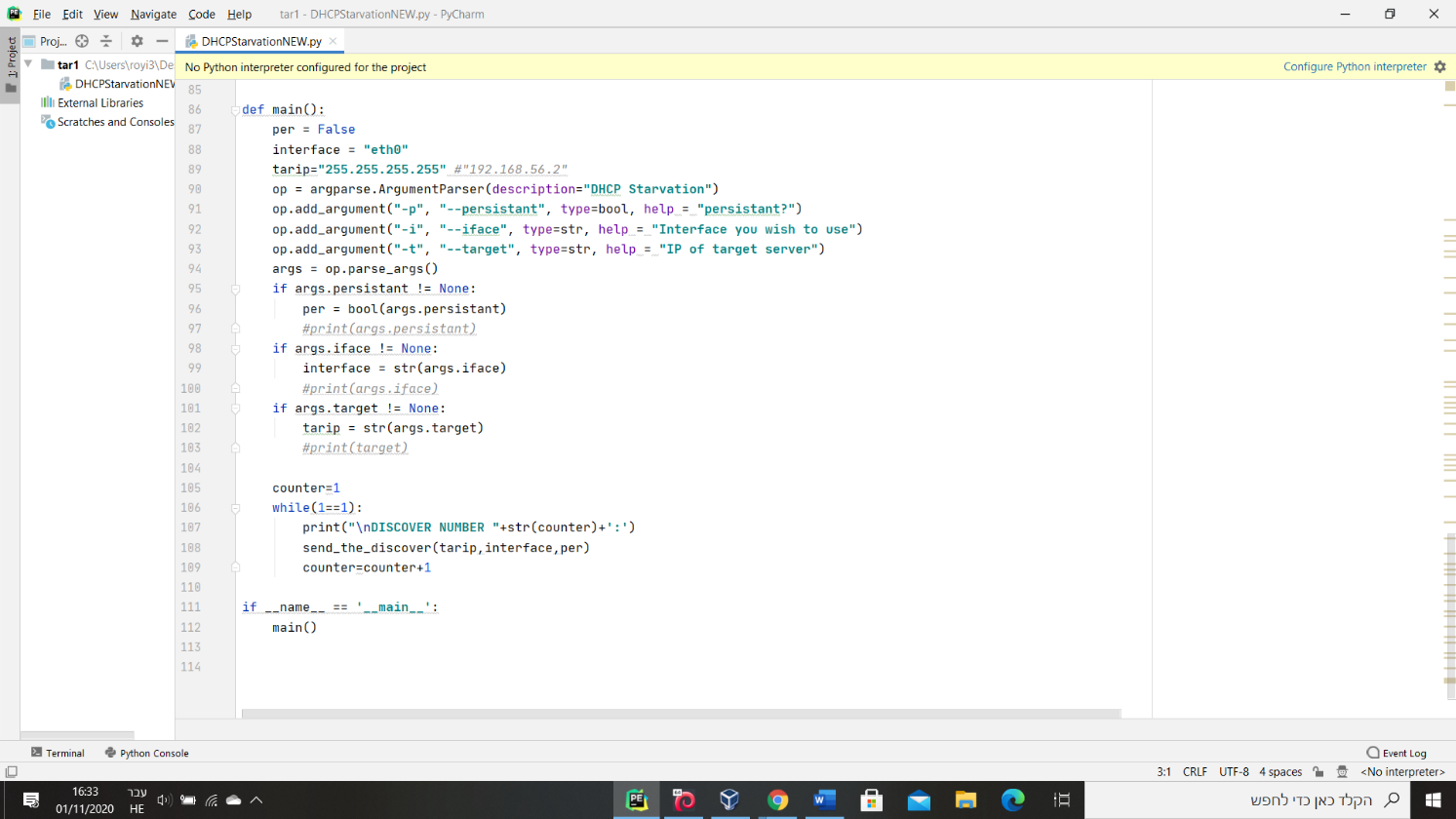
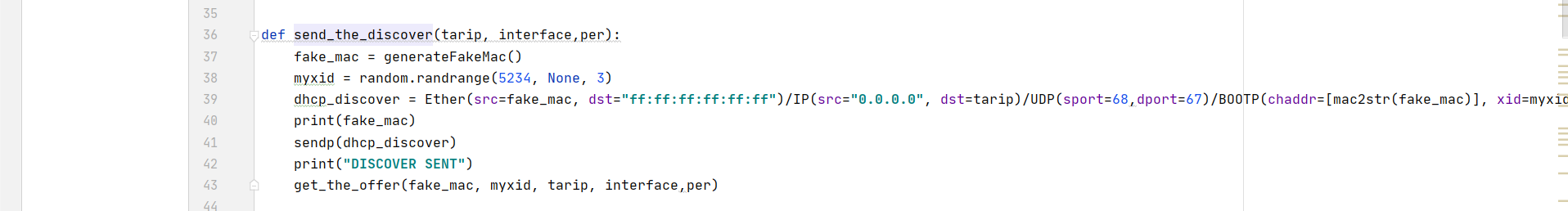
יסודות לאבטחת רשתות



