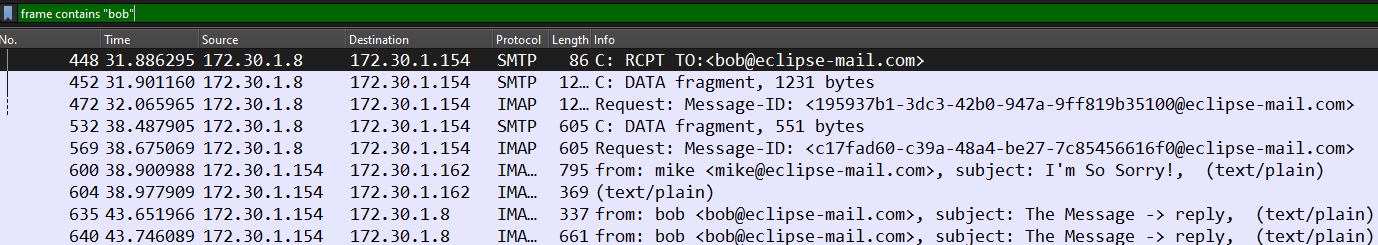
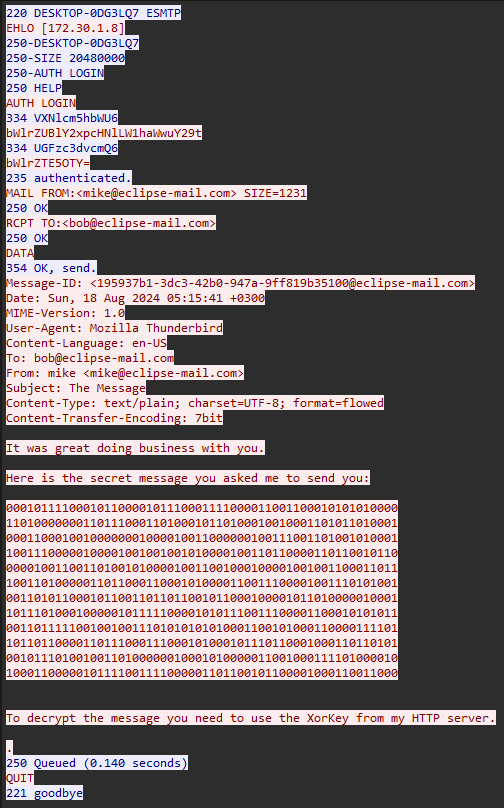
**פתרון ה- CTF**

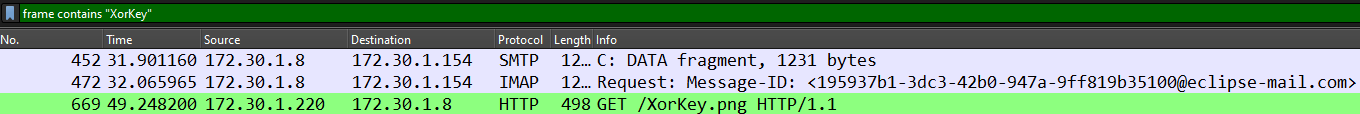
* קריאת מכתב הפתיחה והוצאת מידע חשוב:
  + הארגון – Eclipse
  + האקר – bob
  + הסנפת Wireshark של הארגון – מהיומיים האחרונים
* הוצאת מידע מההסנפה:

