טכנו"צ סייבר – תרגיל 4 - "SPOOF"

ניצול חולשות רשת

מטרת העל

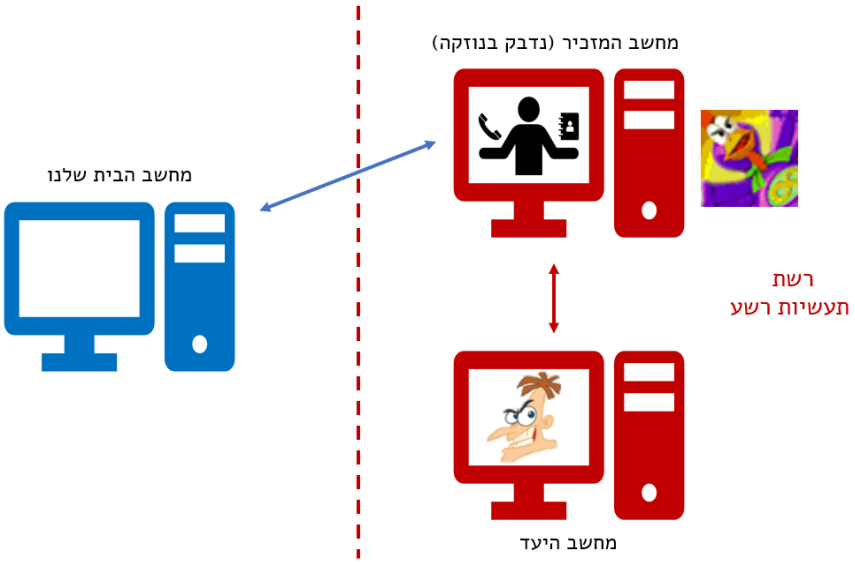
בתרגיל 3 ביססנו את היכולת להריץ פיקודים על גבי הנוזקה שלנו. בתרגיל הזה אנחנו נבנה לנוזקה שלנו פיקוד חדש, שמטרתו להערים על מחשב אחר ברשת ולנתב אותו לדף אינטרנט משלנו באמצעות **DNS Spoofing**.

שימו לב: תרגיל זה יעשה ב**זוגות**. בנוסף קראו לעומק את ההוראות של התרגיל ואת האזהרות בסופו. שימו לב גם שזה תרגיל בלי הרבה קוד אבל עם **debugging** לא פשוט בכלל וכדאי להתחיל אותו מוקדם.

את התרגיל עליכם להריץ ברשת שניצור בין שתי מכונות וירטואליות, ובסביבה כזאת התרגיל גם ייבדק.

רקע

החלטנו להשתמש בנוזקה שלכם כדי לתקוף את ארגון הרשע והפשע "דופנשמירץ תעשיות רשע בע"מ". ברצוננו לגלות את שם המשתמש והסיסמא של דופנשמירץ לכתובת המייל שלו. לצערנו הרב אין לנו שום גישה ישירה למחשב שלו - לא פיזית ולא תקשורתית. עם זאת, לאחר מאמצי שכנוע מרובים הצלחנו למכור למזכיר של דופנשמירץ עותק של המשחק **Chicken Invaders** שבתוכו מסתתרת הנוזקה שלכם. הנוזקה מאפשרת לנו נגישות תקשורתית למחשב המזכיר, ומשם אל תוך הרשת של הארגון.



דופנשמירץ לא משתמש במחשב של המזכיר על כן אין לנו מה לחפש על המחשב עצמו, אבל מחשב המזכיר מאפשר לנו גישה אל הרשת הפנימית של הארגון. נרצה למנף גישה זאת על מנת לרמות את דופנשמירץ לתת לנו את הסיסמא שלו.

