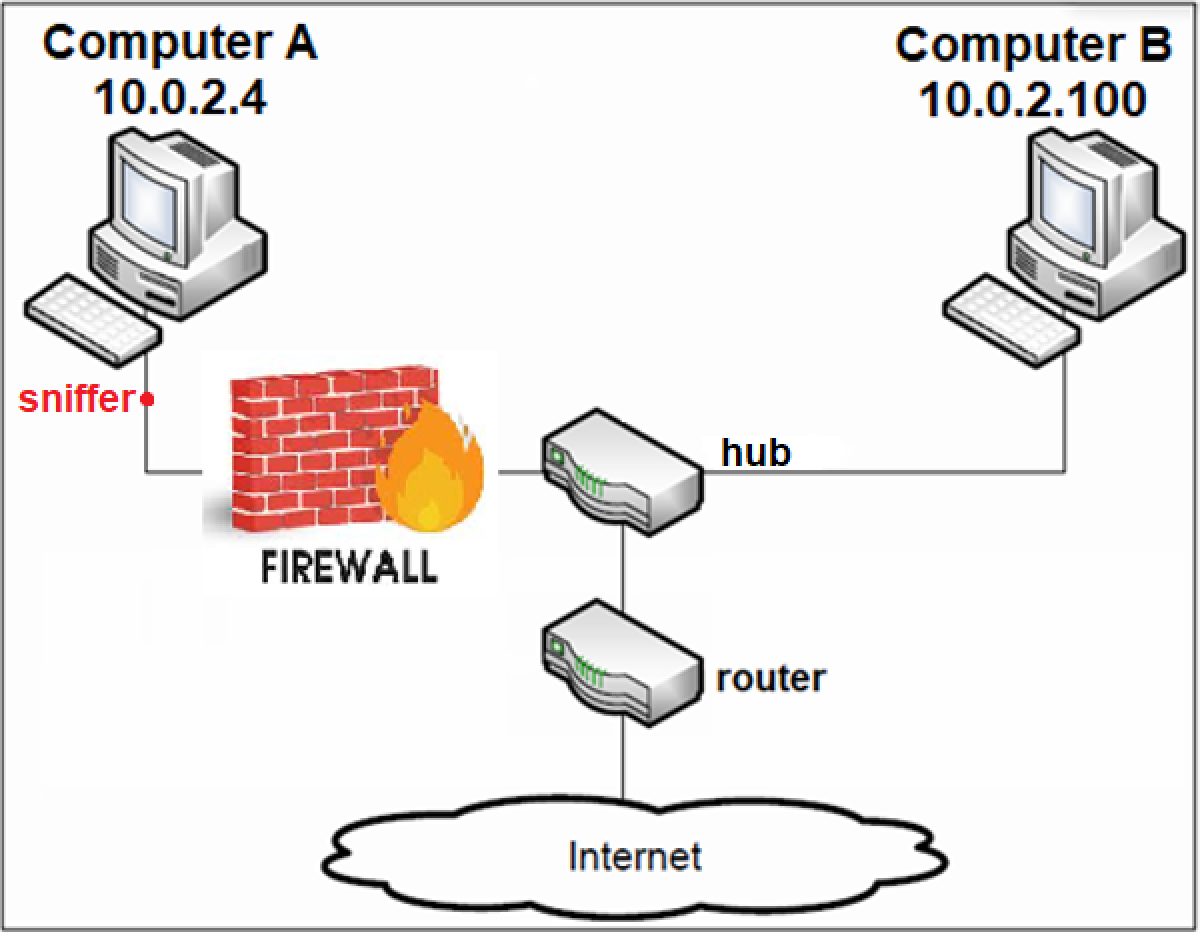
# **A05 – Linux Firewall Exploration Lab**

## כתובות IP לכל מחשב

|  |  |  |
| --- | --- | --- |
| Name | IP | MAC |
| Computer A | 10.0.2.4 | - |
| Computer B | 10.0.2.100 | - |

## Firewall

התמונה מטה מייצגת את מבנה הרשת



Task 1: Using Firewall

### מבוא:

#### תיאור

במשימה זו נגדיר חוקים שונים לטבלאות הFIREWALL

#### מטרה

לבטל אפשרות ליצירת חיבור TELNET בין שני מחשבים, וחסימת אפשרות גישה לאתר ספציפי דרך מחשב מסוים.

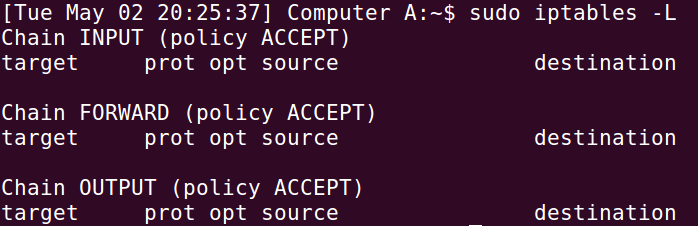
#### תוצאה מצופה

כאשר ננסה ליצור חיבור TELNET או ננסה לגשת לאתר החסום נכשל ולא נקבל חיבור מוצלח.

#### ביצוע המשימה

##### **Prevent A from doing telnet to Machine B**

תחילה בדקנו שטבלת החוקים של חומת האש ריקה



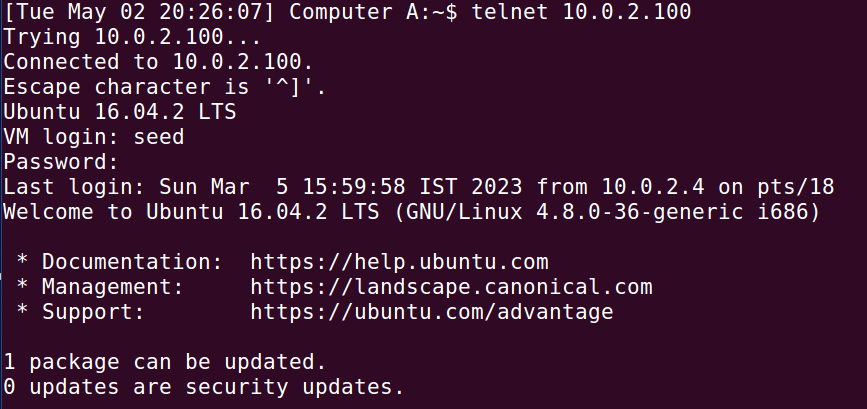
ניתן לראות שאין חוקים מוגדרים בטבלת מחשב A

INPUT – טבלה האחראית על הכנסת מידע למחשב

FORWARD – טבלה האחראית על העברת המידע למחשב אחר

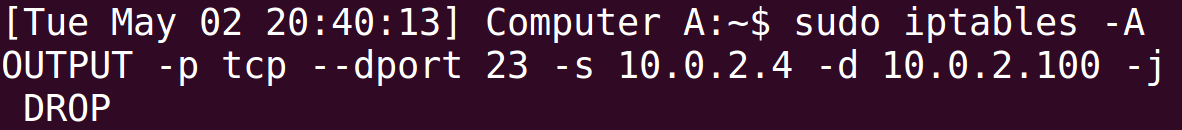
OUTPUT – טבלה האחראית על הוצאת המידע מהמחשב

כעת נרצה לבדוק שאכן ניתן ליצור חיבור TELNET בין מחשב A למחשב B



ניתן לראות שהחיבור נוצר בהצלחה

כעת נרשום את הפקודה הבאה במחשב A כדי לחסום את האפשרות ליצור חיבור TELNET עם מחשב B



-A – הוספת חוק חדש

OUTPUT – הטבלה אליה אנחנו מוסיפים את החוק

-p – פרוטוקול התקשורת של TELNET אותו אנחנו רוצים לחסום

--dport – פורט היעד אליו נשלח הפאקט

-s – מחשב הSOURCE ממנו נשלח הפאקט

-d – מחשב הdestination אליו מיועדת הפאקט

-j – הפעולה אותה נרצה לבצע על הפאקט (jump)

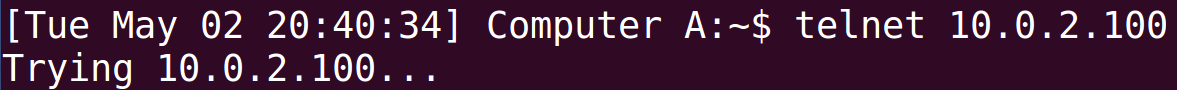
DROP – הפעולה שמתבצעת על הפאקט היא זריקה

כעת נבדוק שאכן הפקודה נוספה לטבלת הOUTPUT



ניתן לראות בטבלה הOUTPUT את החוק החדש שאומר לזרוק פאקטות אשר מועברות מIP 10.0.2.4 מחשב A לIP 10.0.2.100 מחשב B בתקשורת TCP TELNET.

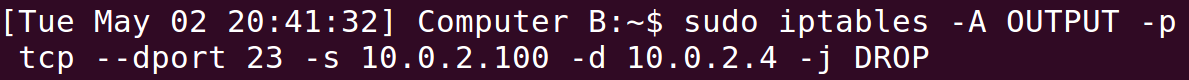
כעת ננסה ליצור חיבור ממחשב A למחשב B בTELNET



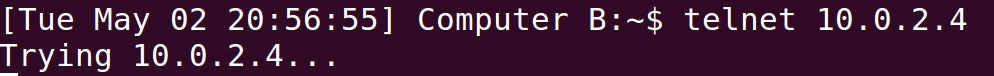
ניתן לראות שמחשב A מנסה ליצור חיבור אך ללא הצלחה, בדקנו גם בWIRESHARK וראינו שהפאקט לא הועברה לכן החיבור לא יצליח לעולם.

##### **Prevent B from doing telnet to Machine A**

כעת ביצענו את אותן הפעולות גם במחשב B רק שהפכנו בפקודה את הIP של המקור ושל היעד וכך חסמנו את אופציית יצירת חיבור TELNET בין מחשב B למחשב A



לאחר מכן ביצענו את אותן הבדיקות וראינו שאכן החוק נוסף לטבלה, ושלא ניתן ליצור חיבור Telnet בין המחשבים.



התוצאות היו זהות.

כמו כן, יכולנו לרשום את הפקודה הבאה ממחשב A ולא לעבור למחשב B:

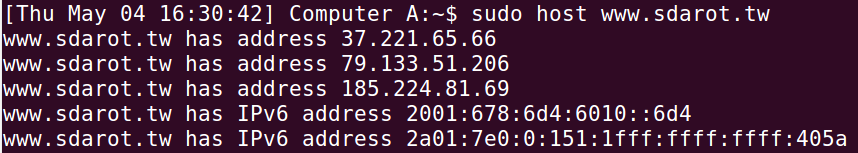
sudo iptables -A INPUT -p tcp --dport 23 -s 10.0.2.100 -d 10.0.2.4 -j DROP

##### **Prevent A from visiting an external web site. You can choose any web site that you like to block, but keep in mind, some web servers have multiple IP addresses.**

כעת ביצענו מחיקה לכל החוקים בטבלאות על ידי הרצת הפקודה sudo iptables -F

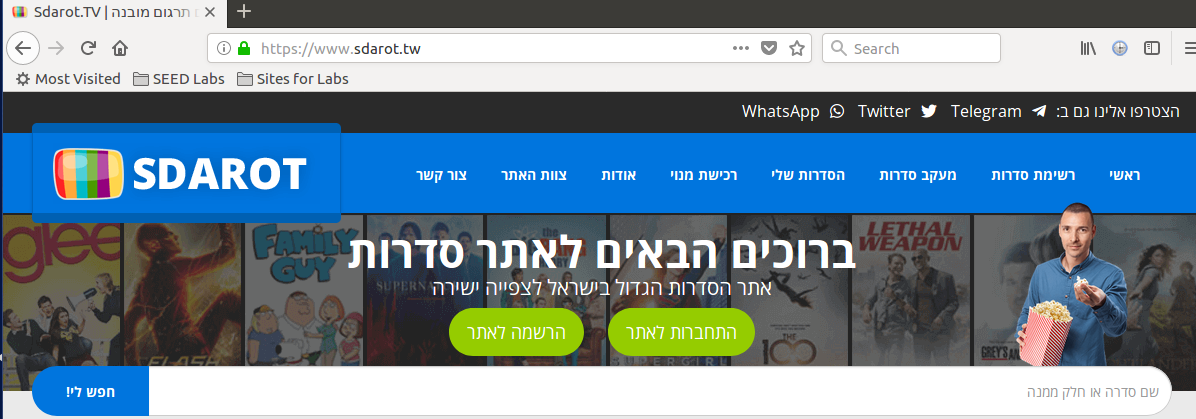
ונרצה להגדיר חוק חדש אשר יחסום גישה לאתר www.sdarot.tw