שימוש בפילטר **frame contains "bob"**

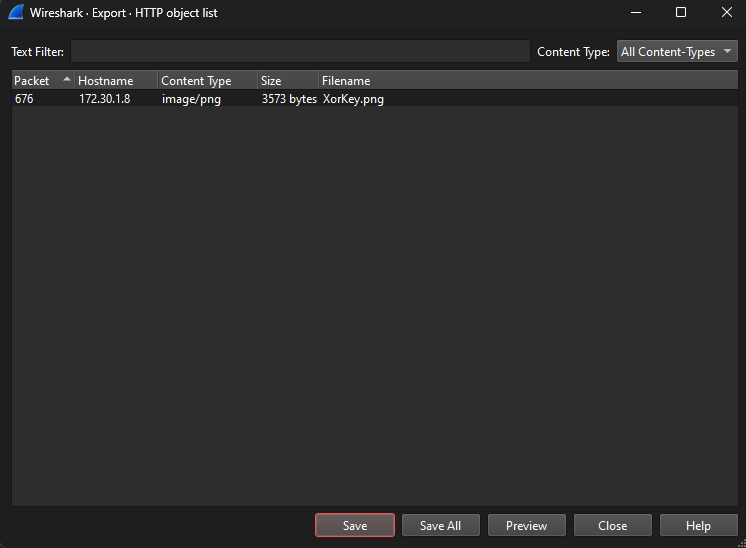
ביצוע **Follw TCP Stream** על החבילה הראשונה

אם נקרא את ההודעה (שמגיעה אחרי 354 OK, send.) נבין ש- Mike שולח מסר מוצפן בביטים ל- bob.

אפשר לראות שבסוף ההודעה Mike מציין איך לפענח את ההודעה – בעזרת XorKey.

נבצע סינון **frame contains XorKey** ונקבל חבילות HTTP,

נוכל לראות שזאת בקשת HTTP לתמונה.

נבצע **File 🡪 Export Objects 🡪 HTTP** נבחר בתמונה ונשמור אותה באמצעות **save**.

נפתח את התמונה ששמרנו, במקום ששמרנו, ונוכל לגלות את המספר **32767**

התמונה נקרת בשם XorKey. נעשה אחד ועוד אחד...

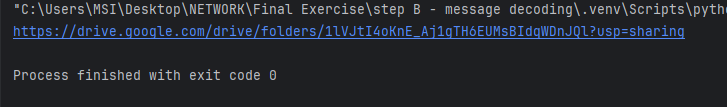
פיענוח המסר על ידי במספר (= המפתח) בעזרת שיטת Xor, כמו בתרגיל בית 6 למי שזוכר ומי שלא בעיה שלו! סתם. אפשר לעשות חיפוש בגוגל ולראות איך זה עובד.

* פיענוח המספר:

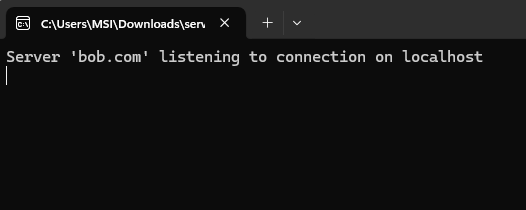
נעתיק את המסר והמפתח וניצר את הקוד הבא

def symmetric\_encryption(input\_data, key):  
 # Determine chunk and key size based on length of data (even or odd)  
 if len(input\_data) % 2 == 0:  
 chunk\_size = 16  
 # Using the all bits of key  
 key\_in\_bits = format(key, '016b')  
 else:  
 chunk\_size = 8  
 # Using the last 8 bits of the key  
 key\_in\_bits = format(key, '016b')[-8:]  
  
 # Split data into chunks of appropriate size  
 chunks = [input\_data[i:i + chunk\_size] for i in range(0, len(input\_data), chunk\_size)]  
  
 # Perform XOR operation on each chunk with the key  
 result = ""  
 for chunk in chunks:  
 for bit, key\_bit in zip(chunk, key\_in\_bits):  
 result += '1' if bit != key\_bit else '0'  
  
 return result  
  
  
Encrypted\_message = '000101111000101100001011100011110000110011000101010100001101000000011011100011010001011010001001000110101101000100011000100100000001000010011000000100111001101001010001100111000001000010010010010100001001101100001101100101100000100110011010010100001001100100010000100100110001101110011010000011011000110001010000110011100001001110101001001101011000101100110110110010110001000010110100000100011011101000100000101111100001010111001110000110001010101100110111110010010011101010101010001100101000110000111101101101100001101110001110001010001011101100010001101101010010111010010011010000001000101000001100100011110100001010001100000101111001111000001101100101100001000110011000'  
  
