קובץ readme :

ישנם 2 גרסאות קוד
עבור היוטיוב השתמשנו בגרסת קוד שיותר מתאימה לייצוג הנתונים , כל ייצוג הנתונים מופיעים בקובץ ההסברים שמצורף. – לקובץ הקוד נקרא youtube.py

ועבור שאר הדברים שנדרשים מאיתנו לנתח השתמשנו בקוד מאוד דומה עם שינויים שונים - לקובץ הקוד נקרא graph_analysis

בכל המקרים צריך לטעון את הקובץ ההקלטות שלנו כולמר הקובץ CSV
נעשה את זה בתחילת הקוד

```
# 1. Load the CSV
df = pd.read_csv('youtube_filtered.csv')
```

גרסאות והתקנות הקשורות לפייתון :

- השתמשנו ב Ubuntu 22.04 ווידאנו שזה עובד גם על windows
- גרסת הפייתון Python 3.10.11
- הורדנו לפיתון שתי ספריות על מנת שנוכל להנפיק גרפים שמבוקש מאיתנו , עשינו כך :

```
Terminal    Local
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

(.venv) PS C:\Users\ronam\PycharmProjects\PythonProject9> pip install matplotlib
```

```
Terminal    Local
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

(.venv) PS C:\Users\ronam\PycharmProjects\PythonProject9> pip install pandas
```

אלו 2 ספריות המשמשות לניתוח ויזואלי של תעבורת הרשת שהקלטנו באמצעות הWireshark
הWireshark מייצר קבצי pcapng, אותם צריך לנתח ולשלוף מהם נתונים.

הספרייה הראשונה הינה **Pandas**, אשר מאפשרת לייבא את הנתונים לקובץ csv, לעבוד עליהם כטבלת DataFrame.

אפשר לבצע עליה חישובים כמו: ממוצע גודל החבילות, מספר החבילות לכל אפליקציה, השוואת תעבורת רשת בין יישומים שונים , ועוד.

הספרייה השנייה הינה **matplotlib**, אשר לאחר שעיבדנו את הנתונים, להציג אותם בצורה גרפית.

ספרייה זו יוצרת גרפים בהתאם לבקשותינו.

שינוים שבצענו בWireshark לביצוע ניתוח מידע באופן טוב :

- הוספנו עמודה בWireshark בשם TLS Server Name

Set:

Title: TLS Server Name

Type: Custom

Field Name: tls.handshake.extensions_server_name

איך עשינו את זה ?
Open Wireshark and go to Edit → Preferences. Navigate to Columns → Click + (Add New Column).

עמודה זו מאפשרת לזהות איזה שירות או תחום החיבור מתבצע, גם אם התעבורה הינה מוצפנת.

הוא מאוד שימושי כשבודקים שירותים שונים כי אפשר להבין לאילו שרתים הוא מתחבר.

- והוספנו עמודת דגלים בwireShark של TCP כדי להציג כעת דגלי TCP בעמודה נפרדת ברשימת המנות.

Right-click on the Flags field in the Packet Details pane. Select Apply as Column. This will now show TCP flags in a separate column in the Packet List.

זוהי עמודת דגלים של TCP, המאפשרת להראות דגלים בפרוטוקול TCP , כמו למשל:
.RST,FIN,ACK,SIN

כעת על מנת שנוכל לפענח תעבורת מוצפנות נצטרך ליצור קובץ מפתחות tls על מנת שיהיה לנו גישה למידע מוצפן .

השלבים שביצענו :

1. כיבנו את האנטי-וירוס משום שהוא יכול לחסום יצירת קובץ tlskeys.log כי הוא עלול לזהות את זה כניסיון "מסוכן" לחשוף תעבורה מוצפנת.
2. הגדרת משתנה סביבת Windows לשמירת מפתחות TLS .

