). Password Authentication Protocol begins asking for an ID and Password (PPP PAP, packet 11), and is acknowledged (packet 12). The Internet Protocol Control Protocol then requests the internet connection but Configure-Nak rejects IPv4, only connecting IPv6 (packet 13-21). ICMPv6 message report configuration (packet 22-35). DHCPv6 configuration (packet 36 – 39).**

1. Take a screenshot of the actual packets within the capture file that you observed this behavior.

Table

Description automatically generated

1. Review the third capture file (**Project Part II-b)** and determine what is happening with the PPP traffic that you are investigating in this capture. What else is involved?

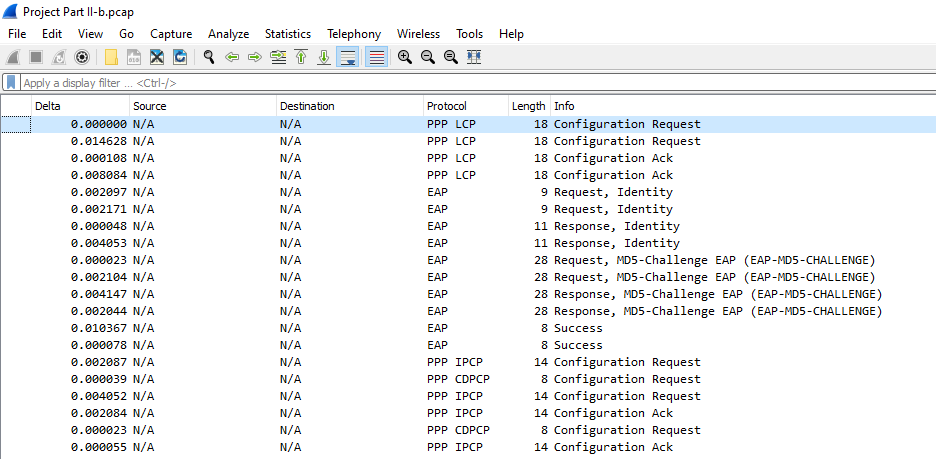
a. Research **one** of the protocols relating to PPP and describe it here.

**EAP establishes authentication between the source host and destination host. It requests the identity of the source host then confirms with the destination host if it is authorized to connect. (packet 5-14).**

1. Describe the traffic: what packets are involved and what is happening? (include source, destination, time of capture)

**On Jun 7, 2010 at 12:00pm, PPP LCP configures the link to the authenticator (packet 1-4). EAP then requests the identity of the source host, the host will response detailing its identity (packet 5-8). An EAP Challenge request (packet 9-10) is sent to the destination server for authorization response (packet 11-12). If approved the authenticator grants access and configuration is allowed to proceed (packet 13-14). Note: the authenticator sit between the requestor and the server, so source and destination addresses are not displayed.**

1. Take a screenshot of the actual packets within the capture file that you observed this behavior.



1. Review the third capture file **(Project Part II-c**) and determine what is happening with the PPP traffic in this capture. What else is involved?

a. Research **one** of the protocols relating to PPP and describe it here.

**0x002f is the Van Jacobson Uncompressed TCP/IP, it belongs to the PPP Protocol ID. It identifies the type of information encapsulated in packets of information that contain configuration details or data.**

1. Describe the traffic: what packets are involved and what is happening? (include source, destination, time of capture)

**On Aug 4, 2008 at 11:00pm, source 191.1.13.1 requests to sync with destination 191.1.13.3 over TCP (packet 1). The request is acknowledged, and a response is sent by 191.1.13.3 (packet 2). The data to be sent is then compressed using the 0x002d PPP Van Jacobson protocol (packet 6-18). PPP LCP then send an Echo Request/Reply to ensure the connection is still good (packet 19-22).**

1. Take a screenshot of the actual packets within the capture file that you observed this behavior.

Graphical user interface, application, table

Description automatically generated

**Part III – 802.11**

1. Review the first capture file **(Nokia)** and determine what is happening with the 802.11 traffic. Hint: use the Analyze/Conversation Filter
2. Describe the traffic: what packets are involved and what is happening? (include source, destination, time of capture)

**On Dec. 31, 1999 at 7:04pm Beacon frames are captured establishing connectivity from a wireless access point, Siemens\_41:bd:6e, to anything in range (packet 1-151). Some data is made visible on the network (packet 152-431). NokiaDan\_3d:aa:57 sends a probe request to the broadcast network (packet 689). Siemens\_41:bd:6e responds to NokiaDan\_3d:aa:57, acknowledging communication (packet 690-701). NokiaDan\_3d:aa:57 then send authentication/acknowledgment to Siemens\_41:bd:6e (packet 715-716), Siemens\_41:bd:6e returns the same (packet 717-718). The association request/response is acknowledged (719-722) and Siemens\_41:bd:6e sends the WPA key packets to NokiaDan\_3d:aa:57 for access using EAPOL protocol (723-742). Data begins to flow (packet 743-776).**