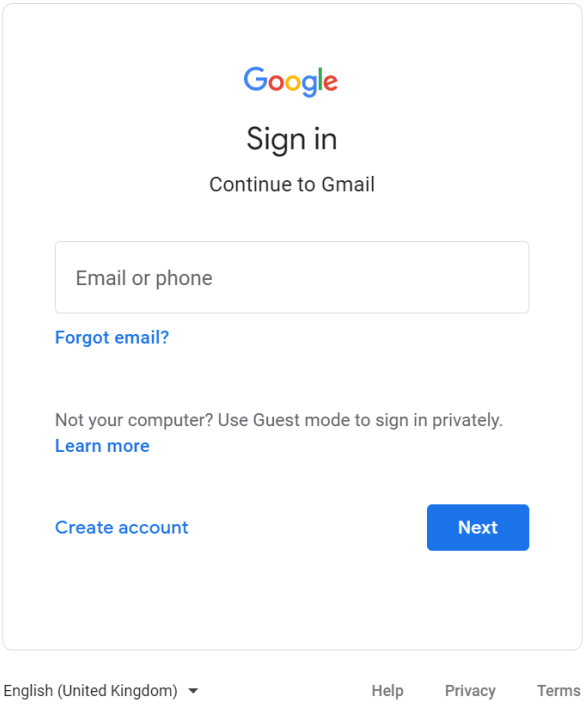
טכנו"צ סייבר – תרגיל 4 - "SPOOF"

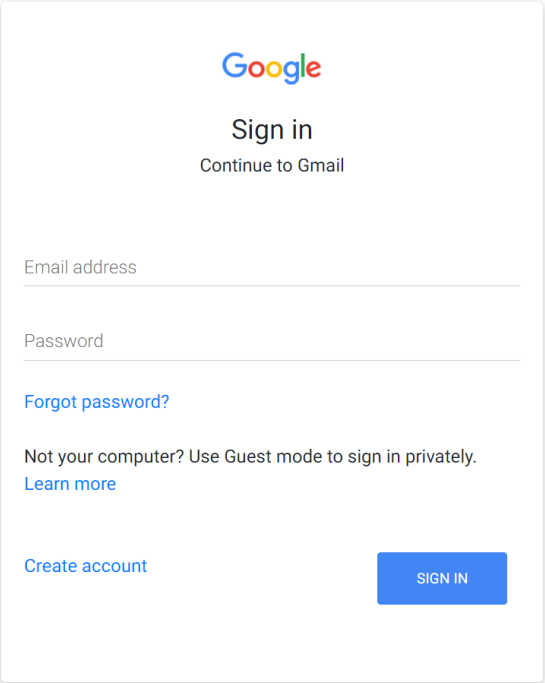
השיטה

הגיעה אלינו ידיעה מודיעית שדופנשמירץ משתמש באתר **GMAIL**. על בסיס יום-יומי על מנת לקרוא ולכתוב מיילים. כתובת האתר היא **mail.google.com**, אבל דופנשמירץ המגלומן ביקש מאנשי ה-**IT** שלו להגדיר את שרתי ה-**DNS** ברשת כך שכשהוא ניגש מהדפדפן שלו ל-**mail.doofle.com** כתובת ה-**IP** שהוא מקבל היא של **mail.google.com** ולכן הוא רואה את אתר **GMAIL** החביב עליו.

כאמור, מחשב המזכיר מאפשר לנו להיכנס לרשת הפנימית. באמצעות גישה זאת נבצע תהליך שדומה למה שראיתם בהרצאה: נרמה את דופנשמירץ לחשוב שמחשב המזכיר הוא ה**DNS Server** של הארגון, באמצעות ניצול פרוטוקול **ARP**. בשלב זה תהיה לנו גישה ישירה לתעבורת ה **DNS** שברשת, אותה ננצל כדי להערים על מחשב היעד. ברגע שהוא יפנה ל-**mail.doofle.com**, נפנה אותו ל**IP** של שרת משלנו, שמציג דף שנראה כמו דף ההתחברות של **GMAIL**.

השרת הזה מציג דף שנראה כמו דף ההתחברות של **GMAIL** עד כדי הבדל קטן - הוא למעשה מפוברק לחלוטין. כשדופנשמירץ יכניס את פרטיו, השרת הזה שומר בצד את הסיסמא שלו, ואז מנתב אותו ל-**mail.google.com**, אתר האמיתי אליו ניסה להגיע! לדופנשמירץ זה ייראה כמו התחברות רגילה לחלוטין ל-**GMAIL**, אבל כמובן שזה לא המצב. לטכניקה הזו קוראים דיוג (**Phishing**) ונדבר עליה עוד בהמשך הקורס.





טכנו"צ סייבר – תרגיל 4 - "SPOOF"

מימין - חלון ההתחברות ל-**GMAIL** המזוייף שעל השרת שלנו. משמאל - חלון ההתחברות ל-**GMAIL** האמיתי.

