17.1 Wireshark

This tutorial tries to give you some hands-on experience in using wireshark. Wireshark offers you a lot of assistance when it comes to analyzing sniffed traffic. We encourage you to make use of them.

17.1.1 Warming Up

- Start up wireshark **in the lab VM**
- Start a capture session on your main network interface (`eth0`).
- Open a terminal and issue the command: `curl -q http://www.nt.uni-saarland.de/`

Answer the following questions **using only your packet capture**:

- How many distinct flows of data can you identify?

Solution: Between one and three, depending on the local caches. They can be easily identified in wireshark using the coloring in the packet view. For busier networks you can use the conversation statistics (*Statistics* → *Conversations*).

- What protocols are used to facilitate your web request?

Solution: ARP, DNS and HTTP

- Do you see any ARP requests or responses? If so why, if not why not?

Solution: You should. In order to send an HTTP request to the web server, which is not on the local network, packets have to be routed through the default router of the subnet. To send data to the router, the system has to know its MAC address.

If there is no ARP exchange, the MAC address was already resolved earlier and is taken from the ARP cache.

- How many DNS request do you see? Why is there more than one?

Solution: You should see two,

The local resolver tries to resolve the name of the server to its IP address. Since there are two versions of IP addresses (v4 and v6), the local resolver issues a DNS request for records of each type (A and AAAA).

If there are no DNS requests, the name has been resolved before and cached locally.

- How many RRs are included in the answers to the DNS queries? What are their record types?

Solution: The answers can be taken from the packet analysis breakdown of the response packets.

The IPv6 answer contains exactly one record of type CNAME pointing to web-f6.rz.uni-saarland.de.

The IPv4 answer contains two, one is the same as for IPv6 and an additioal A record for web-f6.rz.uni-saarland.de.

- What was the HTTP status code returned by the server?

Solution: 200

- What is the name of the software serving the web page?

Solution: Apache

- Does the web server set a cookie?

Solution: Yes.

There are four *Set-Cookie:* headers in the HTTP response, so the web server sets four cookies.

- How long did it take the webserver to answer your request?

Solution: The actual time may vary.

The easiest way to find out is to set the HTTP request as the capture time reference. To do so select the packet and either press *Ctrl-T* on the keyboard or right-click the packet and select *Set/Unset Time Reference*. The delay then is the capture timestamp of the response.

- Did the complete website arrive from the server in one IP packet? If not, how many packets were sent?

Solution: No. The actual number of packets may vary, but the site does not fit in a single packet.

The number of packets is presented by wireshark in the packet analysis when selecting the HTTP response in the TCP statistics.

- What is the MAC address of your host?

Solution: The specific MAC address is dependent on your VM.

It can be found either in the layer 2 information of the packet details of any packet or using the *Conversations* view (*Statistics* → *Conversations*).

- What is the MAC address of the default gateway?

Solution: same as above

- Can you find the MAC address of the webserver? If so, how? If not, why not?

Solution: You cannot. Since the web server is not on your local subnet your machine will not communicate with it on the Ethernet level directly and therefore this information is just not available.

17.1.2 What the Heck?

Open the file **scan.pcapng** in wireshark.

Background:

This capture file was taken from a very large and long-time established network that had been considered very stable and unchanging. The network administrator has given you this file that contains what he considers "suspicious" behavior and has asked you to evaluate it.

Questions:

- What is the IP address of the scanning host?
- What is the IP address of the target host?
- Which TCP port is open on the target?
- Which ICMP packets contain non-standard Type/Code numbers?

Solution:

- 192.168.1.141
- 192.168.1.123
- 68 [bootpc]
- 3, 4, 832 and 833

This exercise was taken from the Sharkfest 2013 Challenge

17.1.3 Cursed

Open the file **cursed.pcapng** in wireshark.

Background:

Sure, Scott is one of your best friends at the company, but he's always asking for computer help. No amount of training seems to work. Today he sent you a text message to complain that his computer hard drive light is always blinking - even when he's not touching the keyboard. With a promise of decent drinks after work, you remotely connected to his machine and started capturing traffic. Sure enough - loads of packets were flying around. Just then, Scott arrived in your office.

Hmmm... Scott is here, but his computer seemed to have a lot of network activity going on. You stopped the trace to see what happened in the background on his system.