XorKey = 32767  
  
Decoded\_message = symmetric\_encryption(Encrypted\_message, XorKey)  
  
message = ''.join(chr(int(Decoded\_message[i:i + 8], 2)) for i in range(0, len(Decoded\_message), 8))  
  
print(message)

נריץ ונקבל את קישור הבא

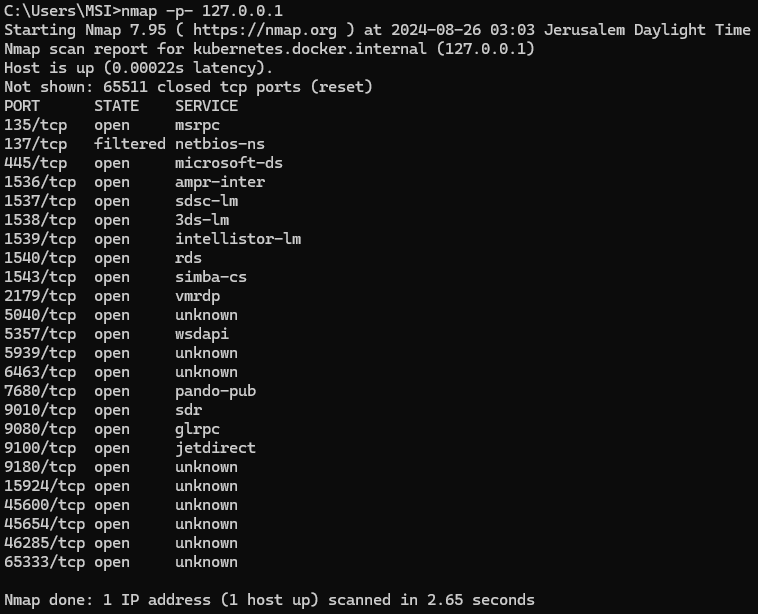


* הרצת השרת:

נוריד את השרת ונריץ אותו

השרת רץ. אנחנו מספיק חכמים ויודעים שעלינו להתחבר לשרת בעזרת קוד לקוח. אך יש בעיה!

אנחנו לא יודעים מה הפורט שהשרת משתמש.

לכן נריץ את פקודת Nmap וככה נמצא את הפורט

עכשיו שקיבלנו את כל האפשרויות האלו, ננסה אחד אחד. נכון שקיבלנו יחסית הרבה תוצאות אבל זה עדיף מאשר לבדוק את כל ה- 65,535 פורטים הקיימים.

נכתוב קוד שרק מתחבר לשרת וננסה כל פעם פורט שונה. ספוילר, הפורט הוא 46,285.

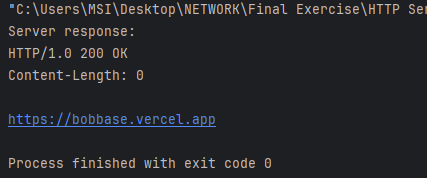
כמובן ש- אנחנו חכמים, ולכן הדבר הראשון שנחשוב לעשות הוא לבקש **עמוד בררת מחדל '/'**.

נתכנת את שאר קוד הלקוח שנשלח מספר הודעות עד שנשלח את ההודעה המתאימה.

