**LABO 17**

**TCP & UDP**

1. Start Wireshark op en open de capture-file met als naam **Labo17.pcapng** (cfr Leho in de map Wireshark-files) waarin onder andere een FTP-sessie tussen een FTP-client en een FTP-server gecapteerd werd.
2. Deze Wireshark-capturefile bevat verschillende TCP- en UDP-sessies. Maak gebruik van de conversatiestatistieken die door Wireshark bijgehouden werden, om na te gaan hoeveel sessies er gecapteerd werden en vul dit hieronder in.

Menu Statistics > Conversations

**Aantal gecapteerde TCP-sessies** **=** 61

**Aantal gecapteerde UDP-sessies** **=** 4

Welke 4 elementen identificeren op een unieke manier zo’n sessie?

IP-adres zender, Poortnummer zender, IP-adres ontvanger, Poortnummer ontvanger (dus een socket)

1. Om enkel de segmenten te kunnen bekijken van één enkele TCP-sessie, dien je gebruik te maken van een filter. De filterexpressie kan je zelf invoeren, maar interessanter is het om dit door Wireshark zelf te laten doen.

Hieronder wordt uitgelegd hoe je bv alle TCP-segmenten kan filteren die behoren tot een sessie waartoe bv ook het eerste frame behoort.

* Klik rechts op het eerste frame en kies voor *Follow > TCP Stream.* Hierdoor wordt er een nieuw venster geopend.
* Sluit dit nieuwe venster. Je zal merken dat Wireshark automatisch een filter gegenereerd heeft op de originele frames. Deze filter zorgt ervoor dat enkel de segmenten getoond worden die tot dezelfde TCP-sessie behoren als het eerste segment.

Noteer deze filter hieronder.

**Filter =** tcp.stream eq 0

Uit hoeveel segmenten bestaat bijgevolg de eerste TCP-sessie?

30

1. Wijzig de ingestelde filter zodat je enkel de frames te zien krijgt die tot de voorlaatste TCP-sessie behoren.

Opgelet: hou er rekening mee dat de sessie vanaf 0 geteld worden!

**Filter =** tcp.stream eq 59

Er zijn 61 TCP-sessies (zie vorige opdracht) 🡪 laatste sessie = 60, maar er wordt vanaf 0 geteld, dus is het sessienummer gelijk aan 59 voor de voorlaatste sessie

Uit hoeveel segmenten bestaat de voorlaatste TCP-sessie volgens deze ingestelde filter?

114

Merk op dat dit aantal ook in de tabel met conversatiestatistieken van het TCP-protocol (zie opdracht 2) getoond wordt. Om in die tabel snel de voorlaatste TCP-sessie te kunnen vinden, dien je wel de tabel anders te sorteren.

Op welk veld moet je hiervoor sorteren?

Rel Start

1. Tot welke TCP-stream behoort frame nr 919?

13

Werkwijze: rechts klikken op pakket 476 en dan Follow TCP-stream

1. Beantwoord onderstaande vragen over de TCP-sessie waartoe frame nr 497 behoort:

Rechtslikken op frame 497 en dan Follow TCP-stream

(Note: niet op pakket 476 klikken, dat was de vorige oefening.)

1. Geef de nummers van de pakketten die voor de TCP-handshake van deze sessie gezorgd hebben

488, 489, 495

1. Geef de nummers van de pakketten die voor de TCP-finalisatie van deze sessie gezorgd hebben

721, 722, 723, 725

1. Welke clientpoort werd tijdens deze sessie gebruikt?

50095

1. Welke serverpoort werd tijdens deze sessie gebruikt?

20

Uit deze “well known port” kan je afleiden dat deze TCP-sessie gebruikt werd door het FTP-protocol (zie dia 8 in de PowerPoint-presentatie van hoofdstuk 9).

FTP staat voor File Transfer Protocol en werd vroeger veel gebruikt om files uit te wisselen tussen een client en een server. Nu gebeurt dit bijna altijd via het HTTP(S)-protocol).

