

דו"ח מעבדה- תרחיש מס' 3

פרטים:

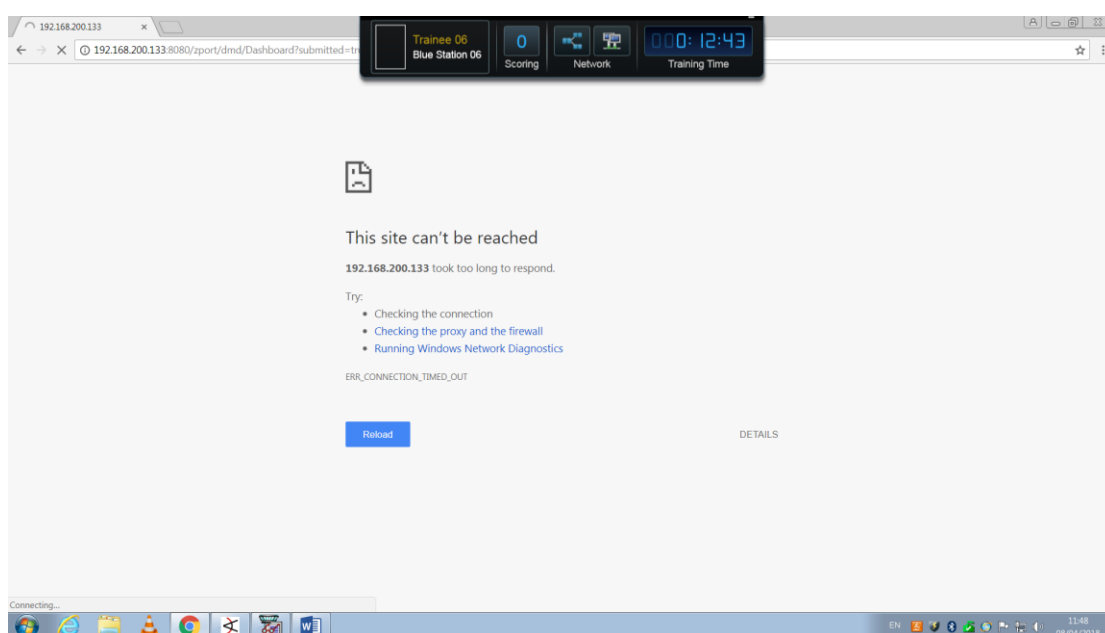
מגיש: שגיא סעדה

תאריך: 25/03/2018

שם התרחיש: הרעלת ה-DNS המקומי של הארגון והפלת שרתים חיוניים.

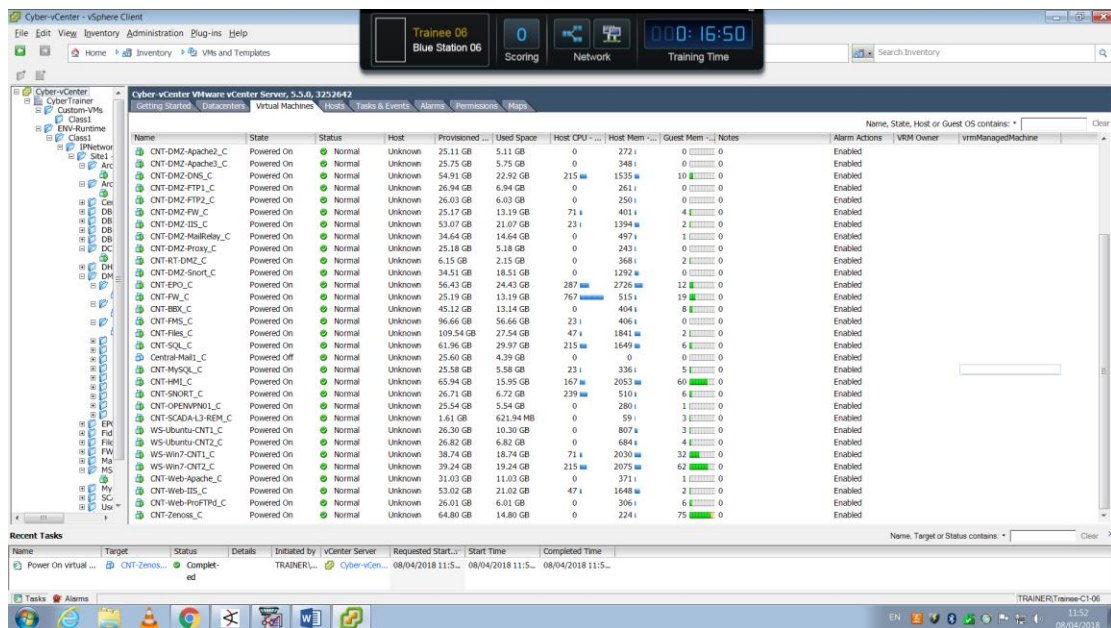
תהליך ההתקפה:

בתהליך ההתקפה זהינו בהתחלה נפילה של השרת Zenoss בארגון.