Questions:

- How many different IP hosts is Scott's machine communicating with?
- What is the average packets-per-second rate seen in this trace file?
- How many HTTP POST requests did Scott's machine send?
- What location information is contained in the POST to scanscout.com?
- What application appears to be generating these GET/POST requests?
- Find export and reassemble *load_small.png*. What shape is displayed in the image?

Solution:

- **142** (select *Statistics* → *Endpoints*, subtract one)
- **30.859** (*Statistics* → *Capture File Properties*)
- **3** (use the display filter: `ip.src_host==192.168.1.108 and http.request and http.request.method==POST`)
- **San Francisco, Oakland, San Jose (807)** (look at the HTML form encoded section and search for location information)

- **Mozilla/4.0** (User-Agent field in HTTP)
- **A star** (find the request using `http.request.full_uri contains "load_small.png"`, select the respective response, export the PNG bytes to a file, inspect it in a photo viewer)

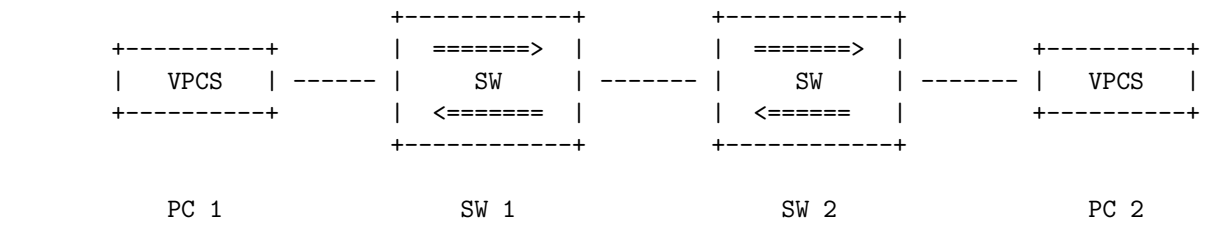
This exercise was taken from the Sharkfest 2013 Challenge

17.2 GNS3

17.2.1 Trouble in Virtual Paradise

For this exercise you will have to build a small network using GNS3.

a. Start up GNS3 and create a new project. Set up the following network.



b. Set up IP connectivity on the network by configuring PC1 and PC2 using IP addresses from the RFC1918-ranges. Verify your configuration by pinging PC2 from PC1

Solution: Possible IP ranges see U10. It is important that all hosts share the same prefix.

c. A friend of yours is asking you for help with a networking issue he has been having. His network is very similar to the network given above and has an additional computer (PC3) attached to a third switch (SW3). To make his network more resilient against link failure he thought it would be a good idea to attach SW3 not only to SW1 but to SW2 as well. You decide to help him in diagnosing the problem.

Add the additional components to your network and configure PC3 with an IP address in the same subnet as PC1 and PC2 and verify your configuration as above.

- What is the problem?

Solution: We get no replies to our pings.

- What is the cause? Describe what is happening step by step.

Solution: A storm of ARP broadcasts is flooding the network.
The problem is the loop in the network between SW1, SW2 and SW3.

Broadcasts received by a switch are sent out on all ports but the one which received it. Assuming we ping PC3 from PC1 the events are as follows:

- PC1 sends out an ARP request to the link-layer broadcast address FF:FF:FF:FF:FF:FF in an effort to find the link-layer address of PC3
- SW1 receives the broadcast and forwards it to all attached devices (SW2 and SW3)
- Both SW2 and SW3 receive the broadcast and forward it to all attached devices (SW2: PC2, SW3 / SW3: PC3 SW2)
- PC3 receives the ARP broadcast and answers it.
- Both SW2 and SW3 receive the broadcast again, this time from a different port, and send it out (again) to all attached devices.
- By now all switches are busy constantly forwarding the ARP broadcast in a merry-go-round fashion. If PC1 received the reply of PC3 it starts sending out ICMP echo requests, but the switches are already far too busy to handle them and they are dropped.

- How can you fix the problem?

Solution: **Immediate Solution:** Remove the redundant link to split up the loop.

Longterm Solution: Make sure loops such as this can be detected. Professional Network equipment offers link-layer protocols to detect such loops and disable redundant ports automatically (see Spanning Tree Protocol [IEEE 802.1D]).