1. Welke file (naam) werd in deze sessie door FTP getransfereerd?

index.txt



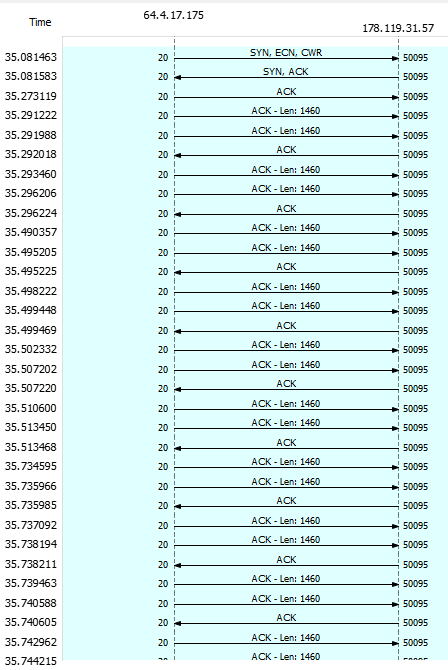
1. Wat is het IP-adres van de FTP-client in deze sessie?

178.119.31.57

1. Wat is het IP-adres van de FTP-server in deze sessie?

64.4.17.175

1. Je kan een TCP-sessie ook grafisch laten weergeven in Wireshark. Klik hiervoor op het menu *Statistics* > *Flow Graph*. Kies onderaan in het venster voor “Limit to displayed filter” en bij het item “Flow Types” voor “TCP Flows”. Plak hieronder een screenshot van deze grafische voorstelling.



1. Je ziet in deze grafische voorstelling heel mooi de TCP-initialisatie tijdens het begin van de sessie en de TCP-finalisatie aan het einde van de sessie. Tussen deze 2 fases gebeurt de data-overdracht (in dit geval: de overdracht van een file mbv het FTP-protocol). Gaat het hierbij om een upload of een download van een file (vanuit het standpunt van de client)? Leg uit hoe je dit kan zien.

Note: zorg dat het vinkje “limit to display filter” aangevinkt staat.

Een download van een file van de server naar de client want de server verstuurt alle data; de client verstuurt enkel bevestigingspakketten (ACKs).

Na hoeveel verstuurde pakketten wordt er hier een bevestiging van de ontvangen pakketten gestuurd?

Na 2 pakketten

Sluit daarna het “grafische” venster.

1. Beantwoord onderstaande vragen over het eerste pakket van de TCP-sessie waartoe pakket 497 behoort:
2. Hoeveel bytes mogen er volgens dit pakket door de zender verstuurd worden vooraleer er een bevestiging teruggestuurd moet worden naar de zender?

Window = 8192 bytes

1. Wordt het gebruik van selectieve ACK’s toegelaten?

Ja: in Info-veld staat er SACK\_PERM=1 en je kan dit ook terugvinden bij de TCP-options (sack permitted) (in het middelste deelvenster bij de details van TCP)

1. Noteer hieronder het sequence number (SEQNR) en acknowledgment number (ACKNR) van het TCP-segment met nummer 497.

**SEQNR =** 1

**ACKNR =** 1

Noteer hieronder het sequence number (SEQNR) en acknowledgment number (ACKNR) van het pakket met nummer 498.

**SEQNR =** 1461

**ACKNR =** 1

Leg uit hoe dit nieuwe SEQNR ontstaat.

Nieuwe SEQNR = vorige + aantal verstuurde bytes = 1 + 1460 = 1461

Leg ook uit waarom het ACKNR ongewijzigd blijft.

ACKNR = nummer van de eerste byte die verwacht wordt. Aangezien de computer 2 pakketten onmiddellijk na elkaar verstuurt, heeft hij ondertussen nog geen pakketten binnengekregen, en dus verwacht hij nog altijd hetzelfde.

1. Aangezien er tijdens deze sessie gebruikgemaakt werd van het onveilige FTP-protocol, kunnen we in Wireshark ook de inhoud van de verstuurde file bekijken. Deze staat in het nieuwe venster dat geopend wordt als je de TCP-stream volgt.

Wat is het eerste woord in die file?

Last

Wat is het laatste woord in die file?