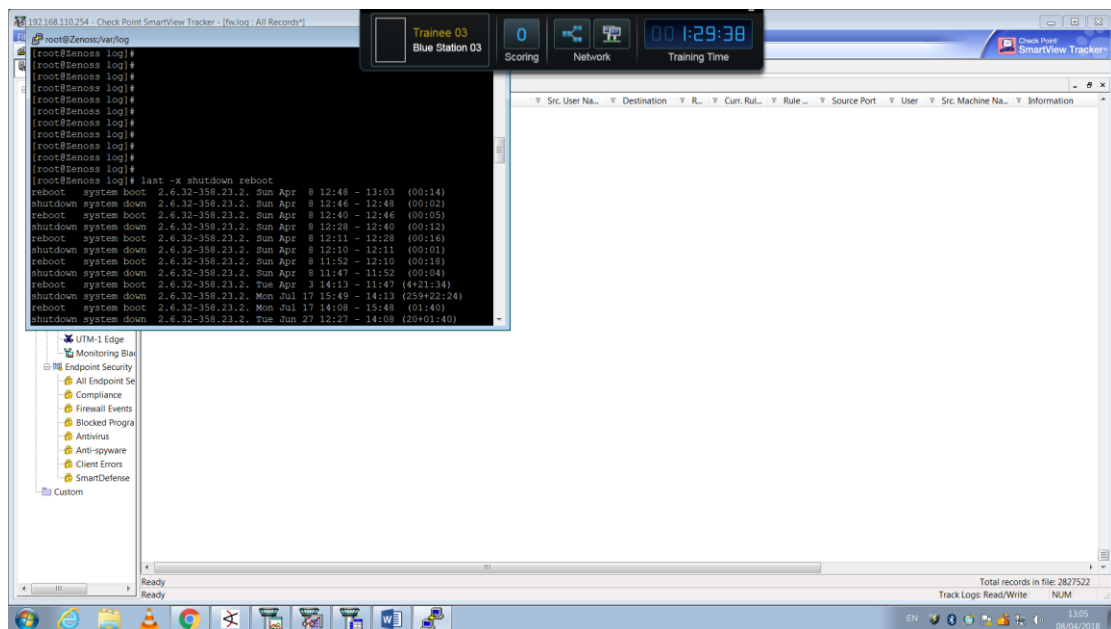


אייפי של השרת שמחזיק את Zenoss – 192.168.200.133

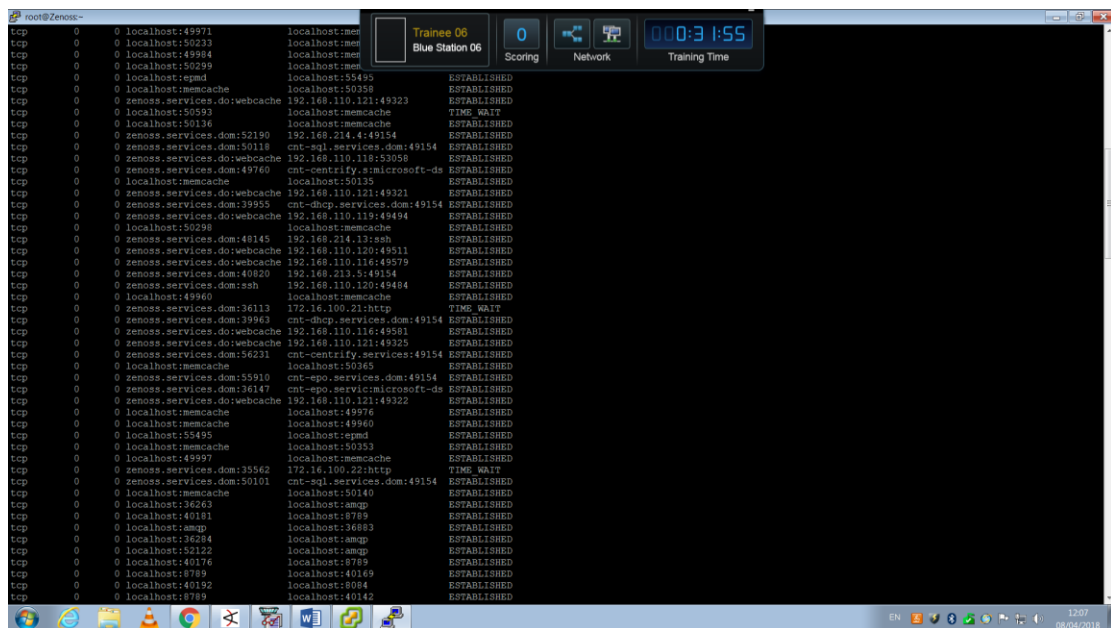
ניסינו להתחבר אליו מרחוק, דרך Putty, ולא הצלחנו – קיבלנו Timeout. (מכאן ניתן להסיק שהשרת נפל לגמרי – לא רק השירות שלו – אלה אינו פועל כלל).