בעת הקלטת תעבורת רשת עם Wireshark לאתרים המשתמשים בפרוטוקול HTTPS המידע מוצפן באמצעות .(Transport Layer Security) TLS הצפנה זו מונעת מאיתנו לראות את התוכן הגולמי של הנתונים. כדי לפענח את התעבורה המוצפנת, יש צורך במפתחות TLS הידועים בשם

Wireshark לא מסוגל לפענח TLS לבד, אבל אם נגדיר את מערכת ההפעלה לשמור את ה- Pre Master Secret keys)=מפתחות הצפנה זמניים, שנוצרים כחלק מתהליך ה-TLS Handshake, זה יגרום לאבטחת התקשורת שלנו בין הלקוח לשרת.

Pre-Master Secret Keys

על מנת לאפשר ל Wireshark-לפענח את התעבורה, הגדרנו משתנה סביבת מערכת בשם SSLKEYLOGFILE שגורם לדפדפן לשמור את מפתחות ה TLS-בקובץ sslkeylog.log שנמצא בתוך תיקיית SSLKeys.

Wireshark יכול להשתמש בקובץ זה כדי לפענח את ההצפנה ולחשוף את הנתונים.

הגדרנו את המשתנה הסביבה שלנו :

Environment Variables נבחר Edit the system environment variables

ונלחץ על כפתור Environment Variables.

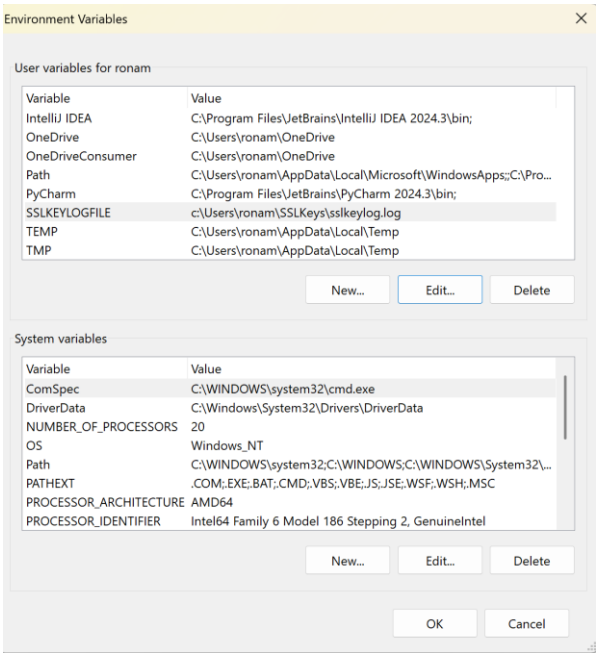
נלחץ על New תחת User variables וניתן שם משתמש SSLKEYLOGFILE

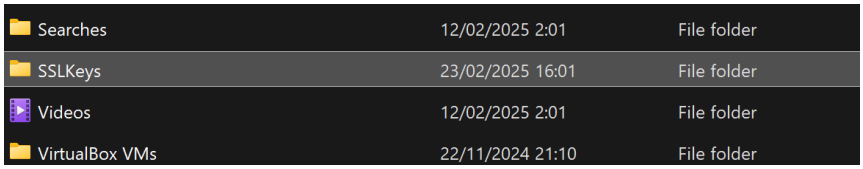
ערך המשתמש c:\Users\ronam\SSLKeys\sslkeylog.log
נלחץ על OK כדי לשמור את המשתנה.

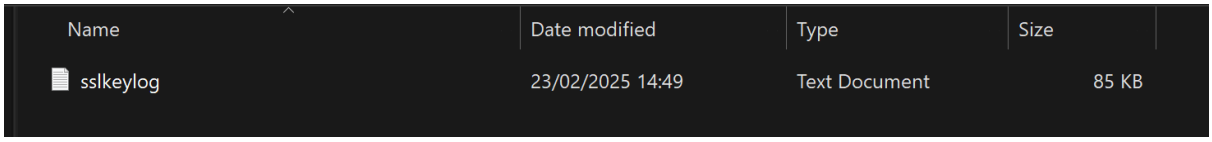
נסגור את כל חלונות הדפדפן ונפתח אותו מחדש כדי שההגדרה תיכנס לתוקף דרך CMD בתור מנהל כך:

```
C:\Windows\System32>set SSLKEYLOGFILE=c:\Users\ronam\SSLKeys\sslkeylog.log

C:\Windows\System32>start chrome.exe

C:\Windows\System32>_
```

לאחר שהגדרת משתנה הסביבה SSLKEYLOGFILE, הדפדפן ישמור את מפתחות TLS בקובץ שהגדרתי.