שלבי התרגיל

שלב ראשון - התחברות לאתר

בתור התחלה, נא וודאו שהשרת הזדוני שלנו עובד לכם. קראו את החלק "חומרי עזר" והריצו את השרת שלכם. ריצה מוצלחת אמורה להיראות בערך כך:

```
>python app.py
 * Serving Flask app 'app' (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on all addresses (0.0.0.0)
   WARNING: This is a development server. Do not use it in a production deployment.
 * Running on http://127.0.0.1:80
 * Running on http://132.64.245.149:80 (Press CTRL+C to quit)
```

התחברו לאינטרנט בכתובת "127.0.0.1" (הכתובת המקומית) ובדקו שאתם מגיעים לדף ההתחברות המזויף של השרת. בידקו שהפרטים שאתם מכניסים מתועדים בקובץ **passwords.txt** ושאתם מקבלים ניתוב מחדש ל-**Gmail**.

לאחר מכן, נסו להיכנס לאתר באמצעות המכונה הוירטואלית השנייה (ניתן למצוא את כתובת ה **IP** של המחשב עם השרת באמצעות **ip addr**).

שלב שני - קבועים

כעת ניגש לכתוב את קוד התקיפה שלנו. בתרחיש המבצעי הקוד שאנחנו כותבים עכשיו נשלח ממחשב התוקף אל מחשב המזכיר באמצעות הנוזקה שכתבתם, והכל מודלף בחזרה לתוקף. כדי לא להסתבך נניח שאנחנו כותבים ומריצים קוד ישירות על מחשב המזכיר, מבלי להשתמש בנוזקה שכתבתם.

בחומרי העזר מסופק לכם שלד לתרגיל. בתור התחלה עליכם למלא את הקבועים החסרים בקוד:

```python
DOOFENSHMIRTZ_IP = "???"  # Enter the computer you attack's IP.
SECRATERY_IP = "???"  # Enter the attacker's IP.
NETWORK_DNS_SERVER_IP = "???"  # Enter the network's DNS server's IP.
SPOOF_SLEEP_TIME = 2

IFACE = "???"  # Enter the network interface you work on.

FAKE_GMAIL_IP = SECRATERY_IP  # The ip on which we run
DNS_FILTER = f"udp port 53 and ip src {DOOFENSHMIRTZ_IP} and ip dst {NETWORK_DNS_SERVER_IP}"  # Scapy filter
REAL_DNS_SERVER_IP = "8.8.8.8"  # The server we use to get real DNS responses.
SPOOF_DICT = {  # This dictionary tells us which host names our DNS server needs to fake, and which ips should it give.
    b"???": FAKE_GMAIL_IP
}
```

טכנו"צ סייבר – תרגיל 4 - **"SPOOF"**

הסבר על הקבועים:

DOOFENSHMIRTZ_IP: ה-**IP** של המחשב אותו אתם הולכים לתקוף.

SECRATERY_IP: ה-**IP** של מחשב המזכיר. למעשה זה ה-**IP** של המחשב שהולך להריץ את הקוד הזה. מאותחל להיות ה-**IP** של המחשב שמריץ את הקוד (127.0.0.1). חשוב שמרגע זה ואילך כשאתם מריצים את הקוד שלכם תעבדו בשתי מכונות וירטואליות שונות, כלומר אסור ש-**DOOFENSHMIRTZ** גם הוא יהיה ה**IP** שלכם (אחרת דברים לא יעבדו, מוזמנים לחשוב למה).

NETWORK_DNS_SERVER_IP: ה-**IP** של שרת ה-**DNS** המקומי, שאליו פונה המחשב של דופנשמירץ על בסיס קבוע. זה בעצם שרת ה-**DNS** הדיפולטיבי של המכונה הוירטואלית, אליו אנחנו הולכים להתחזות. כתוב בנספח כיצד למצוא אותו.

IFACE: כפי שראיתם ב-**Wireshark**, ישנם מספר "כרטיסי רשת" למחשב. קבוע זה הינו כרטיס הרשת בו הקוד שלנו הולך להשתמש (יהיה שימושי בהמשך). מוזמנים לגגל: "**How to list network interfaces**". בזמן **debugging** שימו לב במיוחד לכרטיס הרשת שאתם עובדים איתו, כיוון שלמשל אי תיאום בין כרטיס הרשת שאתם עושים עליו **spoof** וכרטיס הרשת שאתם מסניפים עליו עם **Wireshark** כדי לדבג (לבין כרטיס הרשת שהוא זה שמחבר אתכם עם המחשב של השותף) יכול מאוד לבלבל אתכם וחבל.