תחילה נבצע host לwww.sdarot.tw כדי לראות מה הכתובות של האתר



קיבלנו 3 כתובות IPv4 שאותם נרצה לחסום כדי למנוע כניסה לאתר

כעת ננסה להיכנס לקישור ונראה שהאתר תקין



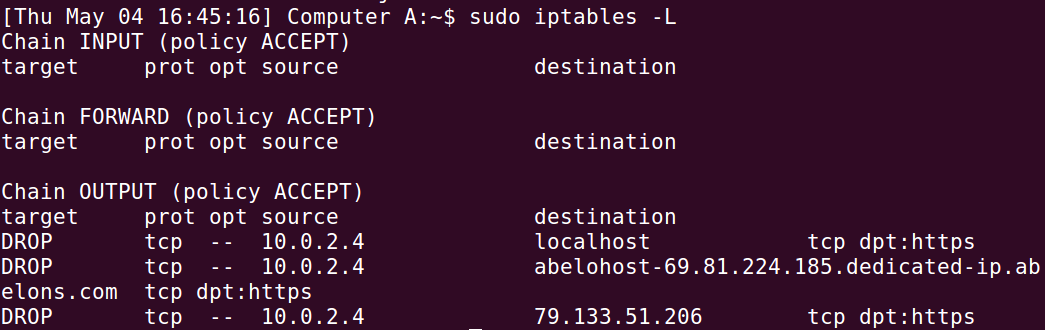
ניתן לראות שהקישור מפנה לאתר בהצלחה

נוסיף את החוקים הבאים לטבלת הOUTPUT לצורך חסימת גישה לאתר



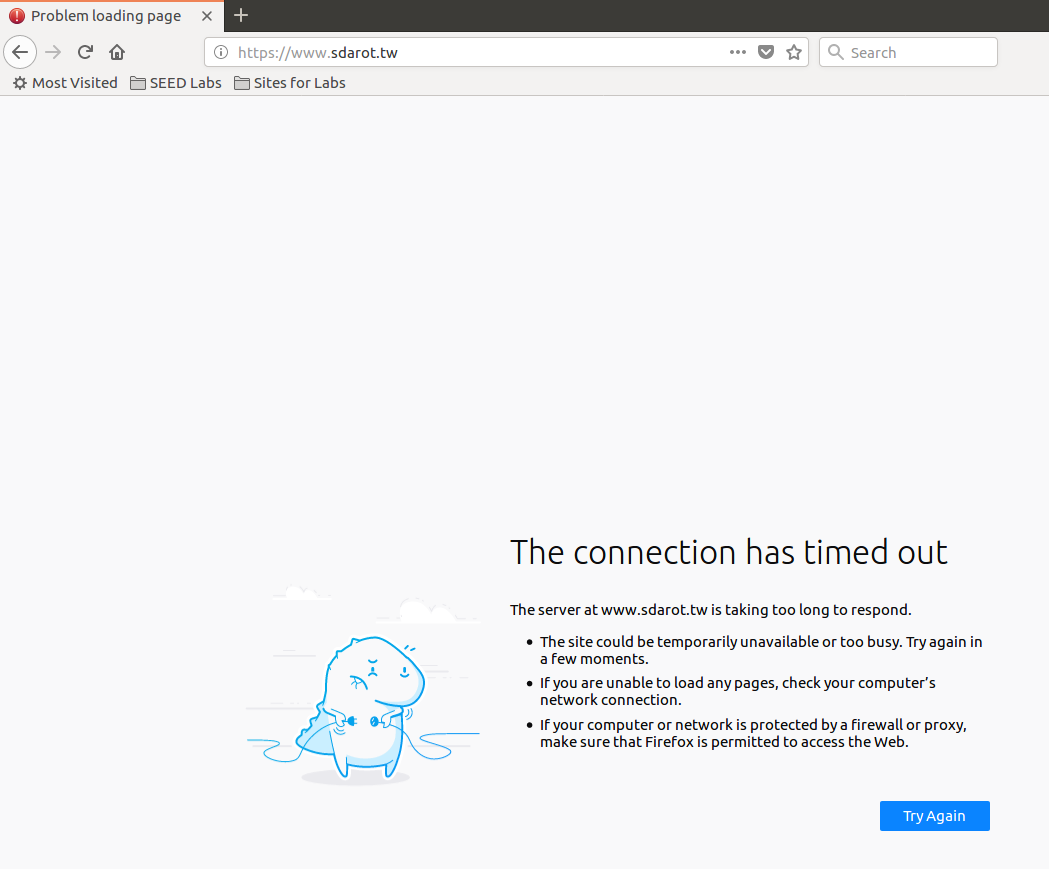
כתבנו חוק לכל כתובת IP

נרצה לוודא שהחוקים אכן נוספו לטבלת הOUTPUT



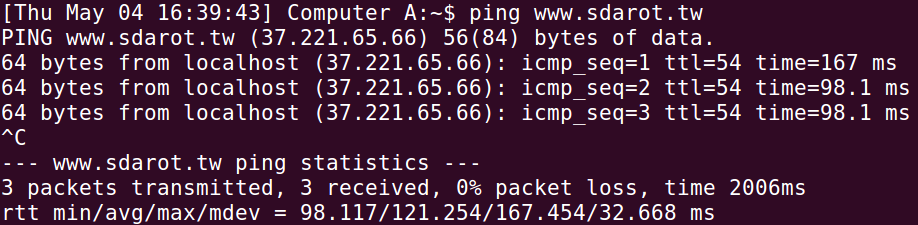
ניתן לראות שהחוקים נוספו בהצלחה לטבלה

כעת ננסה לגשת לאתר [www.sdarot.tw](http://www.sdarot.tw)



לא ניתן לגשת לאתר המבוקש, רואים זאת לפי כך שלא קיבלנו תוצאה זהה למצב לפני כתיבת החוק בFIREWALL וקיבלנו connection has times out מאחר ולא הצלחנו ליצור קשר עם השרת של האתר.

ביצענו בנוסף פינג לאתר כדי לראות שהגישה בפרוטוקול ICMP לא נחסמה גם



וראינו שניתן לגשת לאתר בפרוטוקול שהוא לא TCP כפי שהגדרנו בחוקים.

#### סיכום המשימה

הצלחנו לבצע את המשימה, ניתן לראות שהצלחנו לחסום גישה בתקשורת TELNET ממחשב A למחשב B ולהפך והראנו זאת על ידי כך שניסיון התחברות נוסף לאחר הכנסת החוקים החדשים לFIREWALL נכשל.

בנוסף הצלחנו לחסום גישה לאתר ספציפי שבחרנו [www.sdarot.tw](http://www.sdarot.tw) והראינו זאת על ידי כך שלאחר כתיבת החוקים בFIREWALL קיבלנו תוצאה שהחיבור פג תוקף.

גילינו כיצד להגדיר חוקים לחסימת גישות עבור תקשורת ספציפית ועבור אתר ספציפי.

התוצאות התאימו למצופה מאחר ולפני כתיבת החוקים הייתה לנו אפשרות לגשת לחיבורים שרצינו ולאחר חסימתם והוספת החוקים לטבלה IPTABLES לא הצלחנו לבצע חיבור שוב פעם.

לא נתקלנו בבעיות במהלך ביצוע המשימה.

Task 2: Implementing a Simple Firewall

### מבוא:

#### תיאור

במשימה זו נגדיר חוקים שונים לטבלאות הFIREWALL בעזרת קוד בשפת C

#### מטרה

לבטל אפשרות ליצירת חיבור TELNET בין שני מחשבים, חסימת אפשרות גישה לאתר ספציפי דרך מחשב מסוים, חסימת אפשרות לביצוע פינג לאתר ספציפי דרך מחשב מסוים, ביטול אפשרות לביצוע חיבורי SSH ממחשב מסוים.

#### תוצאה מצופה

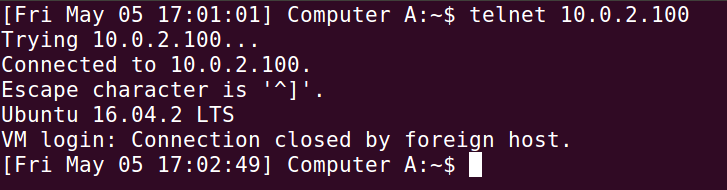
עקב הפעלת החוקים במחשב יהיו כישלונות ביצירת חיבורים המוזכרים מעלה.

#### ביצוע המשימה

* הסבר קצר על NETFILTER  
  זוהי מסגרת בתוך הגרעין של מערכת לינוקס אשר מאפשר לבצע פעולות על תעבורת הרשת, כגון סינון, שינוי, הסנפה (יירוט).  
  משמש בעיקר ליישום חומות אש, כתובות NAT, והתעסקות בPACKETS.
* הסבר קצר על NAT  
  זוהי טכניקה אשר משמשת לתרגום כתובות IP ברשת הפרטית לכתובת IP חיצונית אחת. טכניקה זו מספקת חיסכון בכתובות IP ושכבת אבטחה בכך שמסתירה את הכתובת הפרטית ממנה יוצא הPACKET.
* הסבר קצר על LKM  
  פיסות קוד שניתן לטעון ולפרוק מליבת לינוקס מבלי לדרוש אתחול מחדש או לבצע COMPILE מחדש לליבה.   
  מאפשר שינוי דינמי של הליבה בזמן ריצה, ומאפשר למשתמשים להוסיף או להסיר תכונות ספציפיות לפי הצורך כגון פונקציונליות נוספת, מנהלי התקנים או ממשקי קריאת מערכת.   
  פיתוח ב- LKM נעשה בדרך כלל בקוד בשפות C או C++ שמתממשק עם ממשק תכנות היישומים (API) של ליבת לינוקס.   
  את הקוד מרכיבים לקובץ אובייקט, אשר מקושר לאחר מכן לקרנל כדי ליצור מודול הניתן לטעינה, כדי לטעון את המודול לתוך הקרנל נשתמש בפקודה insmod, וכדי לפרוק נשתמש בפקודה rmmod.

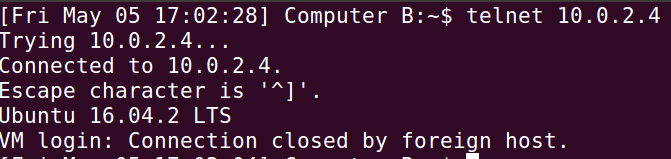
לפני הכנסת החוקים החדשים נעבור על כל החיבורים אותם נרצה לחסום ונראה כי הם תקינים וללא הוספת החוקים ניתן ליצור את החיבורים האלו.