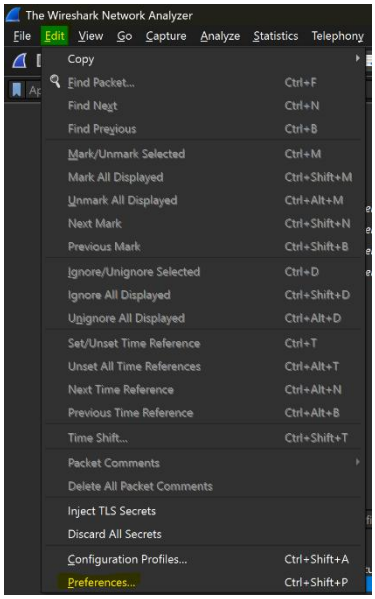


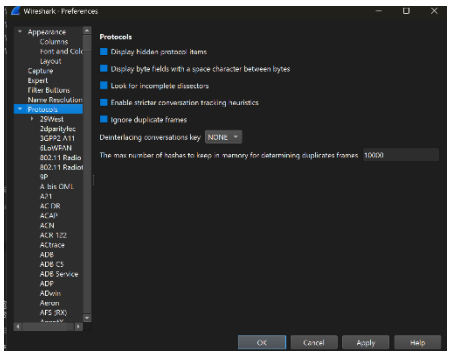


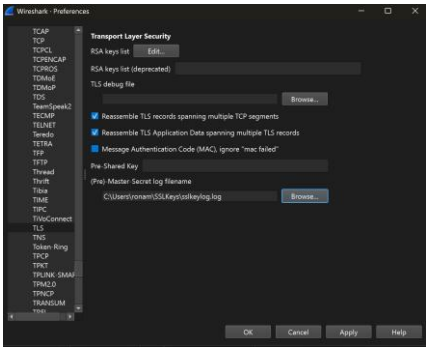


3. שימוש ב Wireshark-עם הקובץ

נפתח את ה Wireshark ונגדיר לו את הקובץ עם המפתחות שקיבלנו מהשלבים הקודמים







האפליקציות שאנו מנתחים צריכה להעביר נתונים במהירות ובצורה מאובטחת.

לכן, היא משתמש ב-2 פרוטוקולים עיקריים ישנם עוד פרוטוקולים בהם היא משתמשת, אך 2 הפרוטוקולים הבאים הינם המרכזיים:

1. **TLS:** פרוטוקול אבטחה שמצפין נתונים, כיד למנוע האזנה ותקיפות.

Spotify משתמשת בו כדי להבטיח שהתעבורה בין השרת למשתמש מוצפנת.

2. **QUIC:** זהו פרוטוקול רשת מהיר, שתוכנן כדי להאיץ חיבורי אינטרנט.

Spotify משתמשת ב -QUIC כדי להעביר את המוזיקה בפחות השהיות.