(כל צילומי המסך המצורפים נעשו לפני תיעוד הקוד! ! ! ישנו תיעוד בנוסף גם בקובץ הקוד...)

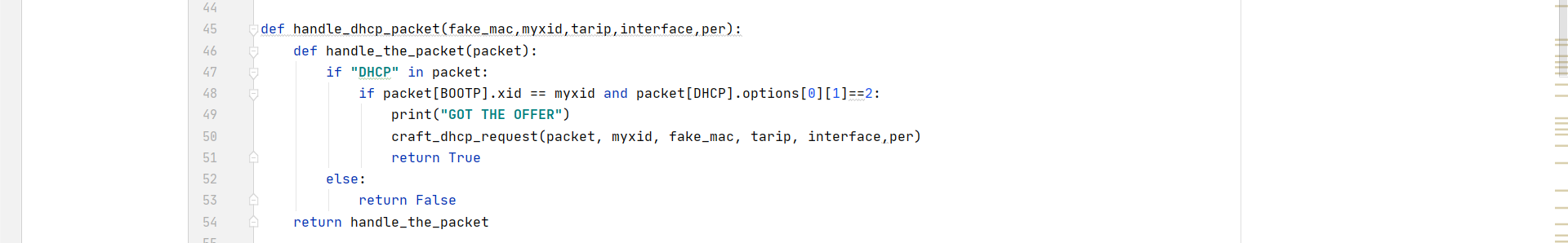
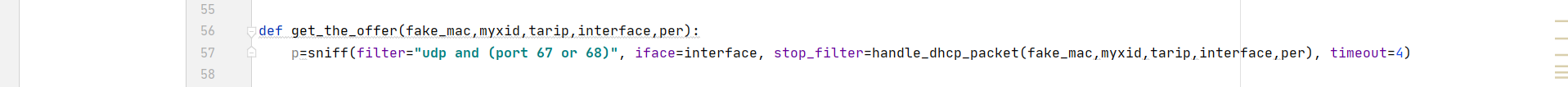


כאן ניתן לראות את פונקציית ה main (בשורה 86) אשר מפעילה את כל התוכנית שלנו, ראשית כל מפעילים את הפונקציה send\_the\_discover (בשורה 108) על מנת לשלוח discover, אשר מקבל את הייעד לתקיפה (tarip), את הממשק (interface), ואת משתנה העקביות (per) שבעזרתו ממומש הבונוס של התרגיל.