ביצוע חיבור TELNET ממחשב A למחשב B



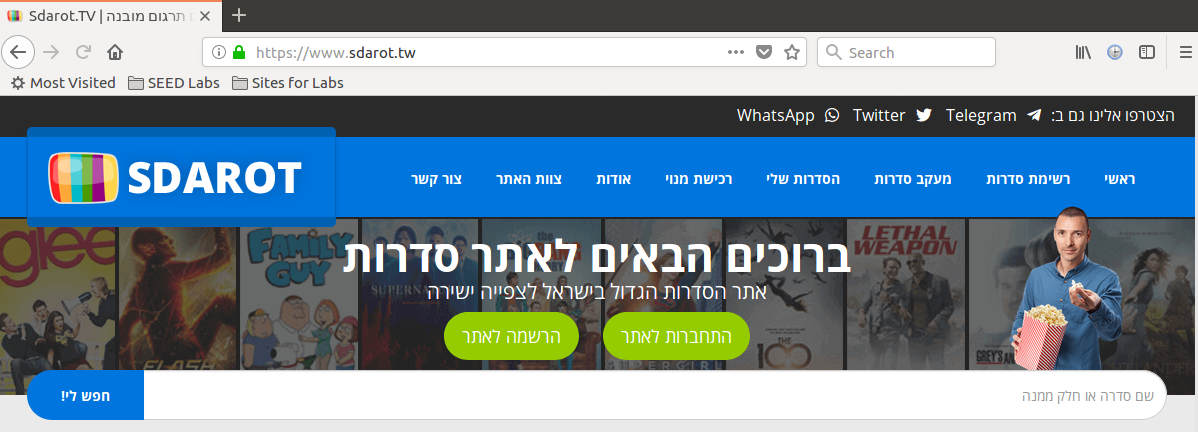
חיבור הTELNET הצליח

ביצוע חיבור TELNET ממחשב B למחשב A



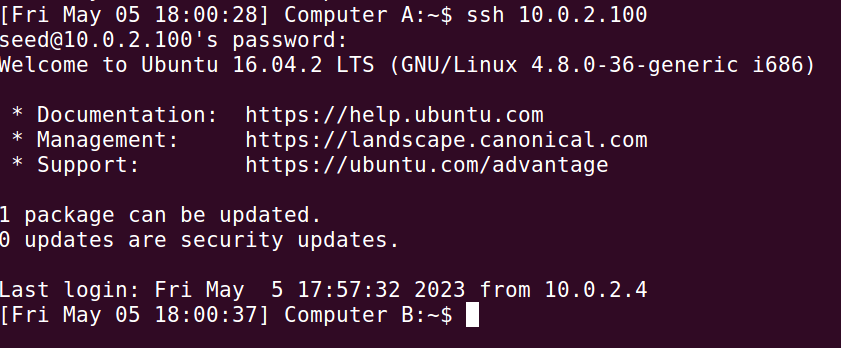
חיבור הTELNET הצליח

ניסיון התחברות לאתר סדרות



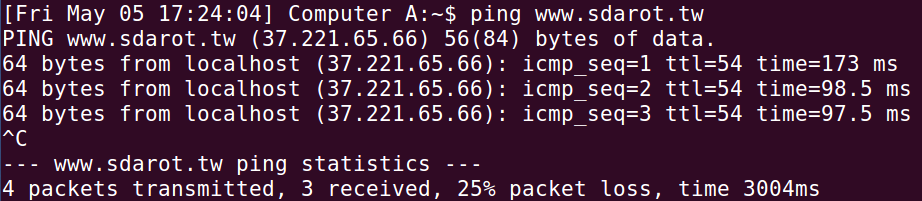
ניסיון החביר הצליח

ניסיון יצירת חיבור SSH ממחשב A למחשב B



חיבור הSSH בין המחשבים הצליח

ניסיון שליחת פינג לאתר סדרות



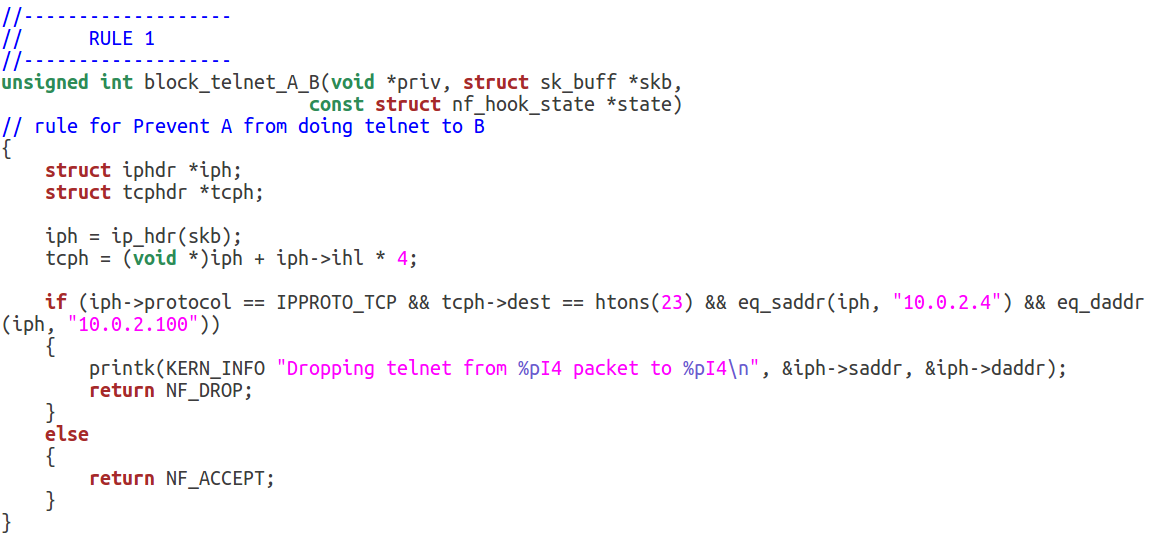
הפינג התקבל בהצלחה

כעת נרשום את הקוד הבא:



פונקציות אשר בודקות את האייפי בPACKET לבין IP שנשלח לפונקציה, פונקציה אחת לSRC ואחת לDST.

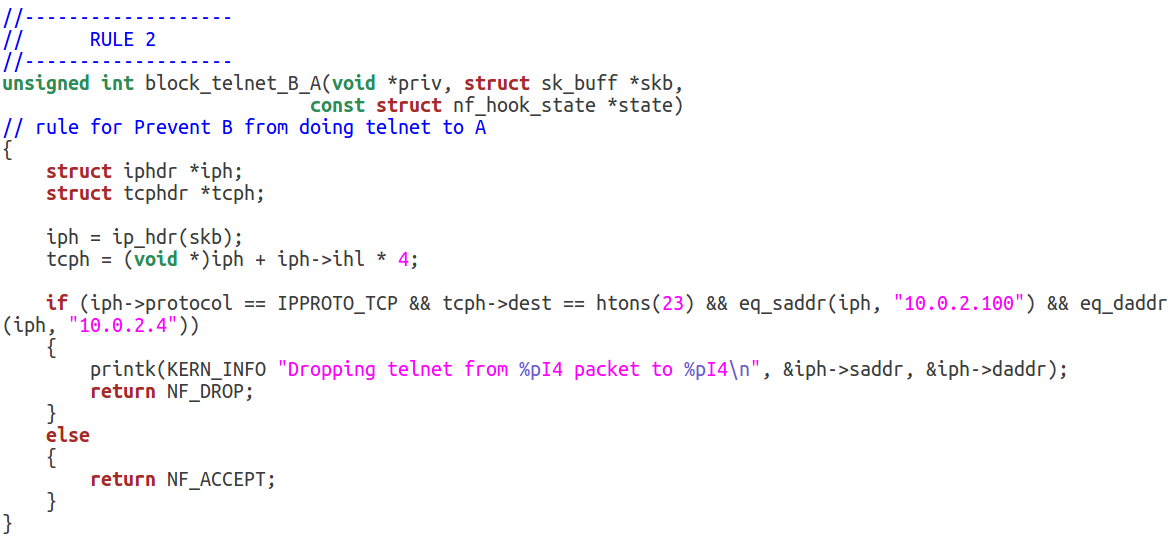
פקודת הSNPRINTF לוקחת את האייפי מההדר של הPACKET והופכת אותו מביטים בHEX לצורה שניתן לקרוא כגון 10.0.2.4



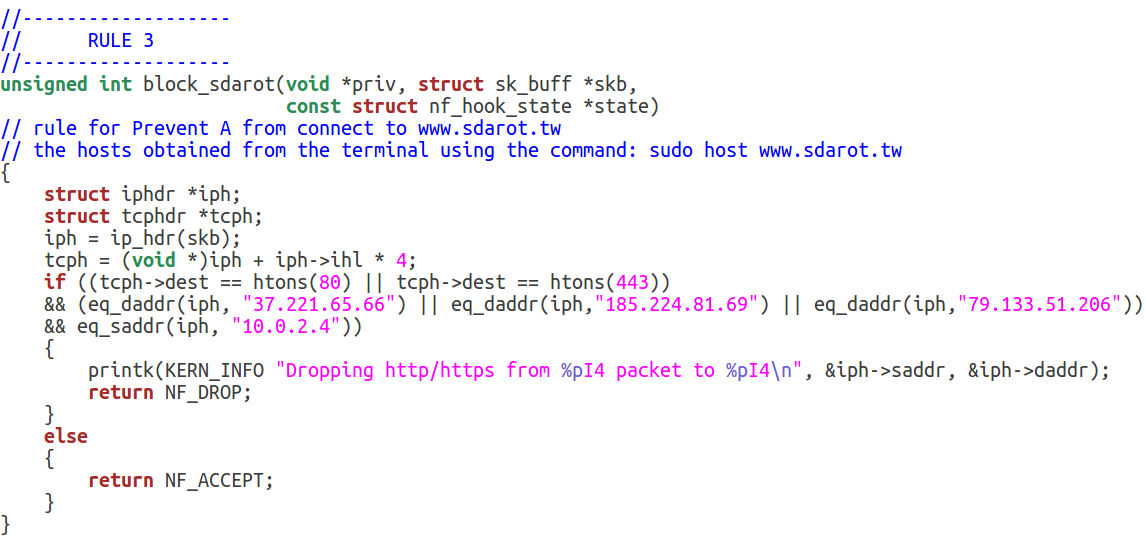
חוק 1 – מונע גישה בתקשורת TELNET ממחשב A למחשב B

Htons – ממיר את הפורט לHEX כך שנוכל להשוות עם הפורט שרשום בתוך ההדר של TCP

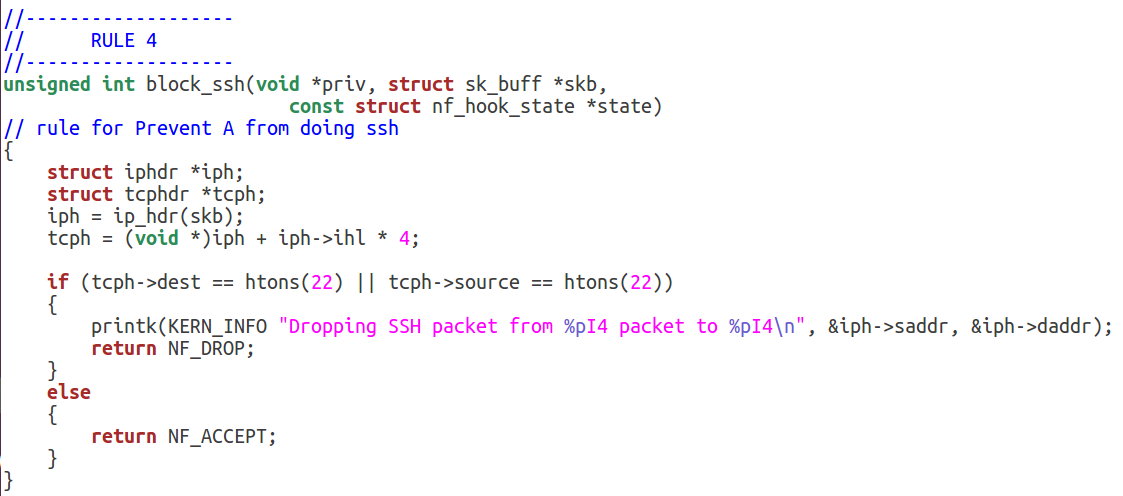
Tcph – לוקחים את המצביע להדר של IP שהוא ההתחלה של הבאפר של הPACKET ומוסיפים את אורך ההדר של IP כפול 4 ביטים ומגיעים לחלק שאחרי ההדר של IP והוא ההדר של TCP.



חוק 2 – מונע גישה בתקשורת TELNET ממחשב B למחשב A



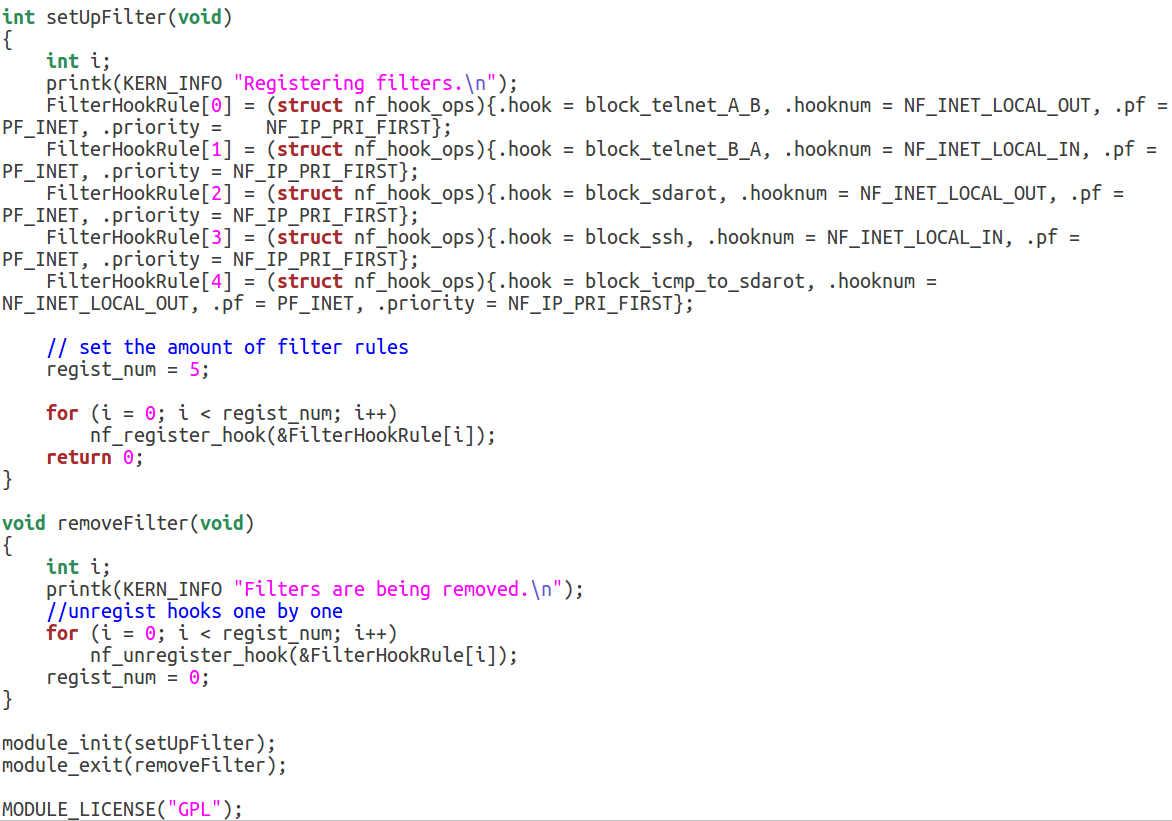
חוק 3 – מונע ממחשב A לגשת לאתר [www.sdarot.tw](http://www.sdarot.tw) על ידי חסימת גישה בפורט HTTP 80 ובפורט HTTPS 443 לכל כתובות הHOST של האתר



חוק 4 – מונע ממחשב A לקבל או ליצור חיבורי SSH מכל מחשב שהוא בפורט 22 שזה הפורט המיועד לSSH

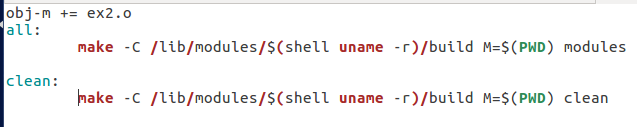


חוק 5 – מונע ממחשב A לבצע PING לשרתים של אתר www.sdarot.tw (מונע שליחת ICMP ECHO REQUEST)



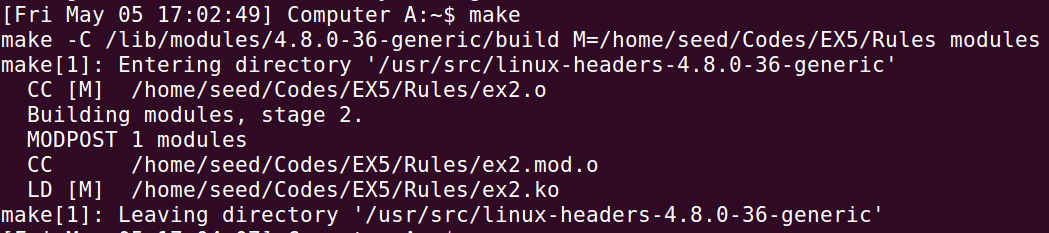
פונקציות setUpFilter ו- removeFilter מתארות מה יתבצע כאשר נרצה לטעון את המודול לקרנל ולהסירו.