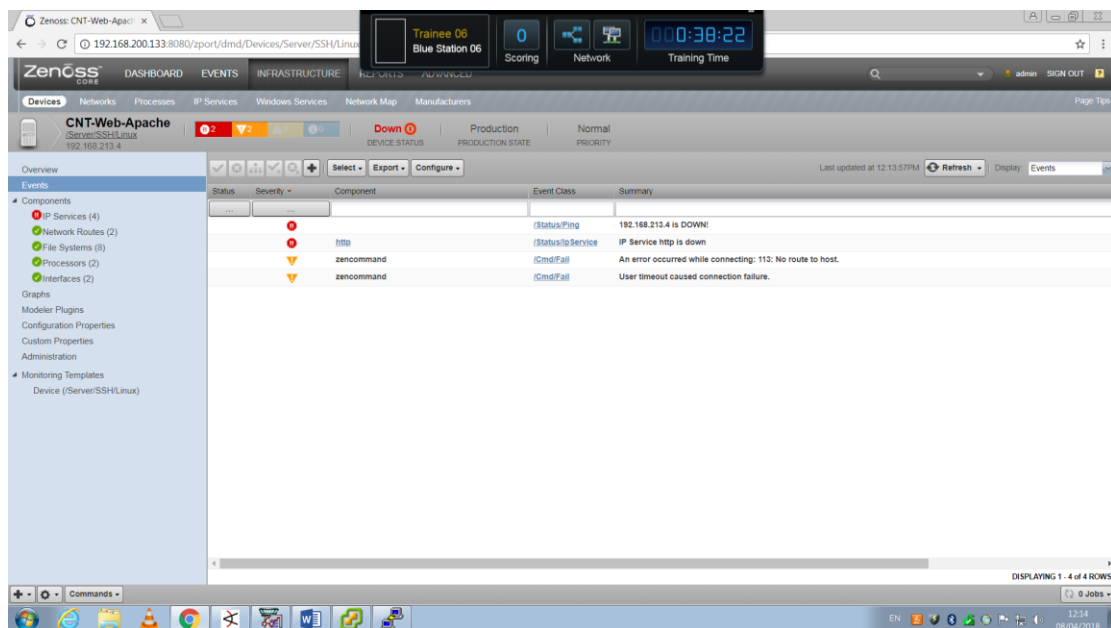
התחברנו דרך ה- Putty לשרת ה- Zenoss באמצעות ה- ssh והתחלנו לחקור קבצי לוגים. ראינו את זמני נפילת שרת ה- Zenoss לאורך ההתקפה – (בהמשך ההתקפה השרת נפל שוב לאחר שהרמנו אותו כמה וכמה פעמים).

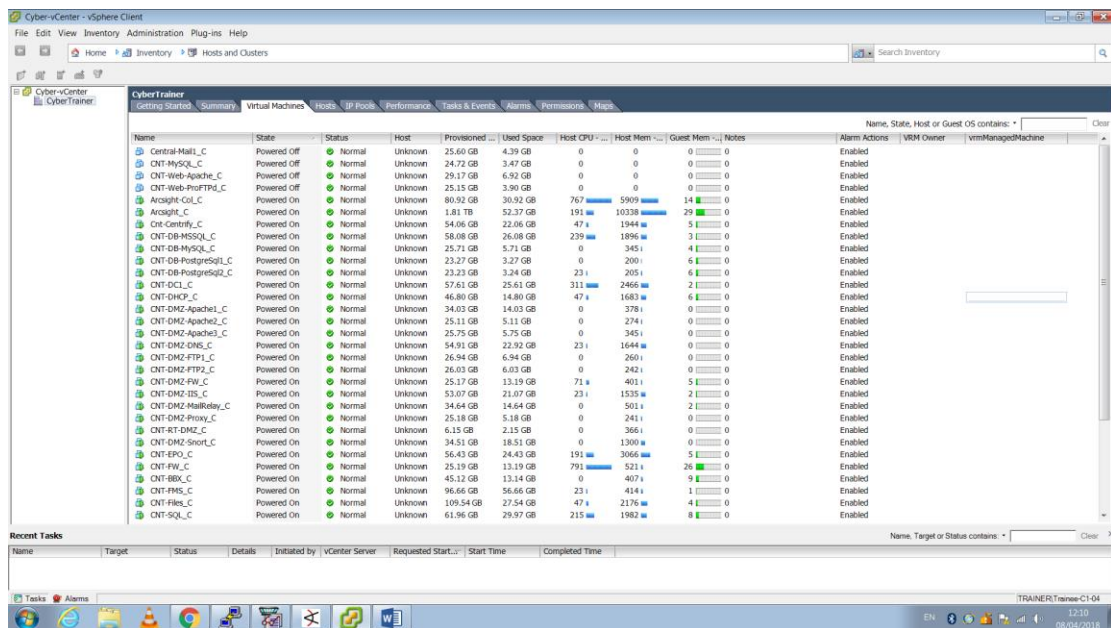


אך לא מצאנו משהו נוסף שיכול לעזור לנו. הרצנו – netstat אך אי אפשר להבין כלום מפני שהשירות השרת נותן רלוונטי לכל הרשת ולכן יהיה קשה לאתר משהו יוצא דופן.



במקביל לחיפושים אלו ראינו ב- Zenoss וכמוכן גם ב- vSphere שעוד שרתים נופלים לנו בארגון –