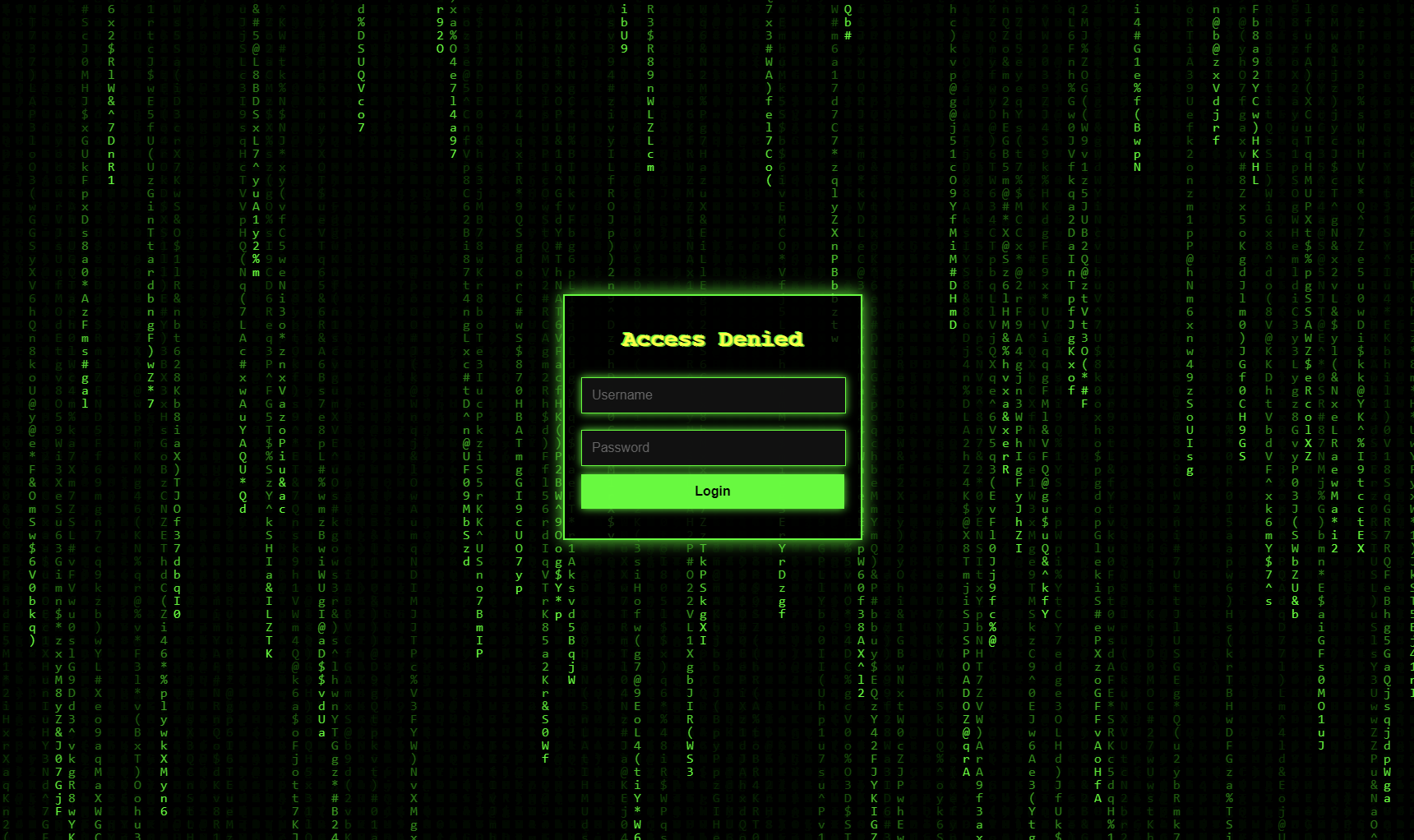
כמובן שהשרת מתקן ומכוון עבור גל טעות. קוד הלקוח המתקבל

import socket  
  
IP = '127.0.0.1'  
PORT = 46285  
  
def main():  
 client\_socket = socket.socket(socket.AF\_INET, socket.SOCK\_STREAM)  
 client\_socket.connect((IP, PORT))  
  
 request = (  
 "GET bob/website/link HTTP/1.0\r\n"  
 "Host: bob.com\r\n"  
 "\r\n"  
 )  
 client\_socket.send(request.encode())  
  
 response = client\_socket.recv(1024).decode()  
 print("Server response:")  
 print(response)  
  
 client\_socket.close()  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()

עבור קוד לקוח זה, נקבל הודעת תגובה 200 ok מהשרת עם תוכן. התוכן ההודעה קישור לאתר



* כניסה לאתר:

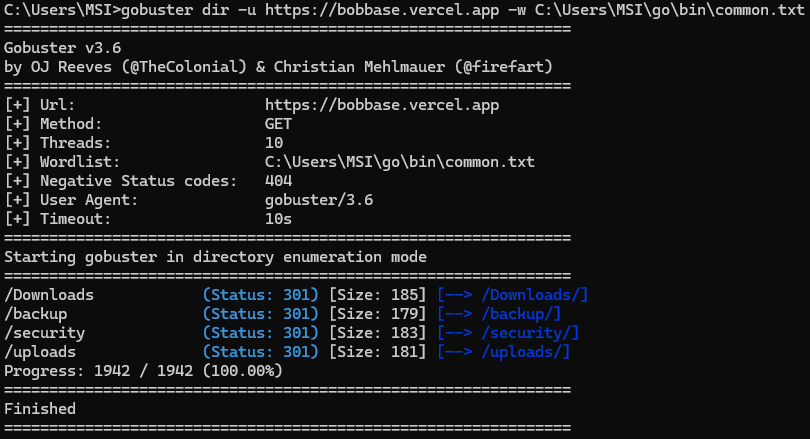
כאשר נכנס לאתר נתקל בבקשת התחברות!

לעבור את השלב הזה בלי מעט ידע על כלי סייבר – בלתי אפשרי!

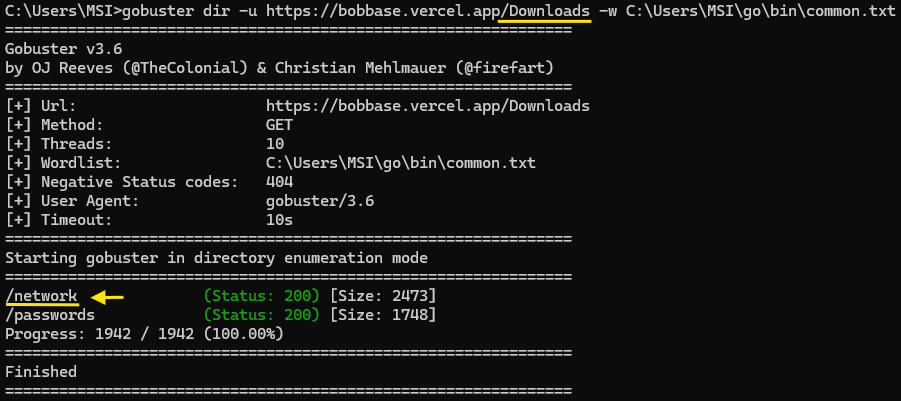
המטרה היא למצוא אתרים נתיבים מוסתרים אך קיימים שיכולים להדליף לנו מידע.

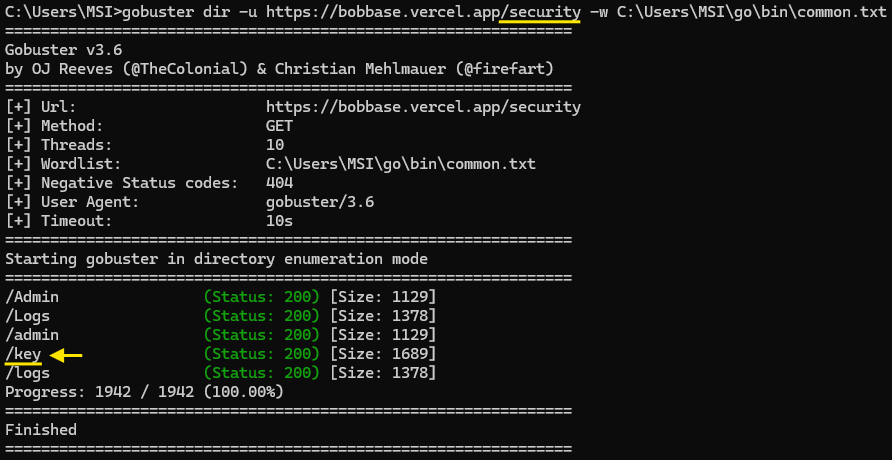
כדי למצוא את אותם נתיבים, נשתמש ב- Gobaster המשתמש ב- brute force למציאת הנתיבים.