נפתח קובץ חדש בשם Makefile אשר יכיל את הקוד הבא:



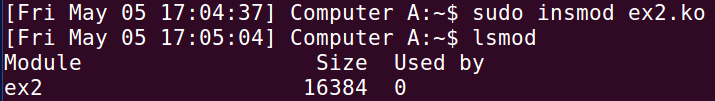
הקובץ הנ"ל אומר למערכת לבנות את המודל שכתבנו ולהוסיף אותו לתיקיית המודולים בקרנל.

נבצע הרצה לקוד על ידי כתיבת הפקודה make אשר מבצעת את הפקודות שרשמנו בקובץ Makefile



הקבצים נבנו בהצלחה

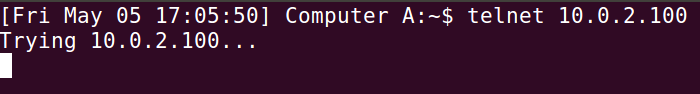
כעת נטען את המודול לקרנל



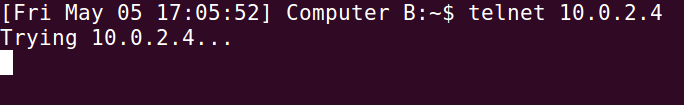
ניתן לראות שהמודול נטען לקרנל בהצלחה

כעת ננסה לבצע את החיבורים בהתחלה שוב פעם ונרצה לראות שהם נכשלים

נבצע חיבור TELNET ממחשב A למחשב B

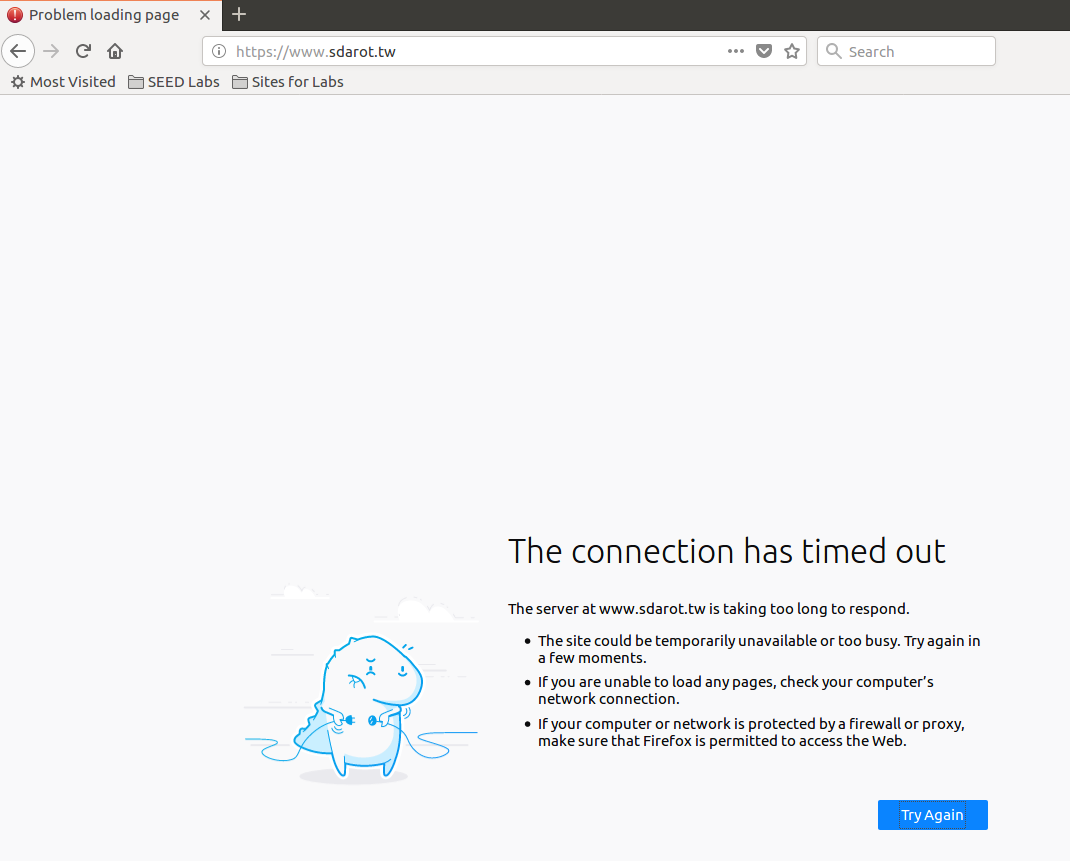


ניתן לראות שהחיבור נכשל בהצלחה מאחר ולא נוצר חיבור בין המחשבים

נבצע חיבור TELNET ממחשב B למחשב A

ניתן לראות שהחיבור נכשל בהצלחה מאחר ולא נוצר חיבור בין המחשבים

ננסה להתחבר ממחשב A לאתר www.sdarot.tw



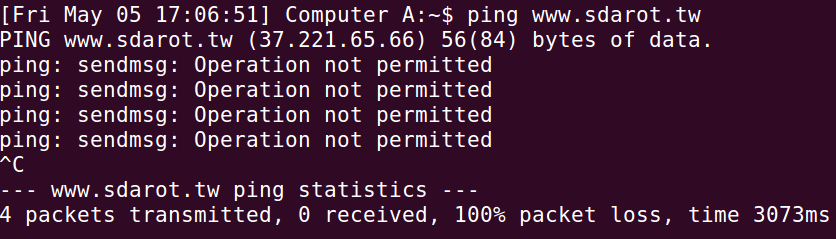
ניתן לראות שהחיבור פג תוקף ולכן החיבור נכשל בהצלחה

ננסה ליצור חיבור SSH ממחשב A למחשב B



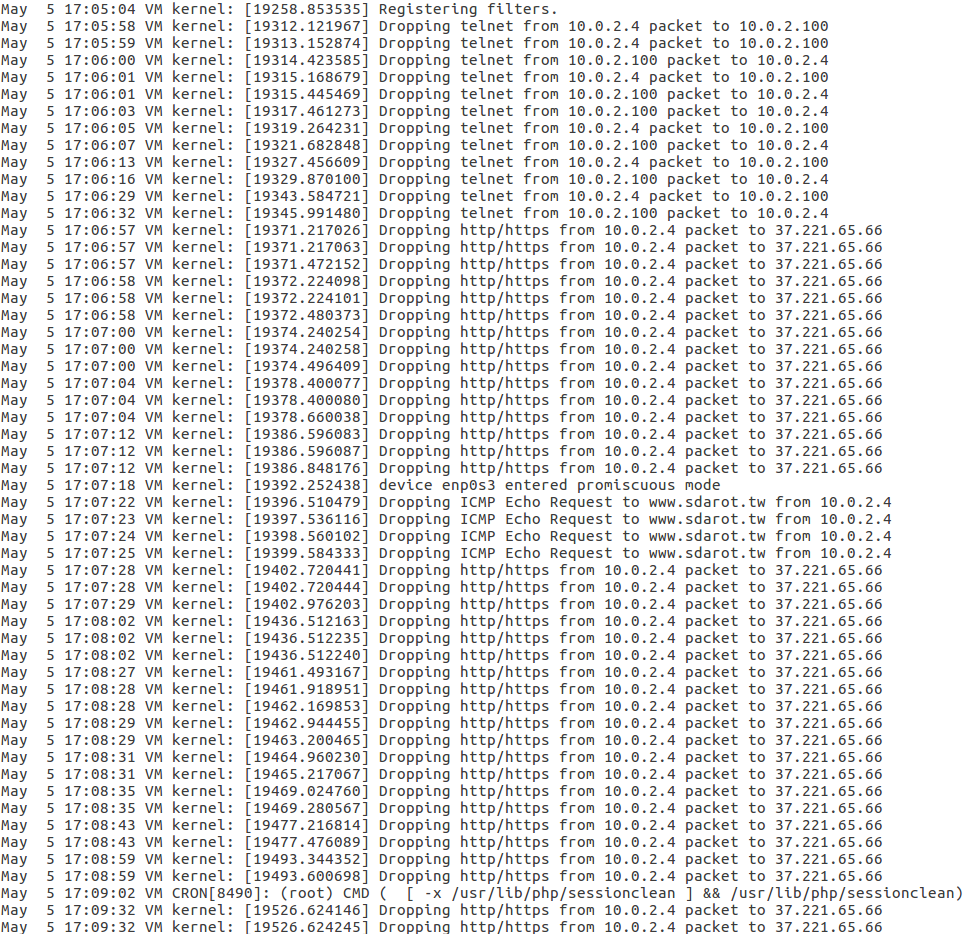
ניתן לראות שהחיבור נכשל בהצלחה מאחר ולא נוצר חיבור בין המחשבים

ננסה לשלוח ICMP ECHO REQUEST (PING) לאתר www.sdarot.tw



ניתן לראות שביצוע PING נכשל בהצלחה מאחר והבקשות הICMP אבדו ולא התקבלו אצל היעד.

ניתן לראות את הלוגים שהודפסו על ידי הקוד שלנו בקובץ הLOG של הKERNEL בנתיב /var/log/syslog



ניתן לראות שהודפסו הודעות על כל הPACKETS שנזרקו עקב החוקים שהגדרנו בFIREWALL.

#### סיכום המשימה

הצלחנו לבצע את המשימה, ניתן לראות שלפני הגדרת החוקים בKENREL יכולנו ליצור את כל החיבורים שהם:

TELNET בין A לB

TELNET בין B לA

חיבור לאתר [www.sdarot.tw](http://www.sdarot.tw)

יצירת חיבורי SSH ממחשב A למחשב B

שליחת PING לאתר [www.sdarot.tw](http://www.sdarot.tw)

ולאחר טעינת המודול שבנינו לKERNEL ניסיון נוסף ליצירת החיבורים האלו נכשל בהצלחה.

גילינו כיצד להגדיר חוקים לחסימת גישות בעזרת BACKEND בFIREWALL וכיצד לטעון ולהסיר מודולים מהKERNEL.

התוצאות התאימו למצופה מאחר ולפני כתיבת החוקים הייתה לנו אפשרות לגשת לחיבורים שהוזכרו מעלה, ולאחר טעינת החוקים בKERNEL כל החיבורים שהוזכרו נחסמו לאלתר.

כדי להצליח במשימה נעזרנו רבות בדוקומנטציה של NETFILTER בקישור הבא: <https://netfilter.org/documentation/HOWTO/netfilter-hacking-HOWTO-4.html>

Task 3: Evading Egress Filtering

### מבוא:

#### תיאור

במשימה זו נרצה לעקוף את החוקים המוגדרים בחומת האש בעזרת SSH TUNNEL

#### מטרה

נגדיר חוקים בחומת האש אשר מונעים חיבור בTELNET מהמחשב החוצה, ומונעים חיבור לאתר ספציפי מהמחשב, ונרצה לעקוף את החסימה על ידי יצירת SSH TUNNEL והעברת התעבורה דרכה.

#### תוצאה מצופה

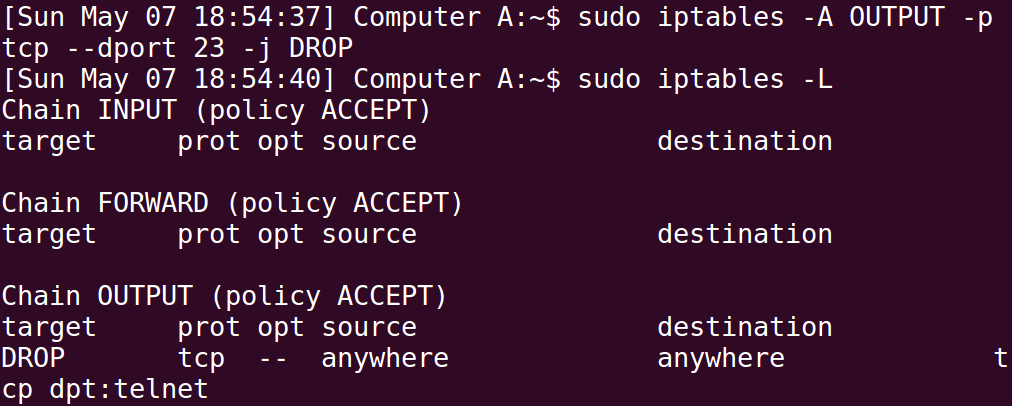
נצליח לבצע חיבור TELNET מהמחשב החסום ונצליח להתחבר לאתר החסום.