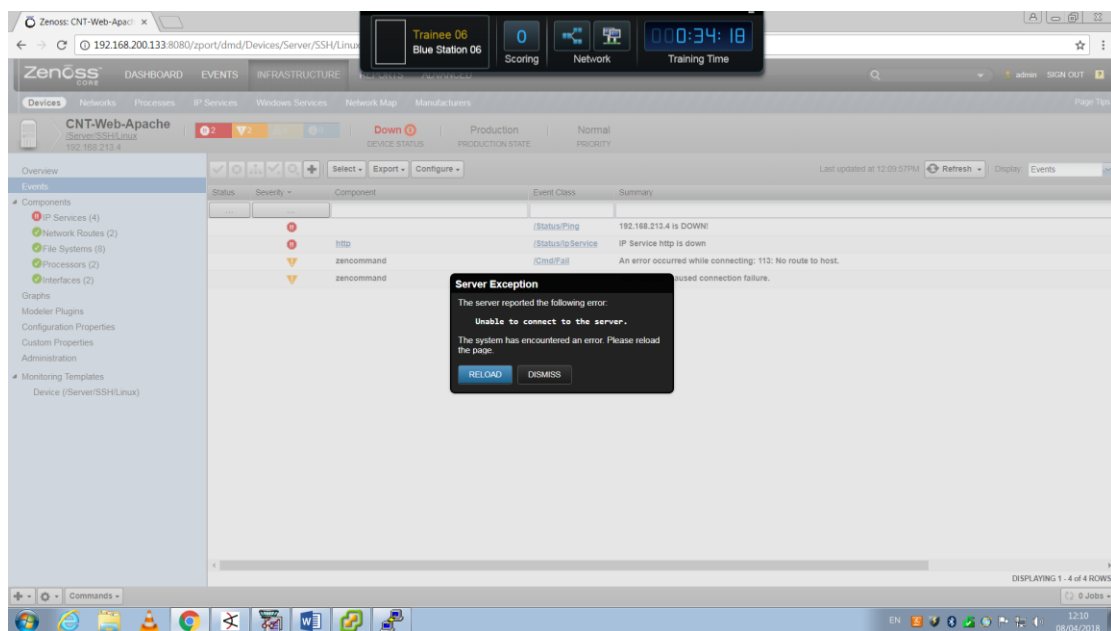




השרת שראינו עליו התראה ב-Zenoss לפני שהוא נפל שוב היה – CNT-Web-Apache
האייפי שלו – 192.168.213.4

ושאר ההתראות שראינו ב-vSphere היו עבור נפילתם של השרתים –
CNT-Web-, CNT-Web-PrtoFTPd_C, CNT-MySQL_C, Central-Mail1_C
Apache_C.

תוך כדי הסתכלות שרת ה-Zenoss נופל שוב.



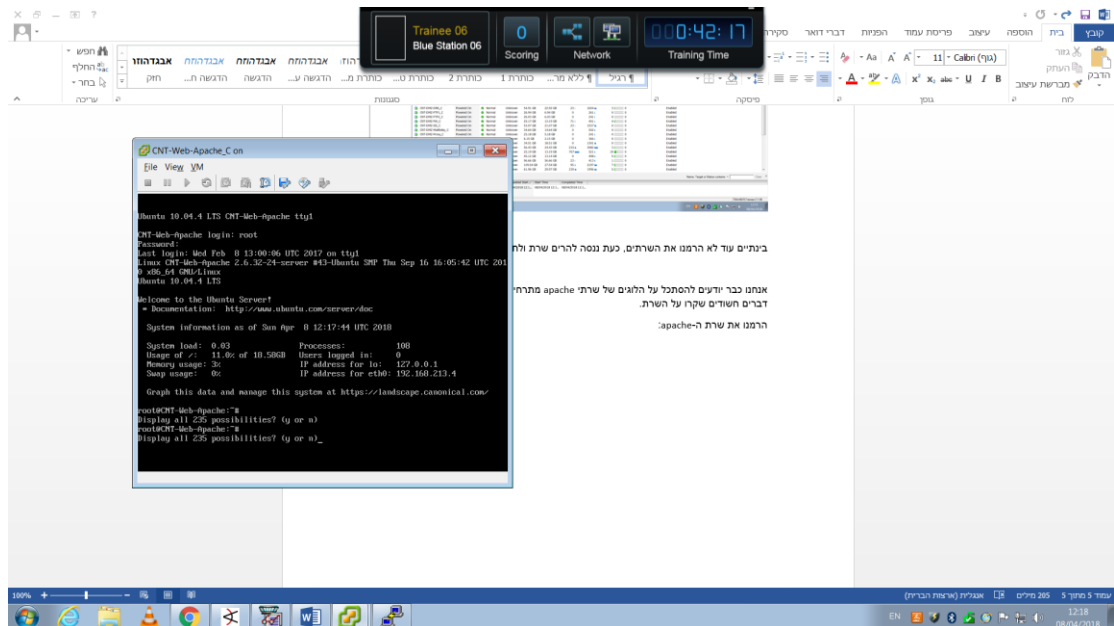
ייתכן שמישהו עושה משהו בתוך הרשת שלנו ולכן הוא מפיל את ה-Zenoss כדי שלא יהיה לנו
חיווי על מה שקורה ברשת. (אך למזלנו יש את vSphere).

התוקף מנסה ככל הנראה להפיל את השרתים העיקריים בארגון.

כעת ננסה להרים שרת ולחפש מה גרם לו ליפול בלוגים שלו (כמו שניסינו ב-Zenoss, אולי זה מה שייתן לנו כיוון).

אנחנו יודעים להסתכל על הלוגים של שרתי Apache מתרחישים קודמים ולכן ננסה לאתר שם דברים חשובים שקרו על השרת.

הרמנו את שרת ה-Apache



כמה דברים שהעלנו לדיון תוך כדי –

- אולי יש מישהו חיצוני (מרשת האינטרנט) שתוקף אותנו ומפיל שרתים דרך שרתי ה-DMZ ומשם מתקשר ל-Zenoss עצמו. (בהמשך נבדוק את האפשרות הזאת)
- פריצה כלשהי ל-Firewall (בהמשך נבדוק את האפשרות הזאת)
- תקשורת יוצאת מתוך הרשת החוצה ולכן ערוץ התקשורת פתוח בין נקודת האחיזה בתוך הרשת אל מחוץ לה.