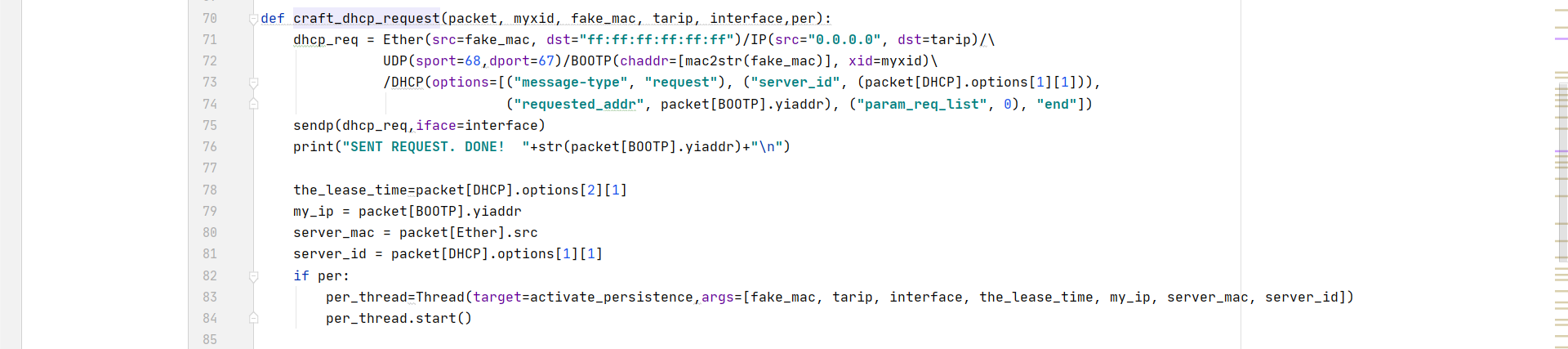


בתוך הפונקציה send\_the\_discover, אנו יוצרים כתובת MAC מזוייפת באמצעות הפונקצייה generateFakeMac בשורה 37 – אשר יוצרת באקראיות כתובת MAC כמו התבנית של כתובת MAC.

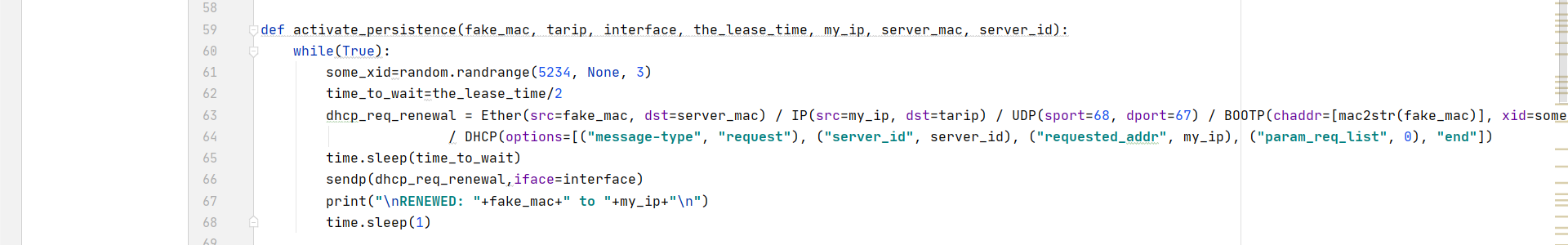
יוצרים באקראיות מספר מזהה (myxid), ויוצרים פאקטה של dhcp (dhcp\_discover), שולחים אותו ואז מנתרים את ההצעה באמצעות הפונקצייה get\_the\_offer (בשורה 43), אשר מקבלת את הכתובת MAC המזוייפת (fake\_mac), את המספר המזהה (myxid),את הייעד (tarip), את הממשק (interface), ואת משתנה העקביות (per).

ב – get\_the\_offer מתבצעת הסנפה ביחד עם סינון עבור הפורטים המתאימים, כלומר 67 ו 68, וכל עוד hundle\_dhcp\_packet (בשורה 57) הוא true אזי נמשיך ונמשיך להסניף.