#### ביצוע המשימה

##### Block all the outgoing traffic to external telnet servers

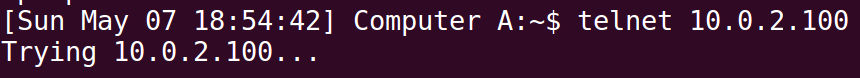
נרצה לבצע חסימה לכל התעבורה שיוצאת ממחשב A בתקשורת TELNET

לכן נכתוב את הפקודה הבאה בטרמינל:



ניתן לראות שהחוק נוסף בהצלחה אל טבלת החוקים של OUTPUT

ננסה לבצע חיבור TELNET למחשב חיצוני

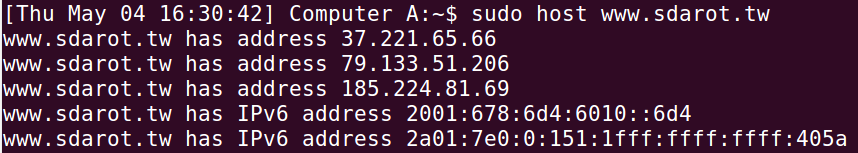


ניתן לראות שהחיבור נכשל בהצלחה מאחר ולא נוצר חיבור בין שני המחשבים

##### Block all the outgoing traffic to [www.sdarot.tw](http://www.sdarot.tw)

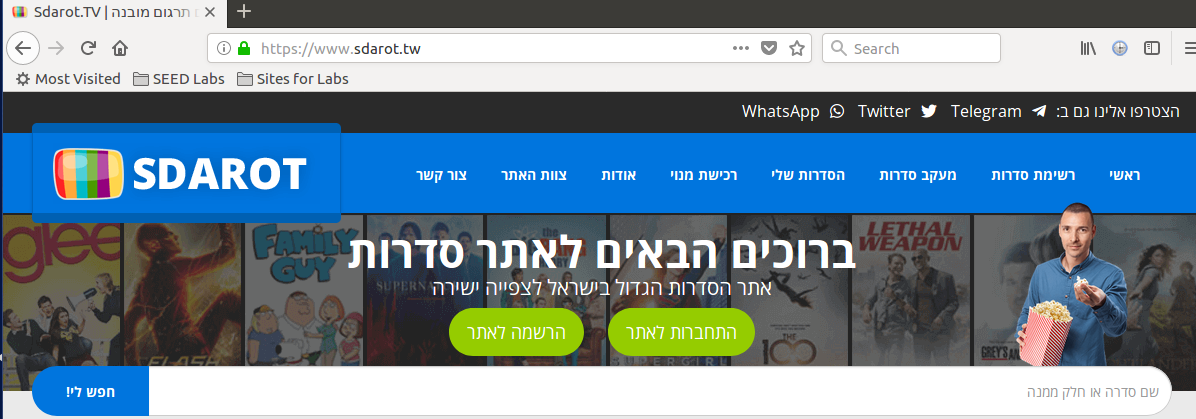
נרצה להגדיר חוק חדש אשר יחסום גישה לאתר www.sdarot.tw

תחילה נבצע host לwww.sdarot.tw כדי לראות מה הכתובות של האתר



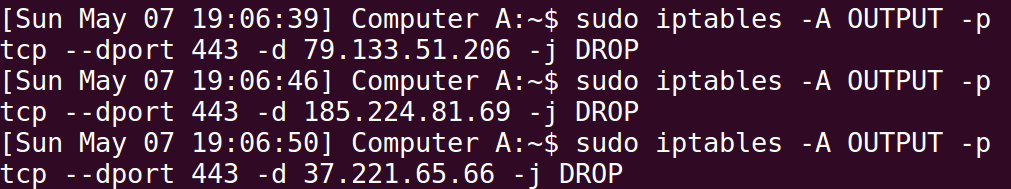
קיבלנו 3 כתובות IPv4 שאותם נרצה לחסום כדי למנוע כניסה לאתר

כעת ננסה להיכנס לקישור ונראה שהאתר תקין



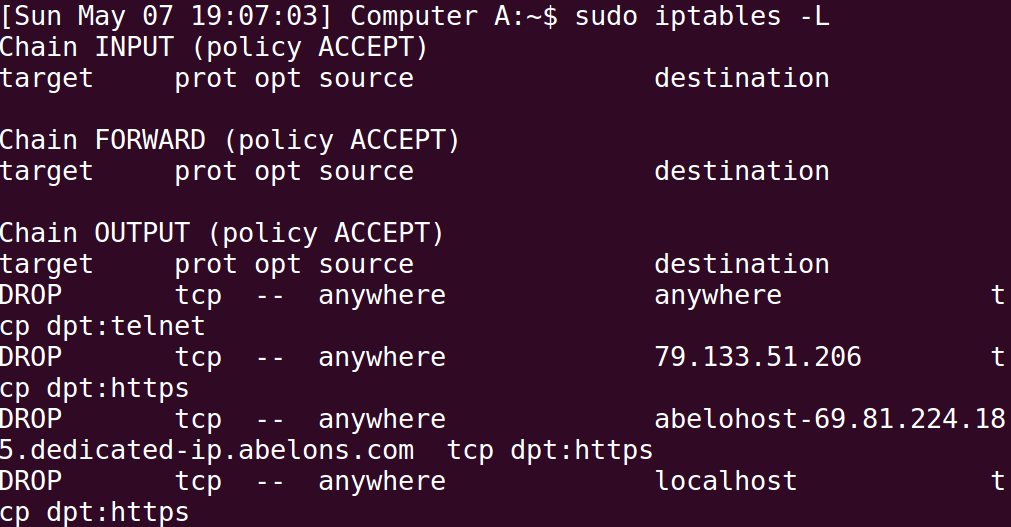
ניתן לראות שהקישור מפנה לאתר בהצלחה

נוסיף את החוקים הבאים לטבלת הOUTPUT לצורך חסימת גישה לאתר



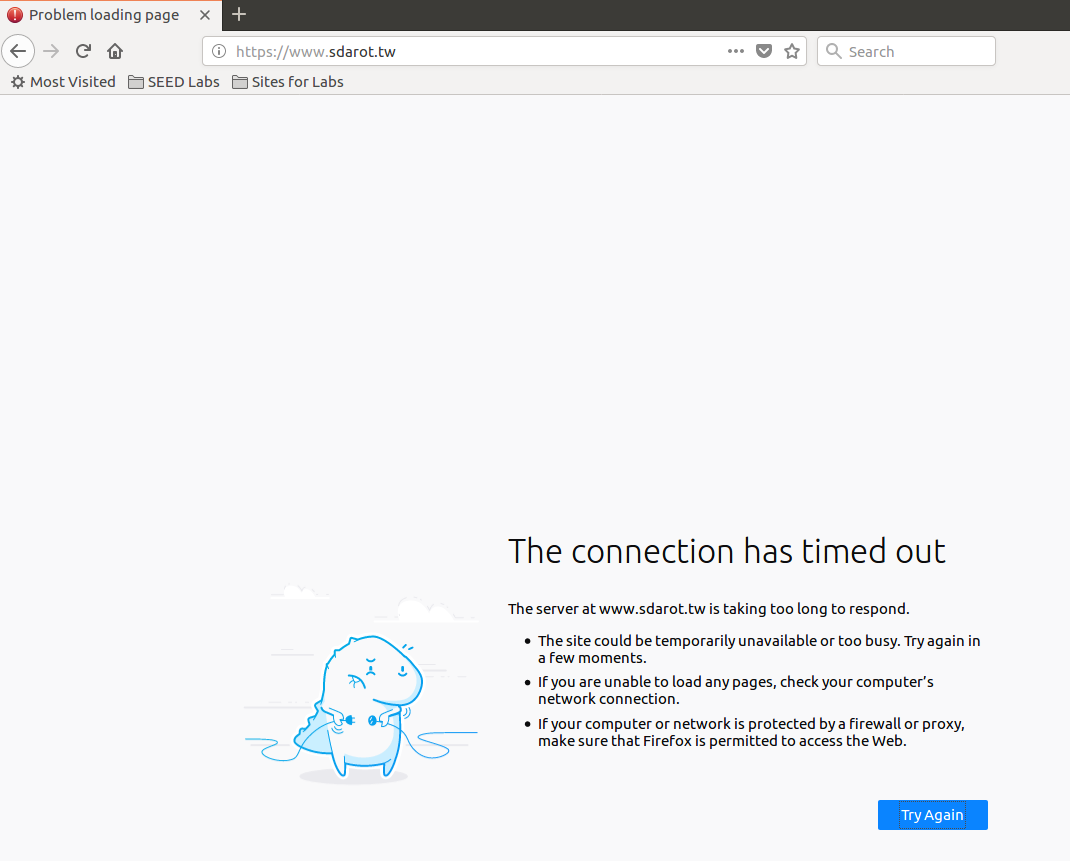
כתבנו חוק לכל כתובת IP

נרצה לוודא שהחוקים אכן נוספו לטבלת הOUTPUT



ניתן לראות שהחוקים נוספו בהצלחה לטבלה

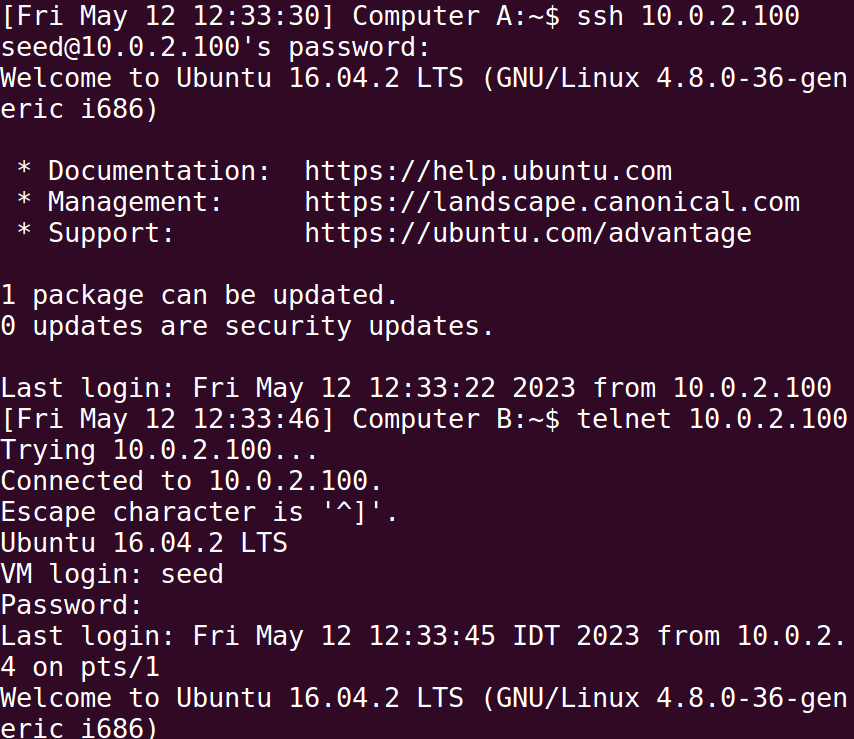
ננסה להתחבר ממחשב A לאתר www.sdarot.tw



ניתן לראות שהחיבור פג תוקף ולכן החיבור נכשל בהצלחה

##### Task 3.a: Telnet to Machine B through the firewall

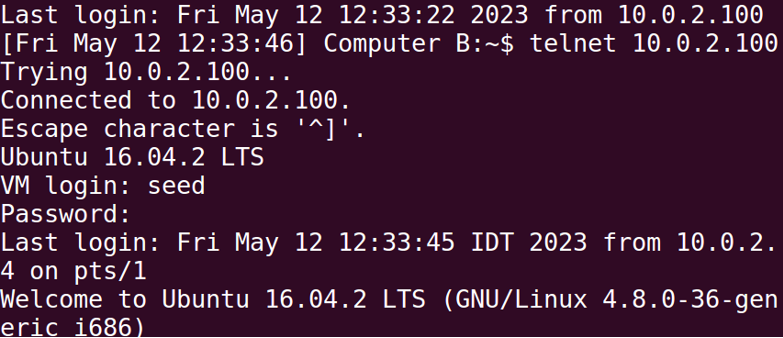
תחילה נבצע חיבור SSH בין מחשב A למחשב B



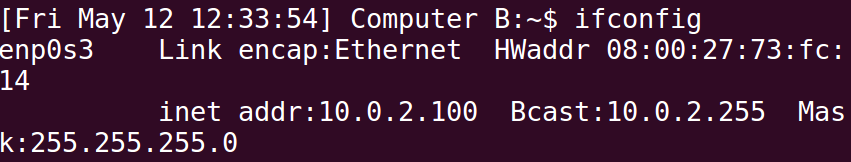
ניתן לראות שהחיבור נוצר בהצלחה

כעת לאחר יצירת הTUNNEL בין מחשב A למחשב B בעזרת SSH נרצה ליצור חיבור TELNET בין מחשב B לעצמו ובכך באופן עקיף יצרנו חיבור TELNET ממחשב A למחשב B ועקפנו את החוקים של חומת האש.