תהליך הזיהוי :

בתהליך הזיהוי ננקטו הצעדים הבאים על מנת להעמיק את הבדיקה של האירוע –

נכנסו ל- Tracker על מנת לבדוק תקשורת בין שרתים בארגון, מפני שלא קפצו התראות ב- ArcSight הסקנו 2 דברים,

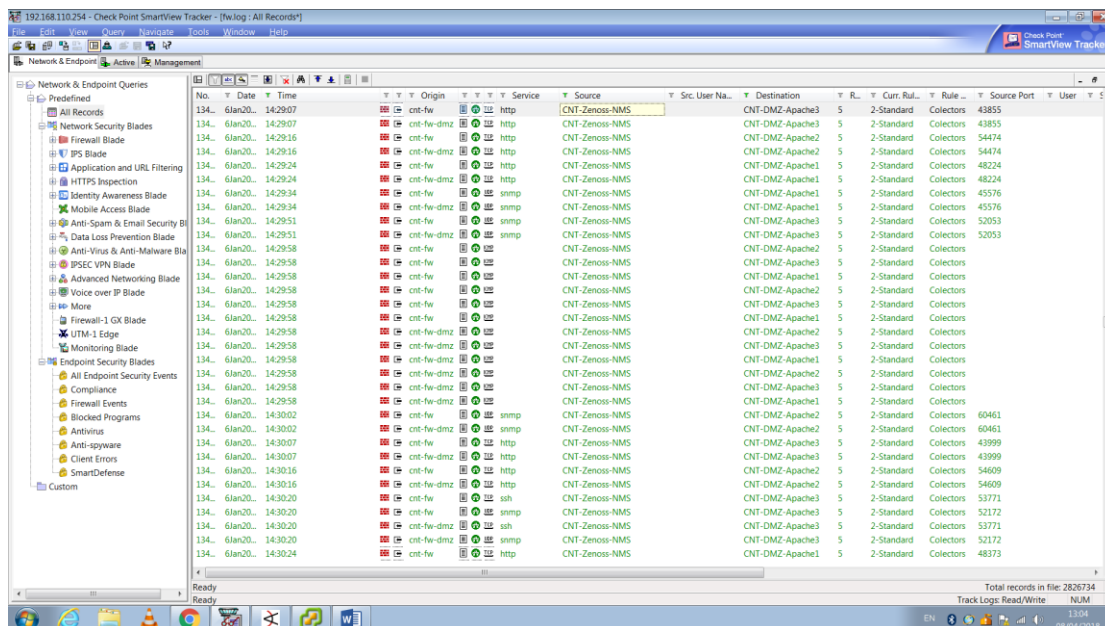
האחד – התוקף תקף באמצעות דרכים בהן הכלי ArcSight אינו מזהה משהו חריג בכך שלא סרק פורטים בצורה בה החוקים כתובים או ניחש סיסמאות שונות וכדומה. (התוקף "עקף" את החוקים)

השני – התוקף החדיר תוכנה זדונית כלשהי לאחד המחשבים בארגון וכך התקשורת אינה מוגדרת כחשודה לכלי ArcSight.

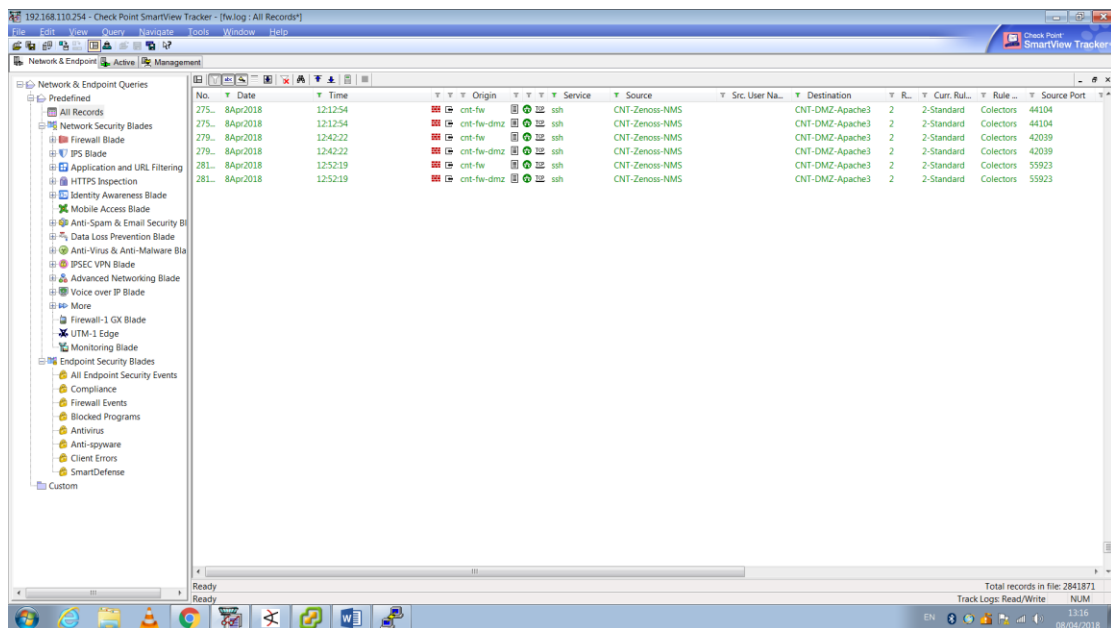
לכן, נכנסו ל- Tracker ובדקנו תקשורת שנראתה חשודה בה ה- Zenoss מופיע. (לא רק תקשורת שמסומנת באדום).

הכנסנו את השרת ליעד ו/או למקור בעזרת פילטור וכך ראינו מי מתקשר עם השרת ולהפך.