SPOOF_SLEEP_TIME: הזמן בין שתי תשובות **ARP** שיישלחו ממחשב המזכיר.

FAKE_GMAIL_IP: הכתובת שבה שוכן השרת המזויף שנראה כמו **GMAIL**. בתרגיל שלנו גם זה יהיה על מחשב המזכיר.

DNS_FILTER: בהמשך נגדיר לספריה ששמה **scapy** (כמו **wireshark** בתוך פייתון) לאיזה הודעות לצותת. נפלטר רק בקשות **DNS** שמגיעות למחשב המזכיר מהמחשב של דופנשמירץ.

REAL_DNS_SERVER_IP: שרת ה-**DNS** שממנו נעתיק תשובות להודעות שאנחנו לא רוצים לזייף. מאותחל להיות השרת של **Google**.

SPOOF_DICT: מילון שאומר לנו את מה לזייף ואיך. המפתחות של המילון הם **host names** והערכים הם כתובות **IP** שהיינו רוצים לזייף עבור ה-**host names**.

ודאו שאתם מבינים מה ההבדל בין קבועים שונים ומה בדיוק המשמעות של כל אחד מהם.

טכנו"צ סייבר – תרגיל 4 - **"SPOOF"**

שלב שלישי - מימוש מחלקות

כעת, עליכם לממש שתי מחלקות:

1) **ArpSpoofer**: מחלקה זו אחראית על השלב הראשון של ההונאה: שליחת פקטות **ARP** שמשכנעות את המחשב של דופנשמירץ שמחשב המזכיר הוא שרת ה-**DNS** של הארגון:

a) פונקציית איתחול (**__init__**) ממומשת עבורכם. מקבלת את ה-**IP** אותו נרצה לתקוף, ואת ה**IP** אליו אנחנו רוצים להתחזות.

b) **get_target_mac**: פונקציה זו שולחת בקשת **ARP** על מנת לקבל את כתובת ה**MAC** של המחשב אותו אנחנו רוצים לתקוף. בנוסף על להחזיר את כתובת ה-**MAC**, על פונקציה זו לשמור את ה-**MAC** בתוך **target_mac**, אחד השדות של המחלקה. (מומלץ בשלב זה לבדוק שהקוד של הפונקציה באמת מחזיר את **target_mac**).

c) **spoof**: פונקציה זו שולחת תשובת **ARP** אל המטרה שלנו, בה מציגה אותנו בתור **spoof_ip** (שדה של המחלקה שמתאר את ה**IP** אליו אנחנו מתחזים). (מומלץ לבדוק בשלב זה שמחשב היעד מקבל את הפאקטות **ARP** ושבאמת תעבורת ה **DNS** עוברת לתוקף)

d) **run**: הלולאה הראשית של הפונקציה. ממומשת עבורכם. פונקציה זו קוראת ל-**spoof** באינטרוולים קבועים.

e) **start**: פונקציה זו מריצה את הפונקציה **run** ב-**Process** נפרד - כלומר במקביל לקוד הרגיל שאתם מריצים. ממומשת עבורכם.

טכנו"צ סייבר – תרגיל 4 - "SPOOF"

2) **DnsHandler**: מחלקה זו היא למעשה מימוש של שרת **DNS** מקומי. שרת זה מצותת לתעבורת **DNS**. השרת בונה תשובת **DNS** רגילה למעט בקשות על **Domains** ספציפים, אותן מפנה ל-**IP** של השרתים המזויפים שלנו. על מנת לדעת מה ה**IP** האמיתי של כתובת מסויימת שרת ה-**DNS** שלנו יפנה לשרת **DNS** אמיתי ו-"יעתיק" את התשובה ממנו. מטרת מחלקה זו היא בעצם לממש את החלק השני של התוכנית הזדונית שלנו: לאחר שמחשב היעד שוכנע ע"י **ArpSpoofer** שכתובת ה **IP** של המחשב הזדוני היא, על המחשב הזדוני להתנהג כמו שרת **DNS**, אחרת ההונאה תיכשל.
המחלקה מכילה את הפונקציות הבאות:

a) **run**: הלולאה הראשית של הפונקציה. ממומשת עבורכם. פונקציה זו מריצה את ה-**sniffing** להודעות **DNS** ומספקת להן מענה באמצעות הפונקציה **resolve_packet**.