Pack

Bewaar deze file onder de naam “gedownloade file” als een aparte .txt file op je laptop en open deze daarna eens met Notepad(++). Normaliter zou je nu dus hetzelfde moeten zien als in WIreshark…

Onderaan in het venster op de knop “Save as” klikken en bewaren onder de naam gedownloade file.txt.

1. In de vorige opdrachten hebben we ons geconcentreerd op 1 TCP-sessie waarin zich FTP-data bevond van één enkele getransfereerde file. Om alle FTP-data te tonen die zich in de hele capture file bevindt, moet je gebruikmaken van onderstaande filter:

***ftp-data***

Werd er bijgevolg in deze capture meer dan 1 file getransfereerd? Leg uit!

Neen want je ziet enkel RETR index.txt (RETR staat voor RETRIEVE).

1. Een FTP-server maakt gebruik van 2 TCP-verbindingen: een dataverbinding (1 per getransfereerd bestand) én één controleverbinding (voor het uitwisselen van gegevens om de verbinding te beheren).

In de vorige opdrachten hebben we al kennis gemaakt met een FTP-dataverbinding. Tijd dus om ook eens de controleverbinding te bekijken. Deze wordt o.m. gebruikt voor het opzetten van de FTP-verbinding, waarbij er naar een gebruikersnaam en wachtwoord gevraagd wordt door de FTP-server.

Maak van onderstaande filter gebruik om de FTP-controleverbinding te bekijken:

***ftp***

Bekijk aandachtig het Info-veld van de gefilterde pakketten om onderstaande vragen te beantwoorden:

* Welk poortnummer werd er door de FTP-server gebruikt?

21

Merk op dat dit dus een ander poortnummer is, dan deze die door de FTP-dataverbinding gebruikt werd (zie opdracht 7d).

* Wat is de naam waarmee de gebruiker zich aangemeld heeft op de FTP-server?

Anonymous

* Met welk wachtwoord heeft deze gebruiker zich aangemeld?

123456

Merk op dat het hier dus een onveilige FTP-sessie betreft, vermits we het wachtwoord zomaar kunnen lezen. Er bestaan uiteraard betere (veiliger) manieren om aan FTP te doen…

* Wat is de naam van de FTP-server die werd gecontacteerd?

[ftp.microsoft.com](ftp://ftp.microsoft.com)

* Welke file werd gedownload en uit welke servermap werd deze gedownload?

Indext.txt uit de map Softlib

1. In deze capture werden er niet alleen van TCP, maar ook van UDP gebruikgemaakt.

Maak gebruik van een filter om enkel de pakketten te zien die tot de **UDP-stream** met nummer 1 behoren.

Waarom zie je hier geen handshake-pakketten?

Omdat UDP niet aan handshaking doet.

Welke UDP-poort wordt er door de server gebruikt?

Poort 53

Van welke applicatie is dit de “well known” poort?

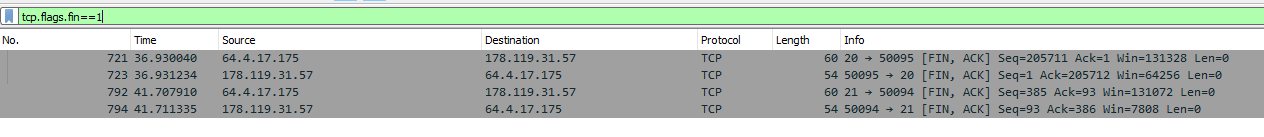
DNS

Beschrijf wat er in deze UDP-sessie gebeurt.

Een DNS-client vraagt via het DNS-protocol naar het IPv4-adres van ftp.microsoft.com en de DNS-server antwoordt met het IPv4-adres van [ftp.microsoft.com](ftp://ftp.microsoft.com).

1. Geef de nummers van de TCP-streams die netjes (mbv de 4 hiervoor gebruikte segmenten) afgesloten werden.

Eerst filteren= tcp.flags.fin==1



Daarna rechtsklikken op die frames en *Follow TCP Stream* kiezen om het nummer van de stream te zien.

Naar het einde van deze streams gaan en controleren of ze inderdaad mbv de 4 hiervoor gebruikte segmenten afgesloten werden.

Antwoord = 41 en 56