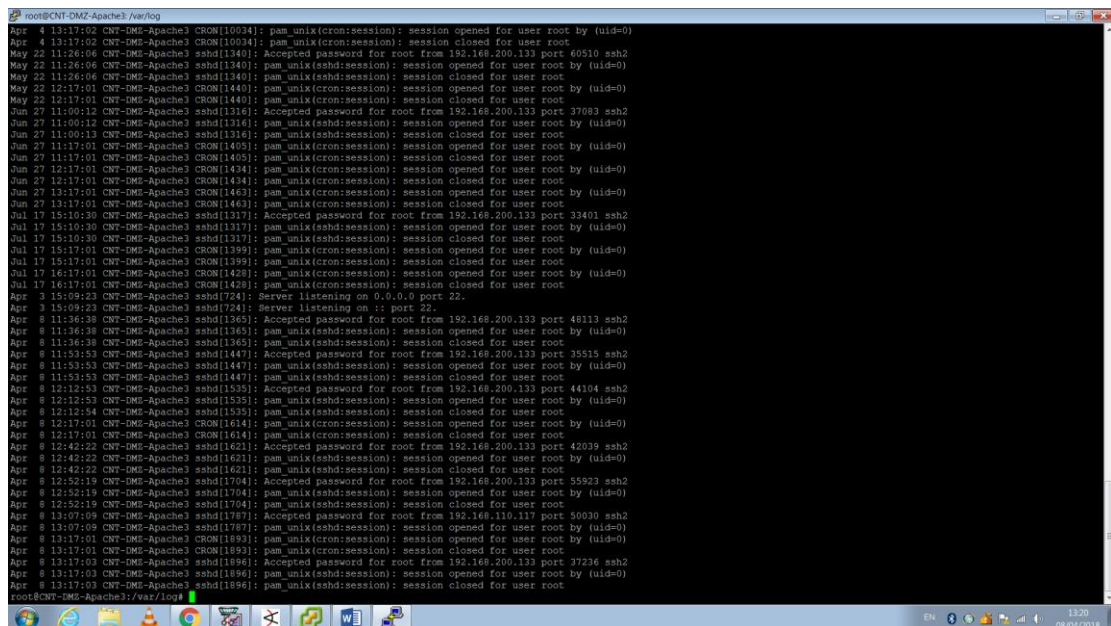
מכאן ראינו תקשורת חשודה באמצעות ssh שלא אמורה לקרות –



No.	Date	Time	Origin	Service	Source	Src. User Na...	Destination	R...	Cur. Ru...	Rule ...	Source Port	User
134...	6Jan20...	142907	cnt-fw	http	CNT-Zenoss-NMS		CNT-DMZ-Apache3	5	2-Standard	Collectors	43855	
134...	6Jan20...	142907	cnt-fw-dmz	http	CNT-Zenoss-NMS		CNT-DMZ-Apache3	5	2-Standard	Collectors	43855	
134...	6Jan20...	142916	cnt-fw	http	CNT-Zenoss-NMS		CNT-DMZ-Apache2	5	2-Standard	Collectors	54474	
134...	6Jan20...	142916	cnt-fw-dmz	http	CNT-Zenoss-NMS		CNT-DMZ-Apache2	5	2-Standard	Collectors	54474	
134...	6Jan20...	142924	cnt-fw	http	CNT-Zenoss-NMS		CNT-DMZ-Apache1	5	2-Standard	Collectors	48224	
134...	6Jan20...	142924	cnt-fw-dmz	http	CNT-Zenoss-NMS		CNT-DMZ-Apache1	5	2-Standard	Collectors	48224	
134...	6Jan20...	142934	cnt-fw	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache1	5	2-Standard	Collectors	45576	
134...	6Jan20...	142934	cnt-fw-dmz	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache1	5	2-Standard	Collectors	45576	
134...	6Jan20...	142951	cnt-fw	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache3	5	2-Standard	Collectors	52053	
134...	6Jan20...	142951	cnt-fw-dmz	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache3	5	2-Standard	Collectors	52053	
134...	6Jan20...	142958	cnt-fw	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache2	5	2-Standard	Collectors		
134...	6Jan20...	142958	cnt-fw-dmz	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache2	5	2-Standard	Collectors		
134...	6Jan20...	142958	cnt-fw	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache1	5	2-Standard	Collectors		
134...	6Jan20...	142958	cnt-fw-dmz	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache1	5	2-Standard	Collectors		
134...	6Jan20...	142958	cnt-fw	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache2	5	2-Standard	Collectors		
134...	6Jan20...	142958	cnt-fw-dmz	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache2	5	2-Standard	Collectors		
134...	6Jan20...	142958	cnt-fw	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache3	5	2-Standard	Collectors		
134...	6Jan20...	142958	cnt-fw-dmz	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache3	5	2-Standard	Collectors		
134...	6Jan20...	143002	cnt-fw	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache1	5	2-Standard	Collectors	60461	
134...	6Jan20...	143002	cnt-fw-dmz	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache1	5	2-Standard	Collectors	60461	
134...	6Jan20...	143007	cnt-fw	http	CNT-Zenoss-NMS		CNT-DMZ-Apache3	5	2-Standard	Collectors	43999	
134...	6Jan20...	143007	cnt-fw-dmz	http	CNT-Zenoss-NMS		CNT-DMZ-Apache3	5	2-Standard	Collectors	43999	
134...	6Jan20...	143016	cnt-fw	http	CNT-Zenoss-NMS		CNT-DMZ-Apache2	5	2-Standard	Collectors	54609	
134...	6Jan20...	143016	cnt-fw-dmz	http	CNT-Zenoss-NMS		CNT-DMZ-Apache2	5	2-Standard	Collectors	54609	
134...	6Jan20...	143020	cnt-fw	ssh	CNT-Zenoss-NMS		CNT-DMZ-Apache3	5	2-Standard	Collectors	53771	
134...	6Jan20...	143020	cnt-fw-dmz	ssh	CNT-Zenoss-NMS		CNT-DMZ-Apache3	5	2-Standard	Collectors	53771	
134...	6Jan20...	143020	cnt-fw	snmp	CNT-Zenoss-NMS		CNT-DMZ-Apache3	5	2-Standard	Collectors	52172	
134...	6Jan20...	143024	cnt-fw	http	CNT-Zenoss-NMS		CNT-DMZ-Apache1	5	2-Standard	Collectors	48373	