בתוך הפונקצייה hundle\_dhcp\_packet נעשה פונקצייה מקוננת - hundle\_the\_packet (שורה 46) שבתוכה נבצע בדיקה בפקאטות של ההסנפה, האם מדובר ב dhcp, ואם כן נוודא את מספר המזהה (myxid == packet[BOOTP].xid), על מנת לוודא שזה הפאקטות שלנו, כי אחרת אין לנו שום צורך בהם, ואז נפעיל את הפונקצייה – craft\_dhcp\_request (שורה 58).



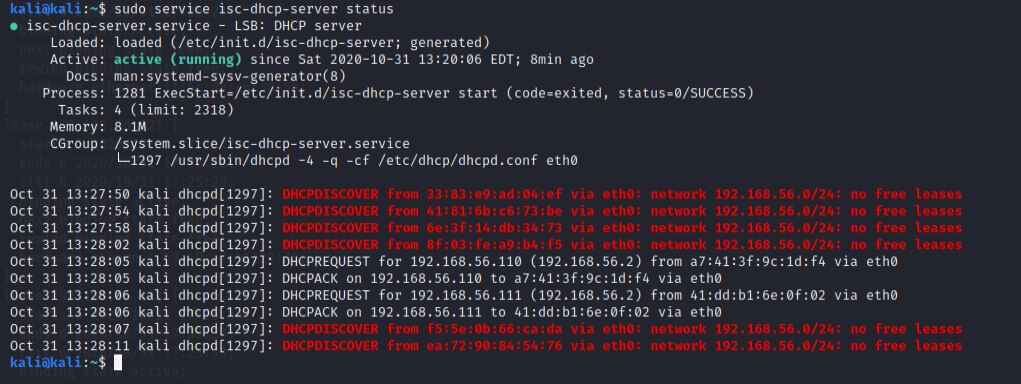
כעת ב - craft\_dhcp\_request אנחנו מתפעלים דרך לבקש request מה dhcp server, ובעצם גם מתחזקים את ההמשכיות של הבקשות על מנת שימשיך להיות חוסר בכתובות פעילות אצל ה dchp server, כלומר שבעצם פה גם ניכנס העניין של הבונוס, שמתבצע באמצעות thread, וממשיך לעדכן באופן קבוע את הכתובת כך שלא יאבד את הכתובות, ואז הם בעצם יהיו פנויים לאחרים לאחר כמה זמן...  
דבר זה מתבצע בעזרת הפונקצייה activate\_persistence, (בשורה 83), שמקבל כערכים את כתובת ה MAC המזוייפת, ייעד התקיפה (tarip), interface, זמן השיחרור של ה IP (the\_lease\_time), את הכתובת של הserver , והמזהה שלו.



בתוך הפונקצייה activate\_persistence, מתבצעת לולאה אין סופית (שורה 68), עבור ה thread, שבעצם מתחזק את הקשר עם השרת בכל מחצית זמן מהזמן שמוגדר לקשר הפועל, ומונע מהשרת להיות פנוי עבור אחרים.