b) **start**: פונקציה זו מריצה את הפונקציה **run** ב-**Process** נפרד - כלומר במקביל לקוד הרגיל שאתם מריצים. ממומשת עבורכם.

c) **get_real_dns_response**: פונקציה זו מקבלת בקשת **DNS**, מעבירה אותה לרונן שקל ומחזירה את תשובת ה-**DNS** שרונן שקל נתן. לפני שמחזירים את התשובה יש לשנות את **IP** השולח מה-**IP** של רונן שקל ל-**IP** של המחשב שלנו. (מומלץ לבדוק שאתם מצליחים באמת לקבל תשובת **DNS** אמיתית)

d) **get_spoofed_dns_response**: פונקציה זו מקבלת בקשת **DNS** וכתובת **IP**. על הפונקציה לשלוח תשובת **DNS** שתגרום ללקוח להאמין שהכתובת שהוא מחפש הוא ה-**IP** שהפונקציה קיבלה. נשתמש בפונקציה זו כדי לזייף תשובות **DNS** ל-**GMAIL**.

e) **resolve_packet**: פונקציה זו מקבלת בקשת **DNS** ונותנת לה תשובה. אם הפקטה מיועדת לכתובת שמופיעה ב-**spoof_dict**, אז יש להפנות לפונקציה **get_spoofed_dns_response** ולהשיג תשובה משם. אחרת יש להפנות לפונקציה **get_real_dns_response** ולהשיג תשובה משם.

מומלץ מאוד להיעזר באתר:
https://thepacketgeek.com/scapy/building-network-tools/part-09/
כדי להבין איך מממשים את הפונקציות האלה :)

טכנו"צ סייבר – תרגיל 4 - **"SPOOF"**

סיכום

הרצה נכונה של התרגיל תכלול שני מחשבים:

- מחשב המזכיר מריץ את **app.py** שבתיקייה **server**, ובו זמנית את **ex4_skeleton.py** (שבעצמו מריץ שני תהליכים, אחד **ArpSpoofer** ואחד **DnsHandler**).
- מחשבו של דופנשמירץ שלא מריץ אף קוד, אבל בגישה ל **mail.doofle.com** מגיע למסך התחברות גוגל מזויף: המייל והסיסמא שתוזן שם תישמר בקובץ **passwords.txt** על מחשב המזכיר

נסו לחשוב: מה אם מחשב המזכיר יריץ רק את **ArpSpoofer** ואת **DnsHandler**? מה אם יריץ רק את **app.py** ואת **ArpSpoofer**? מה אם יריץ רק את **app.py** ואת **DnsHandler**? מה החשיבות של כל אחד מהם להונאה מוצלחת ושליפת פרטי ההתחברות של דופנשמירץ?

בונוס מעשי

בבונוס הקרוב נעשה קצת חילוף תפקידים. הפעם נשחק את התפקיד של המגן, נרצה ללמוד לזהות תקיפת **ARP Spoofing** ולזהות את התוקף.

1) כתבו קוד אשר מנסה לזהות מתקפות **ARP Spoofing** על המחשב עליו הוא רץ
2) כתבו קוד אשר בהינתן פאקטת **ARP** מזויפת ינסה למצוא מה הכתובת אייפי של המחשב התוקף
3) נסו לחשוב כיצד ניתן וצריך להגיב בהינתן תקיפה שכזו? (פרטו ב **README**)

במהלך הכתיבה נסו לחשוב על מקרי קיצון, מה קורה אם מחשב מחליף כתובת **IP**? האם צריך להגן במתקפה על **spoofing** של כל כתובת ה-**IP**? איזה משתנים משתנים בתדירות גבוהה ואיזה פחות? מה המחיר של **false-positive**? מה המחיר של **false-negative**?

הוסיפו קטע קצר ל-**README** אשר מפרט על השיקולים שלכם.