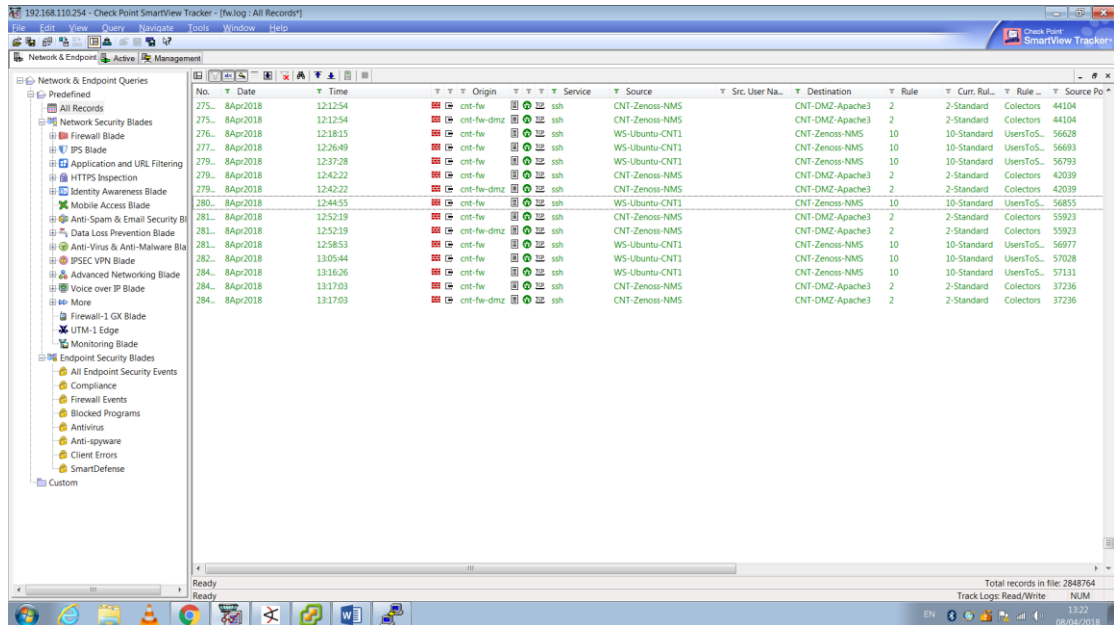
ניתן לראות תקשורת באמצעות ssh בין Zenoss ל- Apache3.
 מכאן – התחברנו לשרת Apache3 באמצעות ssh (בעזרת Putty) על מנת לחקור את השרת ולנסות להבין קצת יותר לעומק לגבי התקשורת החשודה שזיהינו.
 מהקובץ auth.log ניתן לראות –



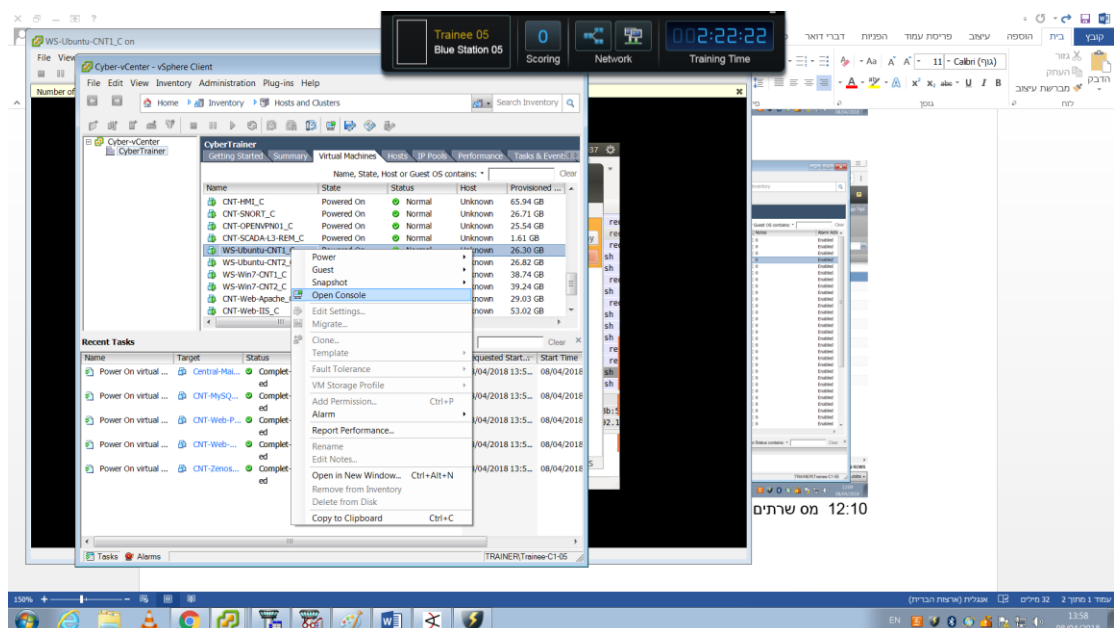
שיחות שנפתחות בין Zenoss ל- Apache3 – פניות אלו בוצעו בעזרת סיסמא והתקבלו על ידי השרת.
 לאחר חקירת השרת – לא הגענו למסקנות כלשהן שיכולות לעזור.