1. Take a screenshot of the actual packets within the capture file that you observed this behavior.

Graphical user interface, text, application

Description automatically generated

1. Review the second capture file **(wpa**) and determine what is happening with the WPA traffic in this capture. (password is “Induction”). Hint: Use Edit->preferences, Protocol IEEE 802.11, decryption with “Induction”); Look at the packets prior to and after decryption. Use this

<https://wiki.wireshark.org/HowToDecrypt802.11> . You are STILL required to answer the following questions in your own words and provide a screenshot!

1. What do you different once you have decrypted the packets?

**The decrypted packets are now displayed below the encrypted packets.**

What is decrypted and what is happening? (include source, destination, time of capture)

**On Jan 4, 2007 at 1:15am, multiple HTTP GET request have been sent to download pictures. The request from Source 192.168.0.50 to destination 209.188.21.206 to download space2.gif (packet 810). The DNS has been queried (packet 813) and HTTP 200 OK (packet 820). The request from Source 192.168.0.50 to destination 209.188.21.206 to download jaws2.gif (packet 823). The request from Source 192.168.0.50 to destination 209.188.21.206 to download jaws1.gif (packet 832).**

1. Take a screenshot of the actual packets within the capture file that you observed.

Graphical user interface, text, application, email

Description automatically generated

**Part IV – Ipv6 – Ipv4**

1. Review the third capture file **(teredo)** and determine what is happening with the Ipv4-IPv6 traffic in this capture.
   1. Research Teredo for encapsulation relating to Ipv6 using the content links in the course and describe it here.

**Teredo allows an IPv4 host full connection to an IPv6 host. It uses a tunneling protocol to provide IPv6 connectivity by encapsulating IPv6 data packets into IPv4 UDP packets. Used as a temporary connection until IPv6 can be fully implemented.**

1. Describe the traffic: what packets are involved and what is happening? (include source, destination, time of capture) (Hint: look at all Ipv6 packets that have Toredo in their details)

**On May 16, 2008 at 11:50am, laptop source 192.168.2.16 is attempting to connect to the internet using an IPv4 host. It is rejected over TCP. The request is reset and acknowledged (packet 18); the DNS query reports the connection requested is IPv6 (21-24). Teredo spins up and runs a connectivity test over UDP (packet 30). Converts IPv4 toIPv6 temporarily (packet 31-57). The DNS IPv6 runs again (packet 58) allowing access to the HTTP page (packet 73).**

1. Take a screenshot of the actual packets within the capture file that you observed this behavior.

Graphical user interface, application

Description automatically generated with medium confidence

1. Review the fourth capture file **(6to4)** and determine what is happening with the Ipv4- IPv6 traffic in this capture. What else is involved?

a. Research 6to4 protocol using the content links in the course relating to 6to4 and describe it here.

**6to4 protocol is a way to gain IPv6 connection via an IPv4 host. It is mainly used for static addresses. Unlike Teredo, it does not use tunneling to complete this. Instead it uses its own gateway that unencapsulates data through the network interface.**

1. Describe the traffic: what packets are involved and what is happening? (include source,

destination, time of capture)

**On May 16, 2008 at 11:59am, source 2002:4637:d5d3::4637:d5d3 initiates an HTTP GET request to 2001:4860:0:2001::68 (packet 1). The 6to4 protocol has converted the IPv4 to IPv6 connection with the first packet. The IPv6 connection is the carried over to all subsequent packet, completing the request (packet 4) without rejection over TCP.**

1. Take a screenshot of the actual packets within the capture file that you observed this behavior.

Graphical user interface, text, application, email

Description automatically generated

1. How do the two different methods differ?

**6to4 is completed within the first packet through an independent gateway. The TCP protocol is completed without being rejected or converting to UDP.**

**Teredo will convert to IPv6 after IPv4 has been rejected, it then must encapsulate the data over UDP for transport.**

1. Are there any other transitioning methods for IPv6 that you came across in your research that are noteworthy?

**The AYIYA protocol can also support UDP encapsulation for transport over NAT connection.**

**Group Members: (List your group Members Here if you worked in a group)**

Task List: (List the tasks that were performed in this project and which team members were involved in this task, plus the % of their contribution to the project)

**Submission**

After you have finished answering all the questions, please submit this part of the project to the drop box for this submission as a group.