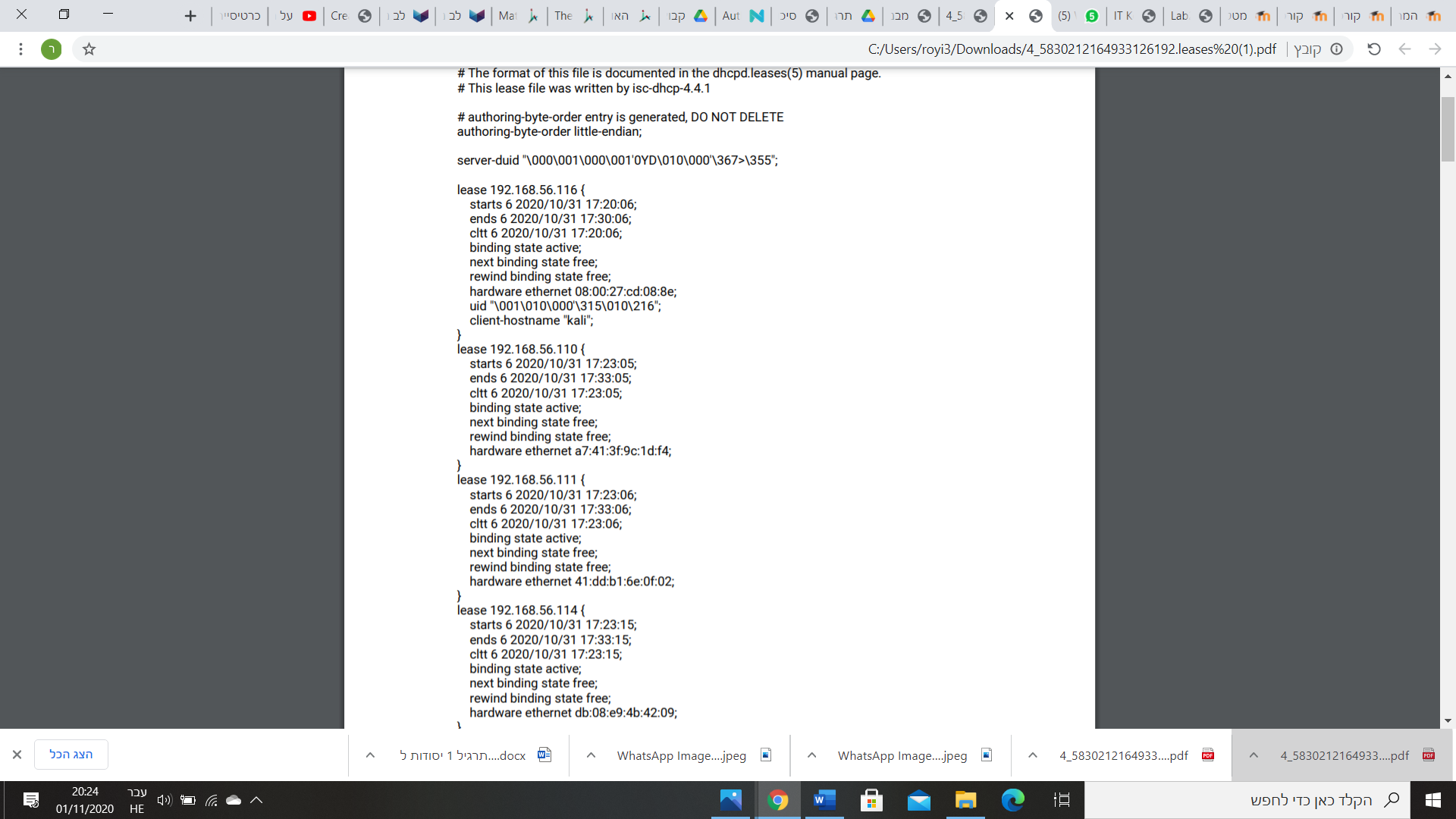
עכשיו מהצד השני – ניבדוק האם אכן מהצד של השרת dhcp ההתקפה אכן עובדת.

הטווח כתובות שלנו הוא: Range 192.168.56.110 - 192.168.56.122 כלומר 12 כתובות.



ניתן לראות שאכן ההתקפה עובדת, כלומר שכאשר אנו ניגשים לטרמינל של השרת dhcp, ומריצים את הסטטוס שלו, ניראה כי כל הטווח שלו תפוס (כלומר באדום), ישנם discover שאין עבורם free lease אבל עבור dhcp request של תחזוק הקשר שכבר קיים, הם כן מתאפשרים (כלומר בלבן).

בקובץ ה Lease של השרת dhcp ניתן לראות את התמונה הבאה:



בבלוק הראשון ניתן לראות את התהליך האמיתי של ה dhcp עם המחשב, ובכל השאר את התוכנית שבעצם לוקחת מהשרת את כל הכתובות שלו.

הסוף !