ביצענו חיבור TELNET למחשב B כאשר אנחנו כבר בתוך מחשב B

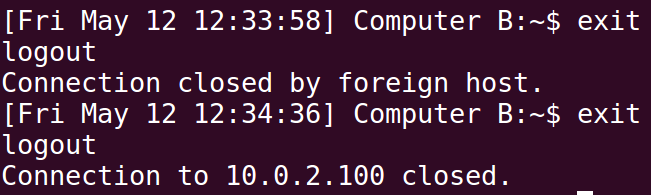


ניתן לראות שהחיבור הצליח ונבדוק שאכן הIP שמתקבל במכונה הוא הIP של מחשב B



קיבלנו את הIP של מחשב B 10.0.2.100

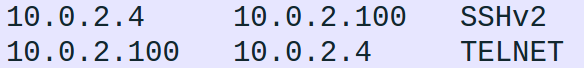
כעת נרצה לנתק את החיבורים על ידי EXIT



ניתן לראות שהיינו צריכים לבצע פעמיים EXIT כדי לחזור למחשב A, פעם ראשונה כדי לנתק את חיבור הTELNET ופעם שניה כדי לנתק את חיבור הSSH

כאשר בדקנו בWIRESHARK לא ראינו PACKETS שעוברים בTELNET מאחר וביצענו בין המחשב לעצמו

לכן ניסינו לבצע חיבור בין מחשב A לB בSSH ולאחר מכן חיבור בין מחשב B למחשב A בTELNET ובכך יצרנו חיבור בין מחשב A לעצמו בTELNET שדבר זה לא אמור להיות אפשרי עקב החוקים שהגדרנו בחומת האש, וניתן לראות שהצלחנו והWIRESHARK הסניף את הPACKETS המראות חיבור בין המחשבים.



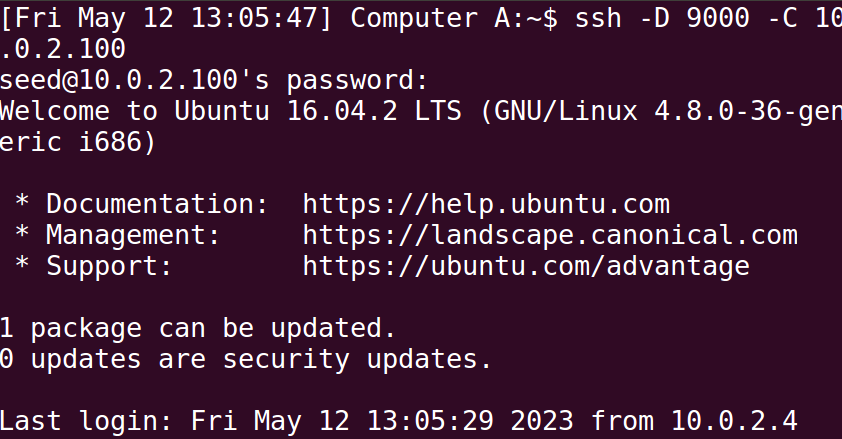
חיבור בין מחשב A לB בSSH

חיבור בין מחשב B לA בTELNET

##### Task 3.b: Connect to Sdarot using SSH Tunnel

נרצה לעקוף את חוקי חומת האש ולנסות לגשת לאתר [www.sdarot.tw](http://www.sdarot.tw) דרך TUNNEL שניצור בחיבור SSH בין מחשב A למחשב B

ניצור חיבור SSH בין מחשב A למחשב B על ידי פורט 9000



-D 9000 – יוצר פורט האזנה על מחשב A לצורך העברת הנתונים דרך פורט זה

-C – מכווץ את נתוני התעבורה שעוברים בפורט כדי להקטין את השימוש ברוחב פס

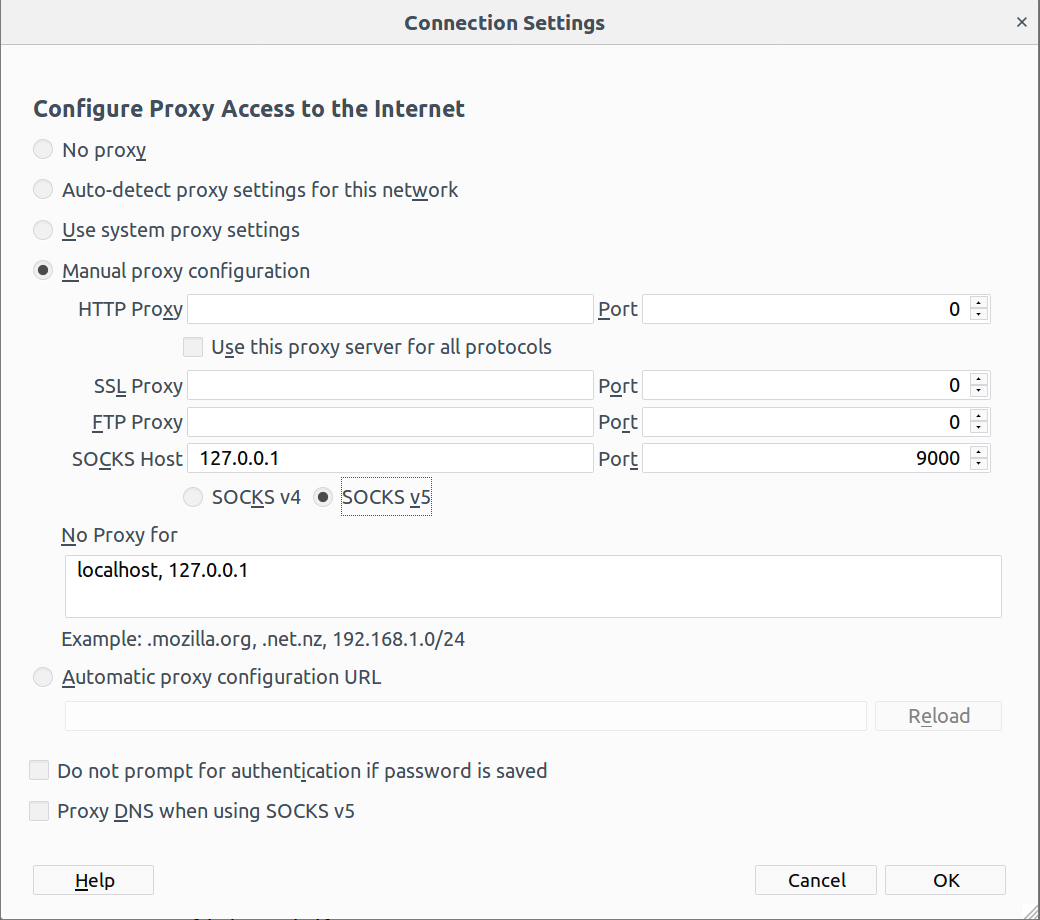
ניתן לראות שיצרנו חיבור SSH עם socks proxy בפורט 9000 וכעת נרצה להגדיר שאפליקציית firefox תעביר את תעבורת הרשת שלה דרך הSOCKET שפתחתנו וכך תבצע העברה למחשב B דרך הssh tunnel שפתחתנו.

SOCKS PROXY יוצר פורט האזנה לאפליקציות על המחשב, דרך הפורט הזה נוכל ליצור חיבור למחשב אחר ובכך להעביר את תעבורת הרשת שנשלחת לפורט הנ"ל.

נכנס בFIREFOX ללשונית

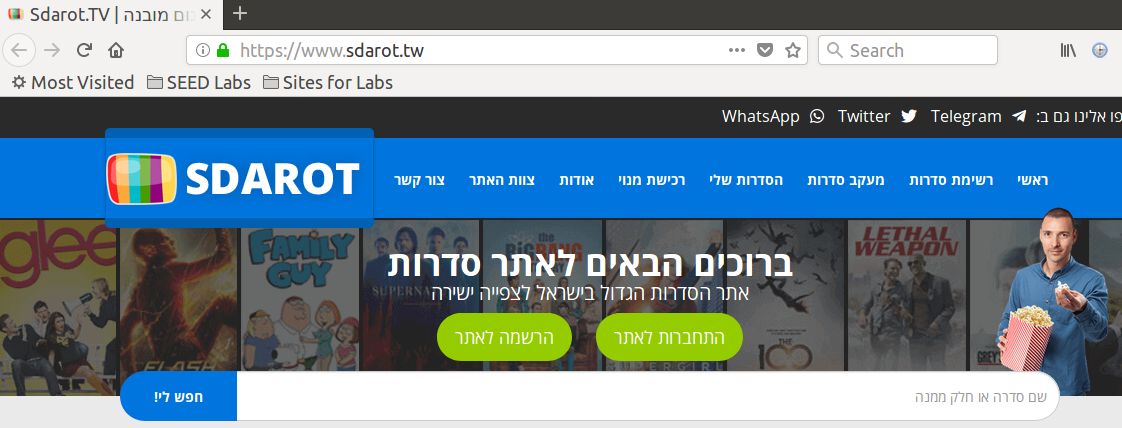
EDIT->PREFERENCES->NETWORK PROXY->SETTING

ונשנה את ההגדרות לפי התמונה מטה



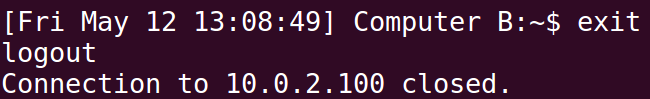
כעת הגדרנו שכל תעבורת האפליקציה של FIREFOX תעבור דרך הSOCKET שפתחתנו בפורט 9000 על המחשב הלוקאלי מחשב A.

כעת נתחבר ממחשב A לאתר www.sdarot.tw



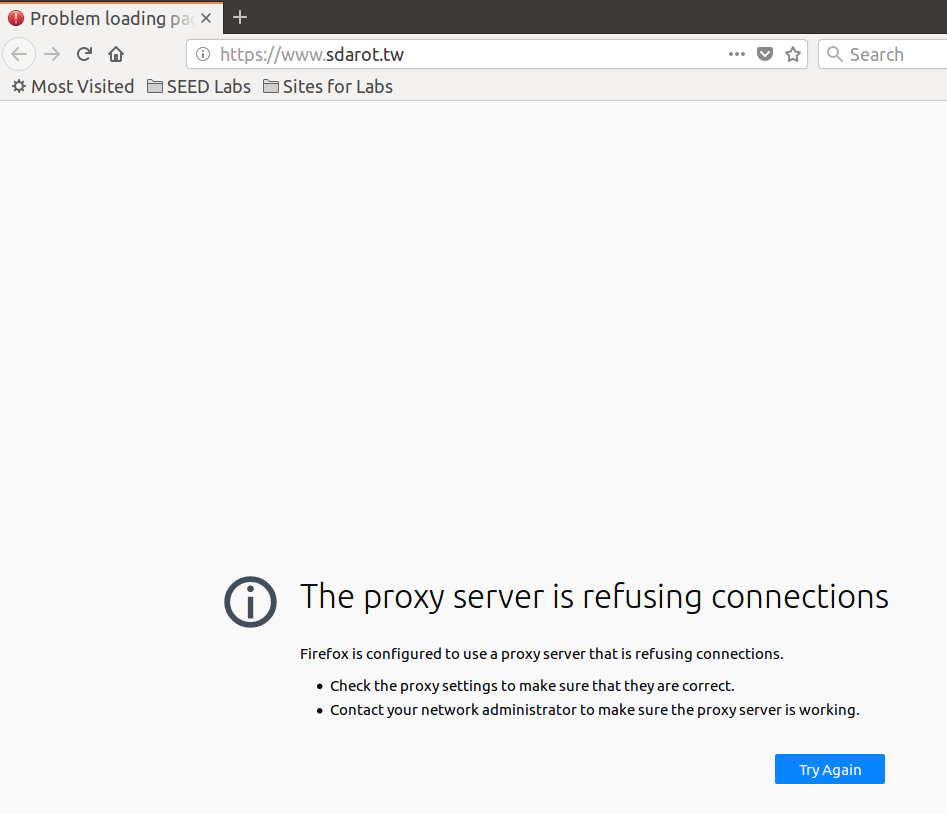
ניתן לראות שהחיבור לאתר הצליח לאחר העברת התעבורה דרך פורט 9000 המחובר בSSH למחשב B