טכנו"צ סייבר – תרגיל 4 - "SPOOF"

חומרי עזר

גם הפעם מסופק לכם קובץ שלד לתרגיל. קובץ השלד מכיל את כל הקוד שעוטף את ה-**Spoofing**. אותו קוד בפועל יכול לשמש בתור פיקוד אותו הנוזקה תקבל מהמחשב שלנו (נדלג על השלב הזה בתרגיל).
מסופק לכם גם השרת הזדוני שמתעד סיסמאות. על מנת להריץ אותו, עליכם להריץ את הפקודה:

sudo python3 app.py

בתיקיה הרלוונטית. לכשהשרת יקבל סיסמאות מדופנשמירץ, הוא יתעד אותן בקובץ:

passwords.txt

ספריות:

יש להתקין את הספרייה **scapy**. בנוסף יש להתקין את כל הספריות הדרושות לשרת הזדוני (ניתן לעשות זאת ע"י **pip install –r requirements.txt** בתיקיית ה-**server**).
ב-**scapy** תשתמשו באופן משמעותי בשתי המחלקות שתממשו בתרגיל, הנה ממש בקצרה מה שכדאי לדעת:

- **scapy** מסוגלת להסניף פקטות באופן חופשי מהרשת ע"י הפונקציה **sniff** שיכולה לקבל פרמטרים להסנפה בדומה ל **Wireshark** וגם פונקציית **callback** שרצה על כל פקטה מוסנפת.
- **scapy** מסוגלת לשלוח פקטה ע"י הפונקציה **send**.
- **scapy** מסוגלת לשלוח פקטה ולהחזיר לכם את התשובה שהתקבלה עליה, ע"י הפונקציות **sr, srp, sr1, srp1**, כאשר **1** מסמן שתוחזר רק התשובה הראשונה ו **p** מסמן עבודה בשכבה 2 כאשר בלי **p** מסמן עבודה בשכבה 3.
- **scapy** מייצגת פקטות ע"י הסוג **scapy.packet.Packet** שהוא זה שמתקבל ע"י פונ' הסנפה ונשלח לפונ' שליחה. כל פקטה מחולקת לשכבות כאשר הגישה לשכבות היא ע"י אופטור **[]** למשל **pkt[DNS]** הוא שכבת ה **DNS** של הפקטה ו-**pkt[IP].src** הוא ה- **ip src** של הפקטה. אפשר (ותצטרכו) גם לבנות פקטות ידנית ע"י התחביר (שכנראה תראו הרבה ממנו באינטרנט):

  L1(<l1 params>) / L2(<l2 params>) / l3(<l3 params>)...

  כאשר **Li** היא השכבה ה-**i** (למשל **IP, UDP, Ethernet, HTTP, ARP** וכו') ושכבה גבוהה יותר היא עם **i** גבוה יותר ("נמוך" יותר בבצל השכבות של הפקטה). שימו לב שאין צורך להגדיר את כל השכבות – אם אין מה להוסיף לגבי שכבה 2 אתם יכולים פשוט להתעלם ממנה ולבנות את הפקטה **IP(...)/TCP(...)/...** וכו'.

אנחנו ממליצים בחום גם לקרוא את הדוקומנטציה של **scapy**. מזהירים מראש שהיא לא מושלמת.

אזהרות

בעבר היינו מריצים את התרגיל הזה בין שני מחשבים מעל הרשת המתל"מית, ואז בעקבות טעויות של צוערים, קרו מקרים בהם הרשת המתל"מית נפלה. אנחנו עושים את התרגיל ברשת סטרילית של שתי מכונות וירטואליות. שימו לב לוודא שאתם מבינים מה הקוד שלכם עושה לפני שאתם מפעילים אותו.
בפרט שימו לב לכך שהתקיפה שלכם מכוונת **באופן ספציפי** למכונה הוירטואלית השניה. **אל תפעילו את התקיפה מחוץ לרשת של שתי המכונות.**
המלצה שלנו, כתבו את החלקים השונים של התרגיל בשלבים ובדקו אותם בזמן שאתם כותבים.
מומלץ להשתמש ב **Wireshark** כדי לוודא שהכל עובד כמו שצריך.

טכנו"צ סייבר – תרגיל 4 - **"SPOOF"**

הגשה

עליכם להגיש רק את קובץ ה-**skeleton** הערוך (ו-**README** במידת הצורך). כרגיל הגישו את התרגיל בתוך: **ex4_[FULL_NAME1]_[FULL_NAME2].zip** (ללא סוגריים מרובעות בשם הסופי לקובץ) שימו לב להגיש קובץ **zip** ולא **rar** או פורמט אחר ולהקפיד על השם הנכון לקובץ.

בהצלחה!