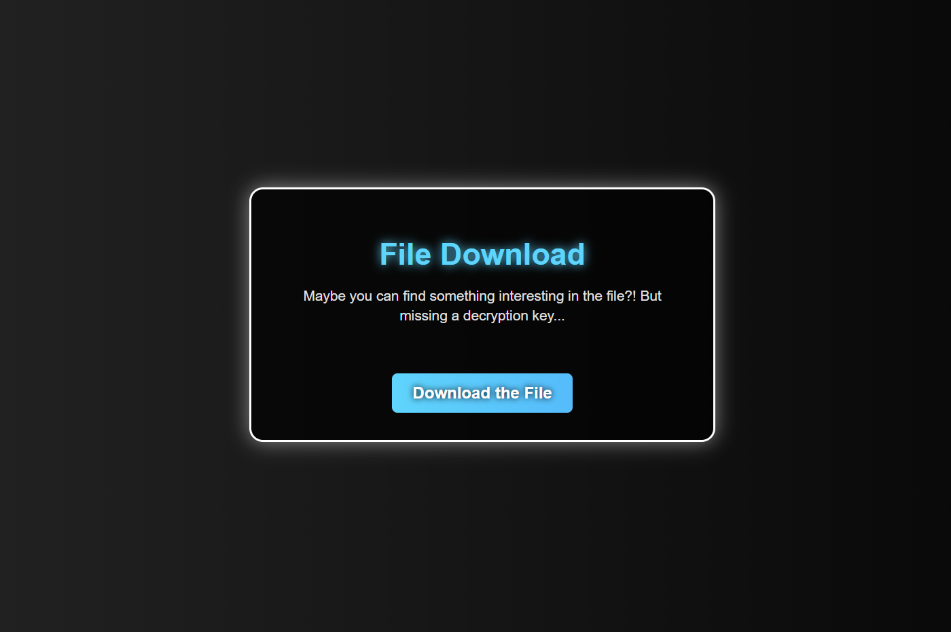
לרוב האנשים אין את הפקודה במחשב ולכן נוריד אותה (הוראות אפשר למצוא בגוגל).

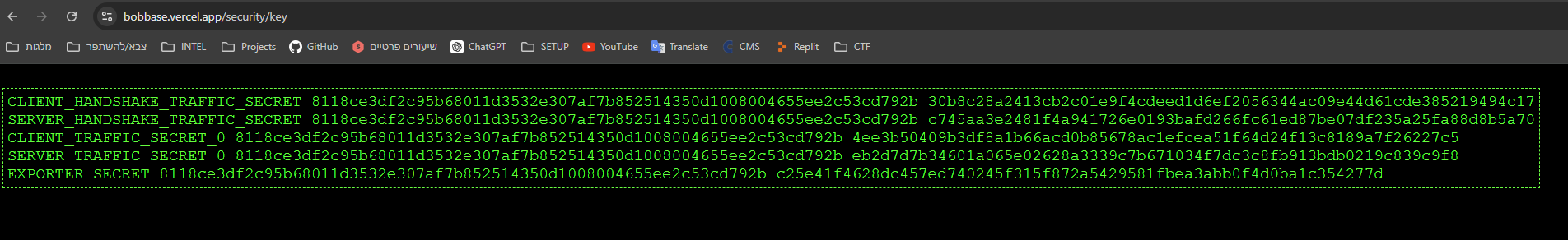
נריץ את הפקודה על כתובת האתר

קיבלנו 4 תוצאות. נחקור כל אחת ונוכל למצוא עוד המשכי נתיבים. אך נוכל למצוא דברים מעניינים רק בנתיבים הבאים:

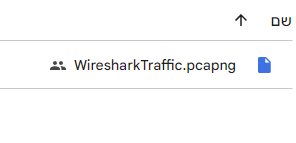
**/Downloads/network**

**/security/key**

נציב את הנתיבים ל- URL של האתר ונקבל את העמודים האלו



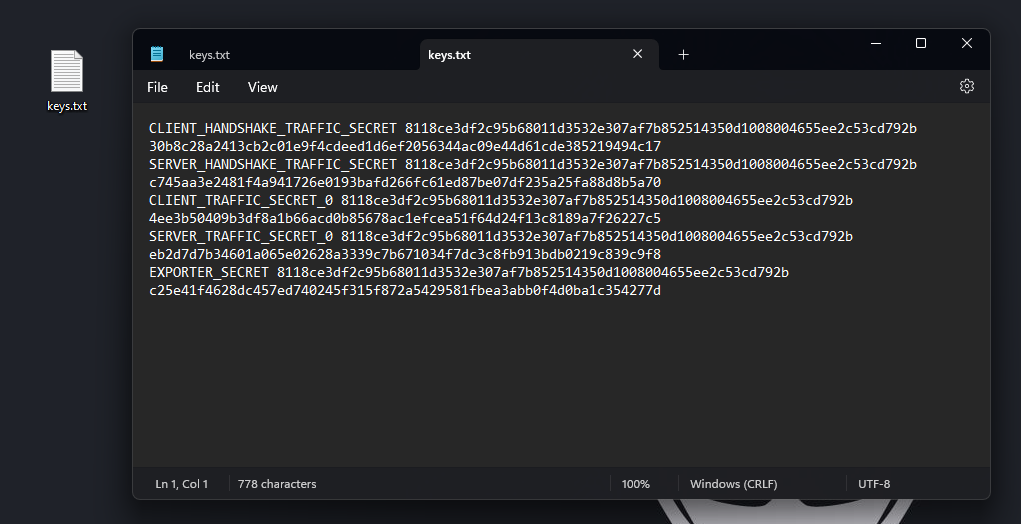
העמוד הראשון מכיל כפתור. אם נלחץ עליו, נגש ל- Google Drive עם קובץ הסנפה.

העמוד השני מכיל מפתחות ו- Client Random.

* פתיחה ופיענוח ההסנפה:

נבצע שוב פעם אחד ועוד אחד...

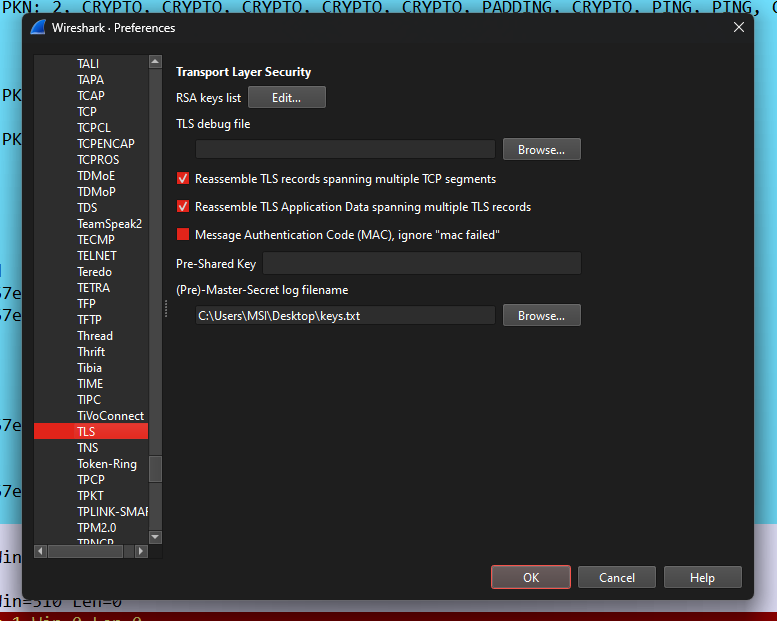
נשתמש במפתחות כדי לפענח את ההסנפה!

