Notice

import os = import a python module, If the path does not exist, it will send an error.

try:

with open("wireshark.txt", encoding="utf8") as fh:

res=fh.read()

except:

print("Le fichier n'existe pas %s", os.path.abspath("wireshark.txt"))

Try to open the wireshark file and if the path is not correct where if opening the file is impossible then it will send an error message.

The entire file is set to "res" (res=fh.read()).

We will then create another variable and then split.

Each time we return to the line, we will create a table.

Every time we come back to the line, it’ll be a new box.

So, box one will be the first line, the second box will be the second line…

Then for "for event" is 0 up to "res", "res" is all rows in the array, it will loop until the array is finished.

if event.startswith('11:42') = To tell you when you start, so we start at 11:42 because the first packages start at 11:42.

With texte=event.split(" "), we do a "split" again but this time we do a "split" by space, so (" ").

So each space will be a new painting, a new box.

if texte[5] == "Flags":

evenement='temps : '+texte[0]+' Adresse Ip source : '+texte[2]+' Adresse IP destinataire : '+texte[4]+' flag : '+texte[6]

if texte[6] == SYN:

evenement\_3 = ' '

evenement\_2 = 'Numéro de séquence : '+texte[8]+' Taille de la fenêtre : '+texte[10]+' Longueur du paquet : '+texte[12]

if texte[6] == POUSSER:

evenement\_3 = ' '

evenement\_2 = 'Numéro de séquence : '+texte[8]+' Numéro accusé de réception : '+texte[10]

if texte[6] == "[.],":

evenement\_2 = 'Numéro accusé de réception : '+texte[8]+' Taille de la fenêtre : '+texte[10]

evenement\_3 = ' Longueur du paquet : '+texte[12]

if texte[6] == "[S.],":

evenement\_2 = 'Numéro de séquence : '+texte[8]+' Numéro accusé de réception : '+texte[10]+' Taille de la fenêtre : '+texte[12]

All this code is for sorting "Flags" because there are several types of "Flagged".

So here for example evenement='temps : '+texte[0]+' Adresse Ip source : '+texte[2]+

This is the reading according to the "Flags"

#print(evenement+evenement\_2+evenement\_3) = This is a comment to say that the program will then display the event, the event2, the event3

ipv4=texte[2].split('.') = With the IP address we have a small problem because there is a . and after the port... So we will then take the text 2 which takes the IP address between two spaces and we will redo a "split" but this time by a .

So that makes us three different spreadsheets.

So I then reason this way, we know that the port is always at the last square of the table, The port is always before the last point.

So we’re going to port: ipv4-1.

In my Program: print("Port : ",ipv4[-1]).

-1 in python language = last, last box

And print for display.

Then to sort the IP addresses we will remove the port that corresponds to each IP address, with del ipv4[-1] in my program.

stripv4 = ".".join(ipv4) = but the problem is that on several IP addresses there are also points that we will do the opposite of the “split”, we will refuse the table, “.” = replacing the boxes, so we replace with a . and after we join, we put the ipv4 address.

Then, after having gathered everything, the IP address is displayed (using a print).

try:

ip[stripv4]

except KeyError:

ip[stripv4]=1

else:

ip[stripv4]+=1

This piece of my program is for counting the number of IP addresses.

print(sorted(ip.items(), key=lambda item: item[1])) = to display the result.

fh.close() = to close the file.