כעת ננתק את חיבור הSSH בין מחשב A למחשב B



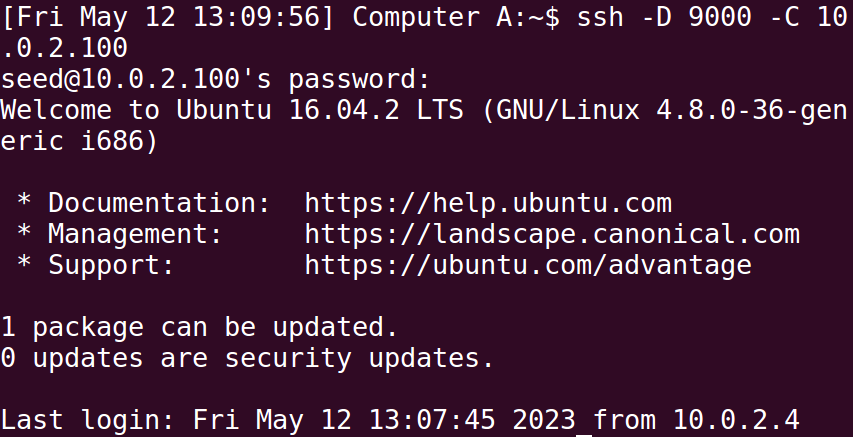
ניתן לראות שהחיבור נותק בהצלחה

ננקה את זיכרון הCACHE של הFIREFOX וננסה להתחבר לאתר סדרות שוב פעם



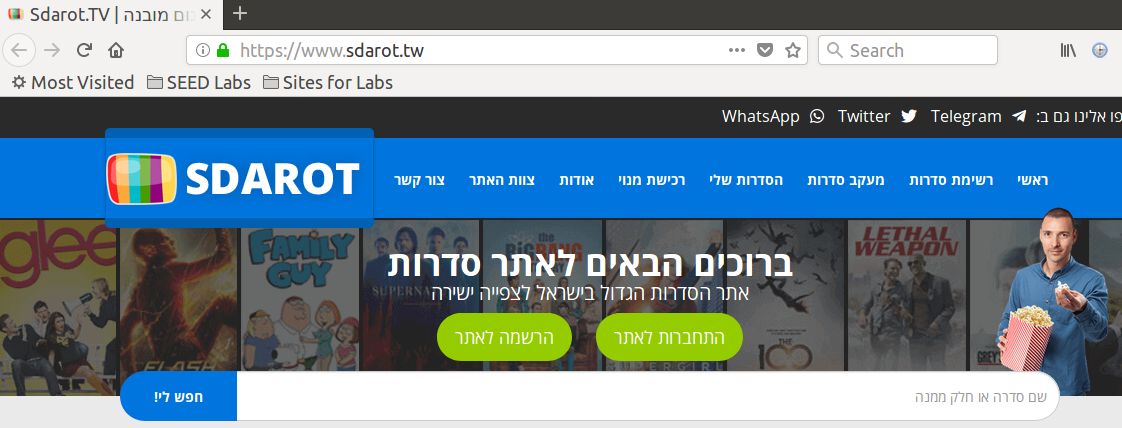
ניתן לראות שהחיבור כעת נכשל מאחר ואין חיבור לשרת הPROXY שהגדרנו בהגדרות הFIREFOX

כעת ניצור את החיבור שוב פעם



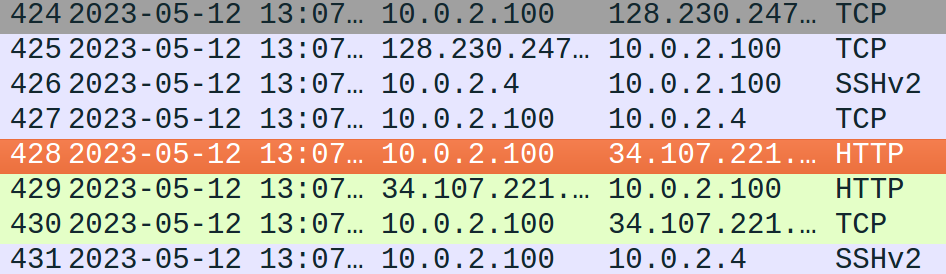
ניתן לראות שהחיבור הצליח

נתחבר לאתר סדרות שוב פעם



ניתן לראות שכעת ניתן לצפות באתר סדרות והצלחנו לבצע מעקף לחוק בחומת האש שהגדרנו שמונע התחברות לאתר סדרות

הקלטנו את הPACKETS שנשלחו בעת ביצוע החיבור לאתר סדרות בעזרת הWIRESHARK



בתמונה מעלה רואים שנוצר חיבור SSH בין מחשב A 10.0.2.4 למחשב B 10.0.2.100

לאחר מכן תעבורת הבקשה לאתר סדרות הועברה ממחשב B לשרתי האתר וחזרה כתשובה בסוף למחשב A כדי שהוא יוכל להתחבר לאתר.

#### סיכום המשימה

תחילה הגדרנו חוקים במחשב A אשר חוסמים יציאת PACKETS ליצירת חיבור TELNET עם מחשב חיצוני וחוסמים PACKETS לאתר ספציפי [www.sdarot.tw](http://www.sdarot.tw)

לאחר מכן יצרנו SSH TUNNEL למחשב B ודרך החיבור הזה הצלחנו במשימה ובצענו מעקף לחסימות שהוגדרו בחומת האש והצלחנו לבצע חיבור TELNET ממחשב A למחשב אחר והצלחנו לבצע חיבור לאתר סדרות דרך מחשב A.

גילינו כיצד ניתן לבצע מעקף לחוקים שהוגדרו בחומת האש, כיצד ניתן לפתוח SSH TUNNEL ואיך היא עוזרת לבצע את המעקף.

גילינו כיצד ניתן להגדיר שתעבורה של אפליקציה תעבור דרך ה SSH TUNNEL שיצרנו בעזרת SOCKS PROXY.

התוצאות התאימו למצופה מאחר והצלחנו להתחבר בTELNET ממחשב A למחשב אחר, וגם הצלחנו להתחבר ממחשב A לאתר סדרות למרות החוקים המוגדרים בחומת האש אשר אוסרים על ביצוע חיבורים אלו.

לא נתקלנו בבעיות במהלך ביצוע המשימה.

### Task 4: Evading Ingress Filtering

### מבוא:

#### תיאור

במשימה זו נרצה לעקוף את החוקים המוגדרים בחומת האש בעזרת REVERSE SSH TUNNEL ולגשת ממחשב מרוחק לאתר פנימי שרץ על מחשב אחר.

#### מטרה

נגדיר REVERSE SSH TUNNEL ממחשב המריץ אתר פנימי שאין למחשבים חיצונים גישה אליו, ובעזרת הTUNNEL שהגדרנו למחשב מרוחק נוכל לגשת מהמחשב המרוחק לאתר הפנימי.

#### תוצאה מצופה

נצליח לגשת מהמחשב המרוחק לאתר הפנימי אשר רץ על מחשב אחר ממנו יצרנו את הreverse ssh tunnel.

#### ביצוע המשימה

תחילה ננסה להיכנס דרך מחשב B לאתר localhost:8000



ניתן לראות שלא רץ שום שרת אינטרנטי פנימי על הפורט 8000 ולכן קיבלנו שהחיבור נכשל

כעת נוסיף חוקים לחומת האש אשר חוסמים מחשבים חיצוניים לשליחת יצירות חיבור SSH למחשב A וחוסמים ממחשב B לשלוח בקשות בפורט 80 למחשב A



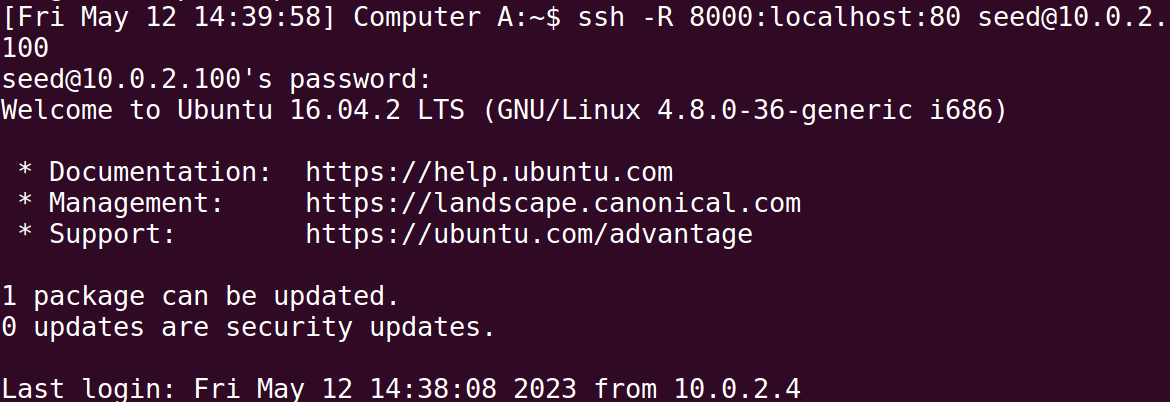
ניתן לראות שהחוקים נוספו בהצלחה לטבלת החוקים INPUT

ננסה ליצור חיבור SSH ממחשב B למחשב A



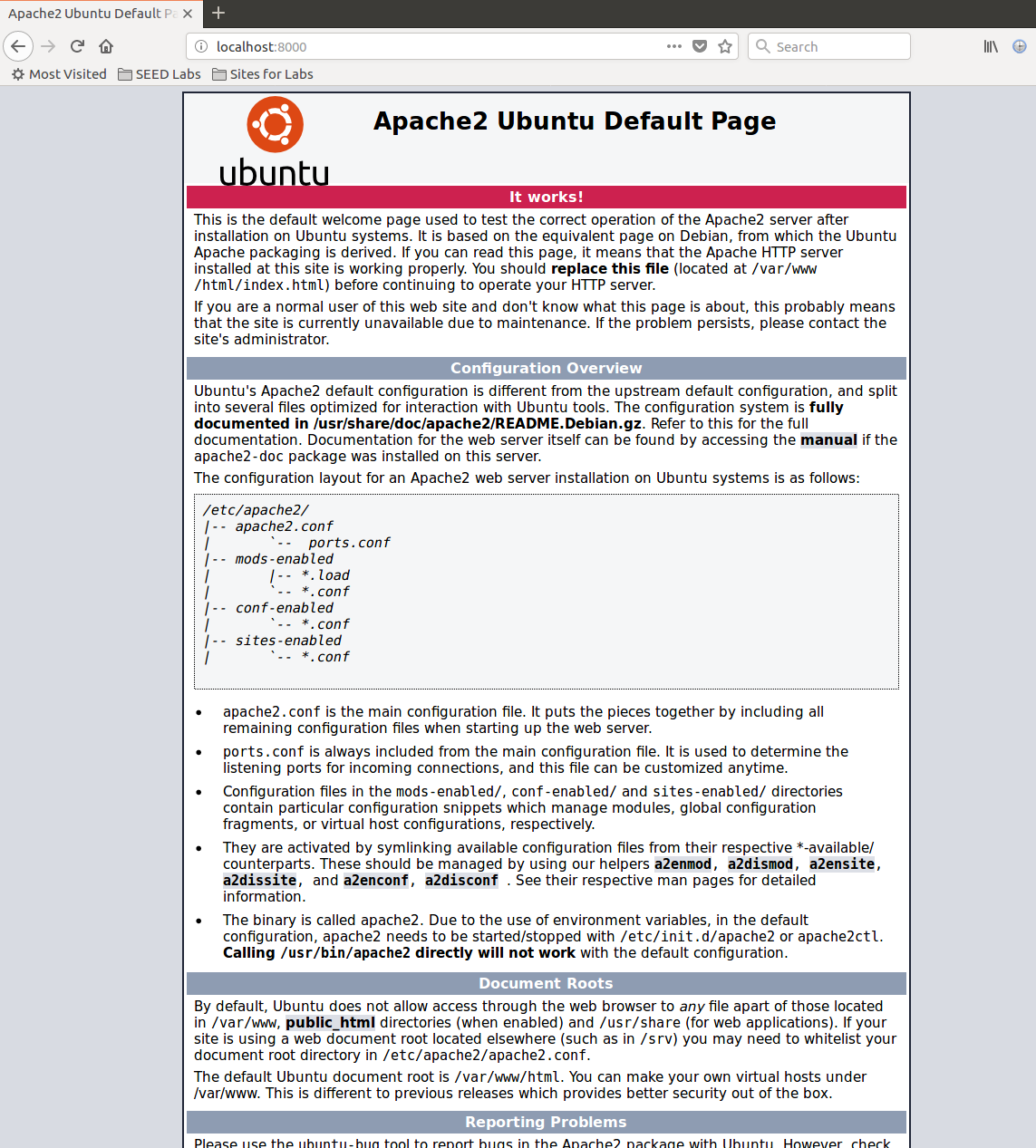
ניתן לראות שיצירת החיבור נכשלה

כעת נגדיר REVERSE SSH TUNNEL במחשב A למחשב B



הreverse ssh tunnel מאפשר למחשב B לשלוח PACKETS למחשב A בפורט 8000 בחיבור SSH שנוצר ממחשב A, ומחשב A יעביר (port forwarding) את הPACKETS האלו לפורט 80 וכך למחשב B תהיה גישה לאתר פנימי אשר רץ על מחשב A למרות החסימות שהגדרנו בחומת האש.

כעת ננסה שוב להתחבר ממחשב B לאתר localhost:8000



ניתן לראות שהחיבור לאתר הפנימי של מחשב A הצליח מאחר והבקשה ששלחנו לפורט 8000 נשלחה דרך הreverse ssh tunnel שיצרנו למחשב A בפורט 80 וכך נכנסנו לאתר הלוקאלי של מחשב A.