TLS Server Name	TCP Flags	Info	Length	Protocol	Destination	Source	Time	No.
		Protected Payload (KP0)	73	QUIC	35.186.224.26	192.168.68.119	246.553889	160
		Protected Payload (KP0)	73	QUIC	35.186.224.26	192.168.68.119	246.582104	161
		Protected Payload (KP0)	78	QUIC	35.186.224.26	192.168.68.119	246.669197	162
	0x002	…_Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK [SYN] 443 → 57922	66	TCP	35.186.224.26	192.168.68.119	246.826017	163
	0x010	Seq=1 Ack=1 Win=65280 Len=0 [ACK] 443 → 57922	54	TCP	35.186.224.26	192.168.68.119	246.841906	164
	0x010	…Seq=1 Ack=1 Win=65280 Len=1412 [TCP PDU rea [ACK] 443 → 57922	1466	TCP	35.186.224.26	192.168.68.119	246.842357	165
gew1-spclient.spotify.com	0x018	Client Hello (SNI=gew1-spclient.spotify.com)	733	TLSv1.2	35.186.224.26	192.168.68.119	246.842357	166
	0x010	Seq=2092 Ack=3458 Win=65280 Len=0 [ACK] 443 → 57922	54	TCP	35.186.224.26	192.168.68.119	246.908938	167
	0x018	Change Cipher Spec, Application Data	118	TLSv1.2	35.186.224.26	192.168.68.119	246.909200	168
	0x010	Seq=2156 Ack=4044 Win=64768 Len=0 [ACK] 443 → 57922	54	TCP	35.186.224.26	192.168.68.119	246.968582	169
	0x010	…57910 → 443 [ACK] Seq=2123 Ack=4044 Win=6476 [TCP Keep-Alive]	55	TCP	35.186.224.26	192.168.68.119	268.951086	170
		Initial, DCID=f1c9735951fb48d6, PKN: 1, CRYPTO	1292	QUIC	35.186.224.26	192.168.68.119	276.655174	171
gew1-spclient.spotify.com		…Initial, DCID=f1c9735951fb48d6, PKN: 2, CRYPTO, CRYPTO, CRYPT	1292	QUIC	35.186.224.26	192.168.68.119	276.655248	172
		Initial, DCID=f1c9735951fb48d6, PKN: 4, PADDING, PING, PADDING	1292	QUIC	35.186.224.26	192.168.68.119	276.685814	173
	0x018	Application Data	146	TLSv1.2	35.186.224.26	192.168.68.119	276.686029	174
	0x018	Application Data	85	TLSv1.2	35.186.224.26	192.168.68.119	276.686182	175
	0x018	Application Data	94	TLSv1.2	35.186.224.26	192.168.68.119	276.686240	176
	0x010	…Seq=3168 Ack=4044 Win=64768 Len=1412 [TCP P [ACK] 443 → 57922	1466	TCP	35.186.224.26	192.168.68.119	276.686302	177
	0x018	Application Data	1313	TLSv1.2	35.186.224.26	192.168.68.119	276.686302	178
		Initial, DCID=f1c9735951fb48d6, PKN: 6, PADDING, PING, PADDING	1292	QUIC	35.186.224.26	192.168.68.119	276.717544	179
		Protected Payload (KP0)	209	QUIC	35.186.224.26	192.168.68.119	276.723067	180
		Protected Payload (KP0)	213	QUIC	35.186.224.26	192.168.68.119	276.727203	181
		Protected Payload (KP0)	1141	QUIC	35.186.224.26	192.168.68.119	276.734164	182
		Protected Payload (KP0)	76	QUIC	35.186.224.26	192.168.68.119	276.736265	183
		Protected Payload (KP0)	75	QUIC	35.186.224.26	192.168.68.119	276.736359	184
	0x010	Seq=5839 Ack=4075 Win=64768 Len=0 [ACK] 443 → 57922	54	TCP	35.186.224.26	192.168.68.119	276.748703	185
		Protected Payload (KP0)	74	QUIC	35.186.224.26	192.168.68.119	276.779427	186
	0x018	Application Data	93	TLSv1.2	35.186.224.26	192.168.68.119	276.781668	187
		Protected Payload (KP0)	75	QUIC	35.186.224.26	192.168.68.119	276.790462	188
		Protected Payload (KP0)	79	QUIC	35.186.224.26	192.168.68.119	276.819773	189
		Initial, DCID=5142e8767785e19b, PKN: 1, CRYPTO	1292	QUIC	35.186.224.26	192.168.68.119	306.457581	190
gew1-spclient.spotify.com		…Initial, DCID=5142e8767785e19b, PKN: 2, CRYPTO, PADDING, PING	1292	QUIC	35.186.224.26	192.168.68.119	306.457991	191
	0x018	Application Data	137	TLSv1.2	35.186.224.26	192.168.68.119	306.458282	192
	0x018	Application Data	93	TLSv1.2	35.186.224.26	192.168.68.119	306.458351	193
	0x018	Application Data	93	TLSv1.2	35.186.224.26	192.168.68.119	306.458380	194

וכעת, לאחר שסיננו את ההקלטה לחבילות הרלוונטיות, כעת נוכל לייצא את המידע לקובץ CSV מתוך ה-Wireshark .