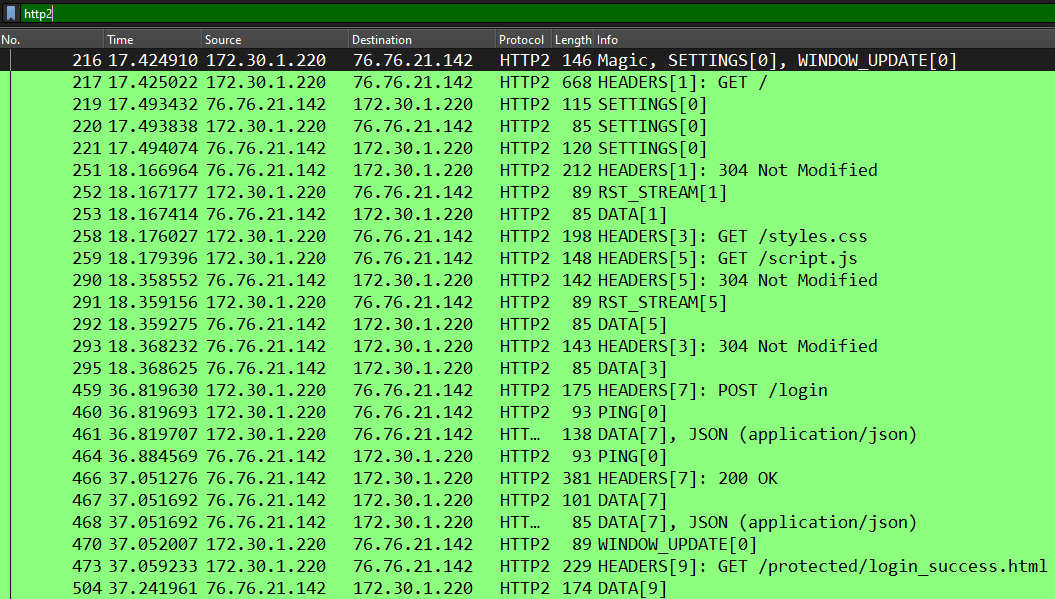
ניצור קובץ txt ונשמור בו את המפתחות

נפתח את ההסנפה נפעיל את הפיענוח

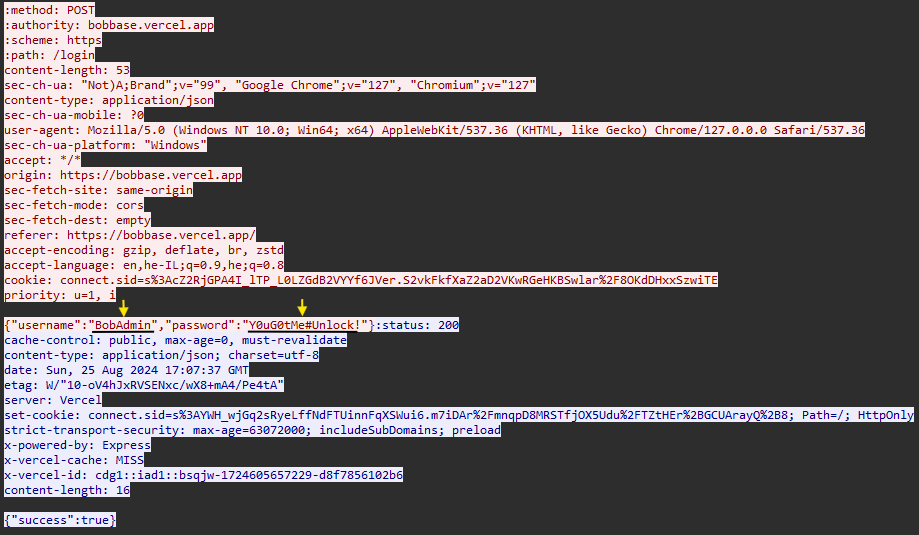
**Edit 🡪 Preferences 🡪 Protocols 🡪 TLS**

נכניס את הנתיב בו שמרנו את קובץ המפתחות



נבצע את הפילטר **http2** ונראה את כל החבילות הצהובות שפוענחו

אפשר לראות שיש חבילה המבצעת פעולת **POST /login** ולכן היא הופכת לחבילה למאוד מעניינת.

נבצע עליה **Follow HTTP/2 Stream** ונקבל את החלונית הבאה

וכך מצאנו את שם המשתמש והסיסמא שהוסנפו כאשר bob לא שם לב והתחבר לאתר, ולכן כך יכולנו למצוא אותם.

התחבר בעזרת שם המשתמש והסיסמא ונקבל את עמוד ההתחברות עם כפתור להורדת הקובץ הגנוב!