#### סיכום המשימה

תחילה הגדרנו חוקים במחשב A אשר חוסמים כניסת PACKETS ליצירת חיבור SSH ממחשבים חיצוניים, וחוסמים PACKETS ממחשב B לפורט 80 של מחשב A.

יצרנו REVERSE SSH TUNNEL ממחשב A אשר ממיר את כל הPACKETS שנשלחים ממחשב B בפורט 8000 למחשב A לפורט 80 ומחזיר נתונים משרתי הWEB.

הצלחנו לבצע את המשימה, מאחר ובעת ניסיון התחברות לאתר הפנימי של מחשב A ממחשב B לפני יצירת הTUNNEL לא הצלחנו לגשת לאתר ולאחר הגדרת הTUNNEL הצלחנו לגשת לאתר הפנימי של A ממחשב B.

גילינו כיצד ניתן לתת גישה ממחשב פנימי למחשב מרוחק לשרתים פנימיים הרצים עליו גם אם קיימת חסימה בחומת האש ליצירת חיבור מהמחשב המרוחק למחשב הפנימי ובכך לאפשר עבודה מרחוק.

התוצאות התאימו למצופה מאחר והצלחנו להתחבר לאתר פנימי שרץ על מחשב A ממחשב מרוחק B אשר לא יכל ליצור חיבור SSH למחשב A.

לא נתקלנו בבעיות במהלך ביצוע המשימה.

**סיכום כללי למעבדה**

תחילה הגדרנו חוקים בחומת האש לחסימות שונות כגון: תקשורת TELNET נכנסת ויוצאת, גישה לאתר ספציפי [www.sdarot.tw](http://www.sdarot.tw) וכו' בעזרת כתיבת פקודות בטרמינל אשר מבוצעות על חומת האש.

והראנו שלפני הגדרת החוקים האלו החיבורים הצליחו ולאחר הגדרת החוקים הגישות נחסמו ולא יכולנו ליצור את החיבורים האלו.

גילינו כיצד להגדיר חוקים לחסימת גישות עבור תקשורת ספציפית ועבור אתר ספציפי.

לאחר מכן, ביצענו הגדרת חוקים לחומת האש על ידי הכנסת מודולים לKERNEL ישירות וטעינתם ללא צורך בביצוע ריסטארט או בנייה מחדש לKENREL.

הגדרנו חמישה חוקים שונים והם:

חסימת TELNET בין A לB

חסימת TELNET בין B לA

חסימת חיבור לאתר [www.sdarot.tw](http://www.sdarot.tw)

חסימת יצירת חיבורי SSH ממחשב A למחשב B

חסימת שליחת PING לאתר [www.sdarot.tw](http://www.sdarot.tw)

לפני טעינת המודול שבנינו לKERNEL ניסיון יצירת החיבורים האלו צלח, ולאחר מכן ניסיון ליצירת החיבורים האלו נכשל בהצלחה.

גילינו כיצד להגדיר חוקים לחסימת גישות בעזרת BACKEND בFIREWALL וכיצד לטעון ולהסיר מודולים מהKERNEL.

כדי להצליח במשימה נעזרנו רבות בדוקומנטציה של NETFILTER בקישור הבא: <https://netfilter.org/documentation/HOWTO/netfilter-hacking-HOWTO-4.html>

רצינו לראות כיצד ניתן לעקוף חוקים שהוגדרו בחומת האש למשל איך לעקוף חסימת ייצור תקשורת TELNET בין מחשבים.

תחילה הגדרנו חוקים במחשב A אשר חוסמים יציאת PACKETS ליצירת חיבור TELNET עם מחשב חיצוני וחוסמים PACKETS לאתר ספציפי [www.sdarot.tw](http://www.sdarot.tw)

לאחר מכן יצרנו SSH TUNNEL למחשב B ודרך החיבור הזה הצלחנו במשימה ובצענו מעקף לחסימות שהוגדרו בחומת האש והצלחנו לבצע חיבור TELNET ממחשב A למחשב אחר והצלחנו לבצע חיבור לאתר סדרות דרך מחשב A.

גילינו כיצד ניתן לבצע מעקף לחוקים שהוגדרו בחומת האש, כיצד ניתן לפתוח SSH TUNNEL ואיך היא עוזרת לבצע את המעקף.

גילינו כיצד ניתן להגדיר שתעבורה של אפליקציה תעבור דרך ה SSH TUNNEL שיצרנו בעזרת SOCKS PROXY.

לבסוף רצינו לראות כיצד ניתן לתת אופציה להתחבר לאתר פנימי במחשב מרוחק על ידי יצירת חיבור מהמחשב הפנימי למחשב המרוחק ונעזרנו ביצירת REVERSE SSH TUNNEL אשר המירה PACKETS מהמחשב המרוחק בפורט ספציפי לפורט 80 במחשב הפנימי.

תחילה הגדרנו חוקים במחשב A אשר חוסמים כניסת PACKETS ליצירת חיבור SSH ממחשבים חיצוניים, וחוסמים PACKETS ממחשב B לפורט 80 של מחשב A.

יצרנו REVERSE SSH TUNNEL ממחשב A אשר ממיר את כל הPACKETS שנשלחים ממחשב B בפורט 8000 למחשב A לפורט 80 ומחזיר נתונים משרתי הWEB.

בעת ניסיון התחברות לאתר הפנימי של מחשב A ממחשב B לפני יצירת הTUNNEL לא הצלחנו לגשת לאתר ולאחר הגדרת הTUNNEL הצלחנו לגשת לאתר הפנימי של A ממחשב B.

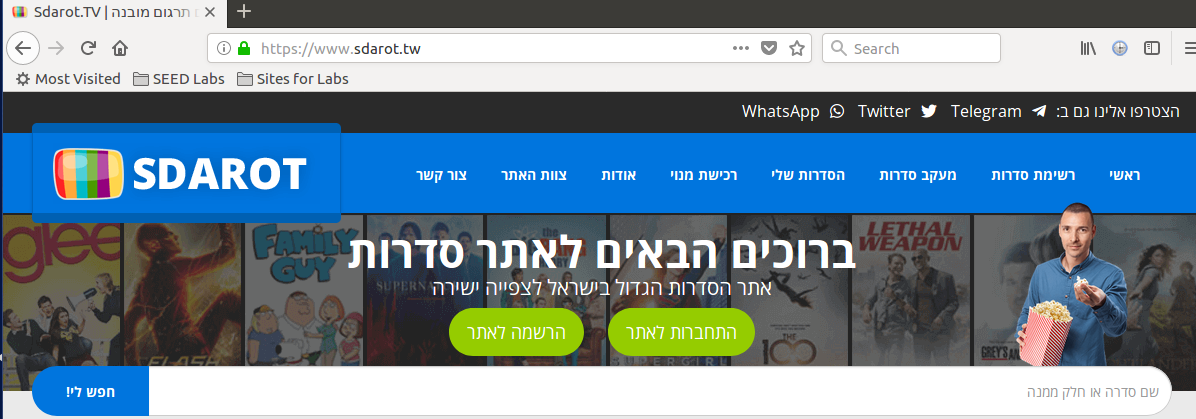
גילינו כיצד ניתן לתת גישה ממחשב פנימי למחשב מרוחק לשרתים פנימיים הרצים עליו גם אם קיימת חסימה בחומת האש ליצירת חיבור מהמחשב המרוחק למחשב הפנימי ובכך לאפשר עבודה מרחוק.

**משהו חדשני:**

חיפשנו דרך נוספת שבעזרתה יהיה ניתן לעקוף את החוקים אותם אנו מגדירים בחומת האש, ומצאנו שניתן לעקוף את החוקים בעזרת דפדפן TOR אשר מצפין את הDATA של הPACKET ועל ידי כך מקשה על הFIREWALL לזהות את הנתונים של הPACKET אותם הוא בודק לצורך השוואה מול החוקים ומצליח לעקוף את החוקים שאמורים לחסום את שליחתה.

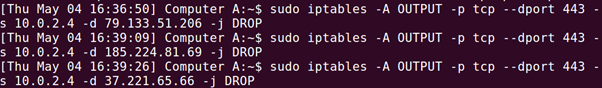
איך הוכחנו זאת?

נכנסו לאתר סדרות

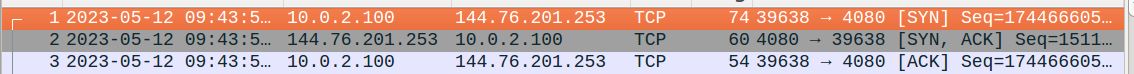


ראינו שניתן לגשת לאתר

לאחר מכן, הגדרנו חוקים לחסימת גישה לאתר

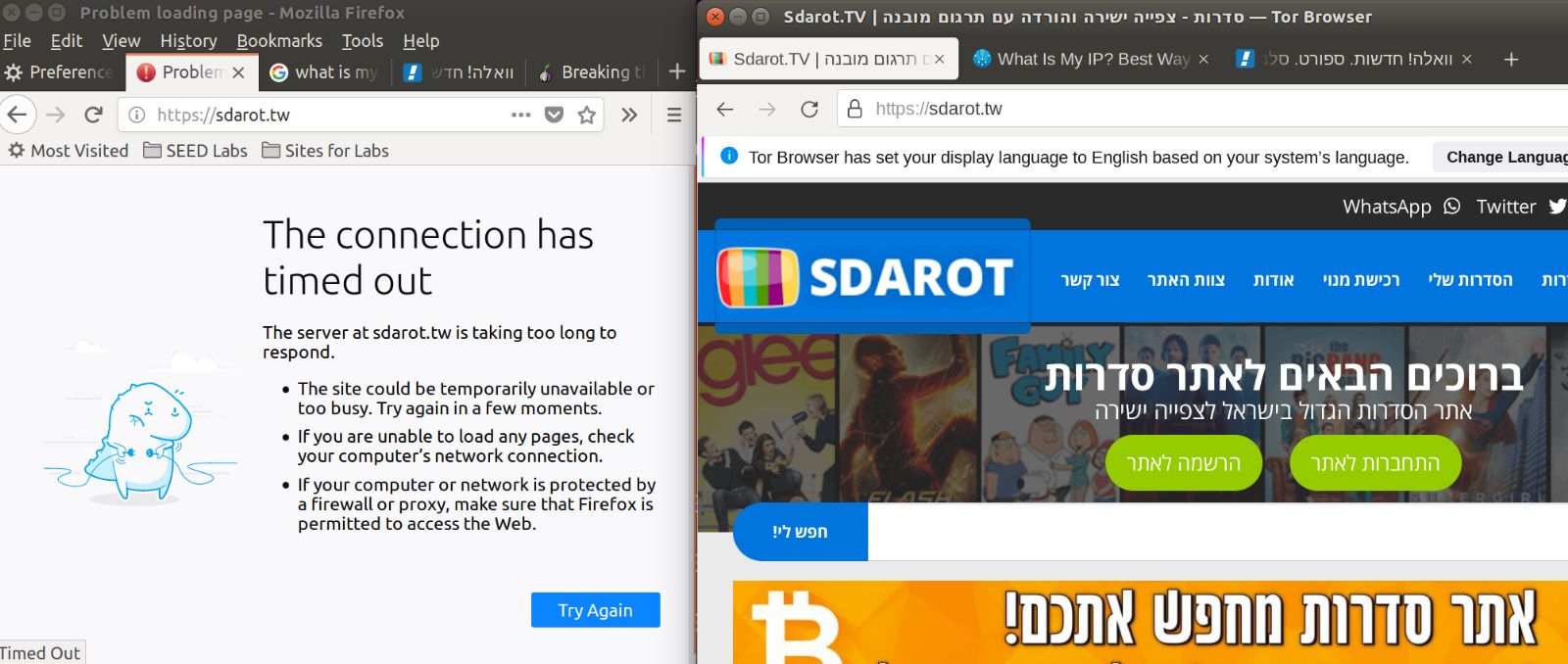


נכנסו לדפדפן TOR

TOR יצר חיבור מאובטח עם מחשב IP 144.76.201.253

ניתן לראות שהחיבור נוצר בהצלחה בין מחשב 10.0.2.10 ל144.76.201.253

חקרנו וגילינו שTOR יצר חיבורים נוספים לאורך הדרך מIP שונים, TOR מספק הצפנה למידע המועבר וחשאיות כך שלא יהיה ניתן לעקוב אחר השולח. במקרה שלנו TOR הצפין את הבקשה לגלוש לאתר [www.sdarot.tw](http://www.sdarot.tw) וניגש אל האתר באמצעות IP אחר ובעקבות זאת הFIREWALL לא הצליח לחסום את בקשת המחשב לגשת לאתר [www.sdarot.tw](http://www.sdarot.tw) כפי שהוגדר לו בחוקי טבלת הOUTPUT כי הDATA היה מוצפן והFIREWALL לא יכל לבצע השוואה עם החוקים.

ניתן לראות שדרך הדפדפן השמאלי של FIREFOX לא היה ניתן לגשת לאתר סדרות אך דרך הדפדפן הימני של TOR הצלחנו לגשת לאתר סדרות בהצלחה.