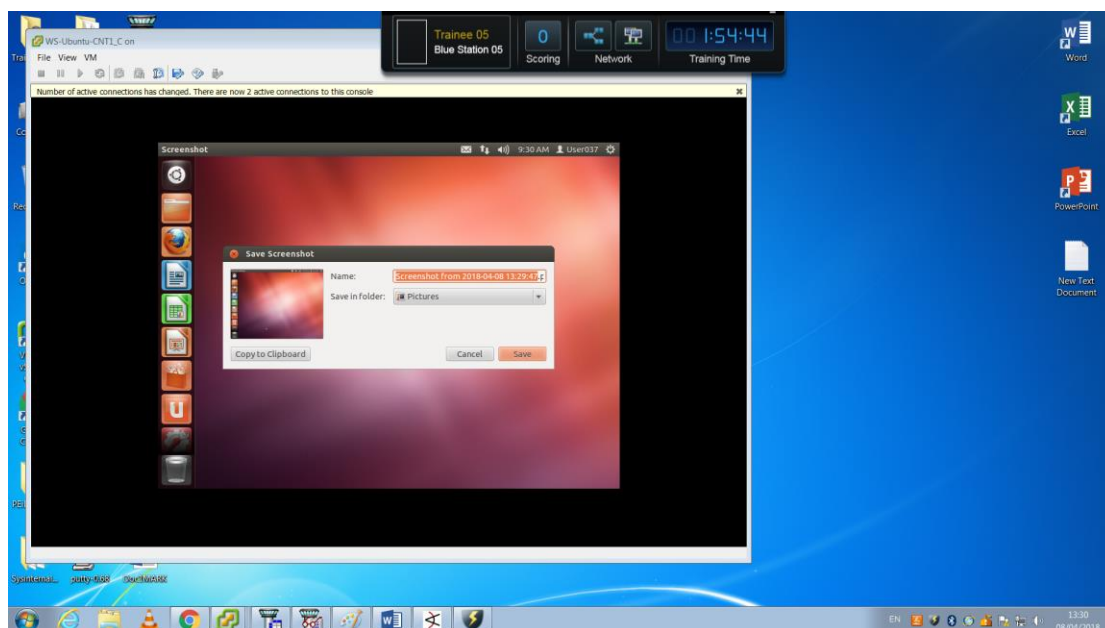
המשכנו להסתכל ב- Tracker ולנסות כל מיני אפשרויות פילטור עם הכנסת שרת ה- Zenoss לאחד מהאפשרויות (מקור/יעד).

ומצאנו תקשורת נוספת באמצעות ssh בין WS-Ubuntu-Cnt1 ל- Zenoss – מכאן, התחלנו לחשוד שהתקשרות הזאת אינה תקינה.

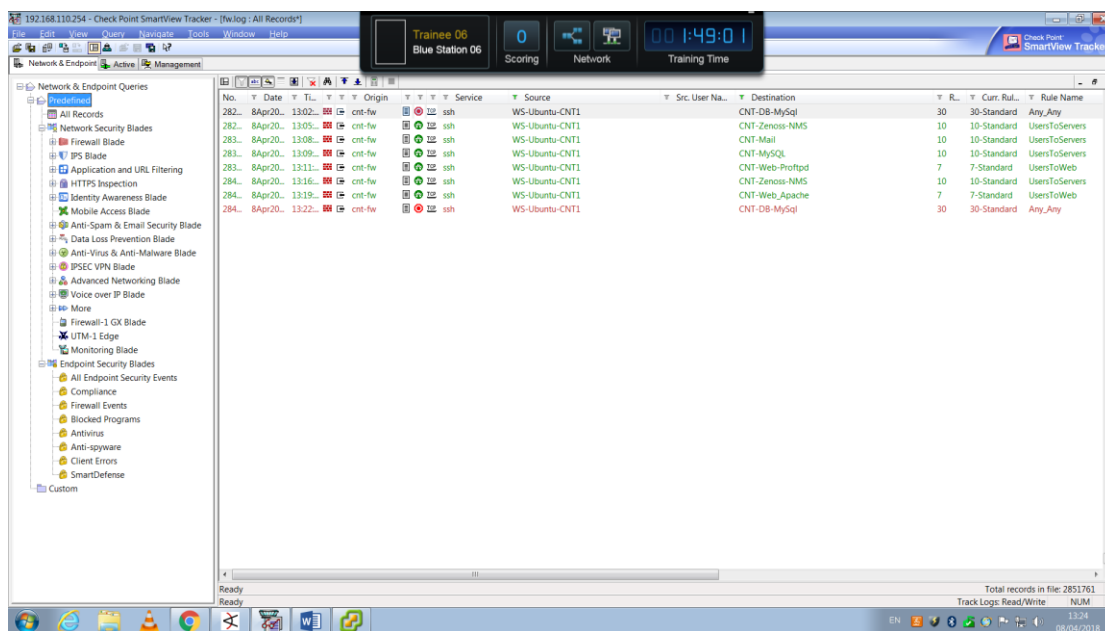


התחברנו מרחק למחשב של העובד (Cnt1) –





גם בעזרת ה-GUI (ממשק למשתמש) וגם בעזרת Putty – ssh.
 מחשב זה – עובד כלשהו בחברה (עמדה בסגמנט העובדים) פונה ב-ssh לשרת CNT-DB-MySQL – גם דבר שאינו לגיטימי ברמת העיקרון בחברה שלנו.



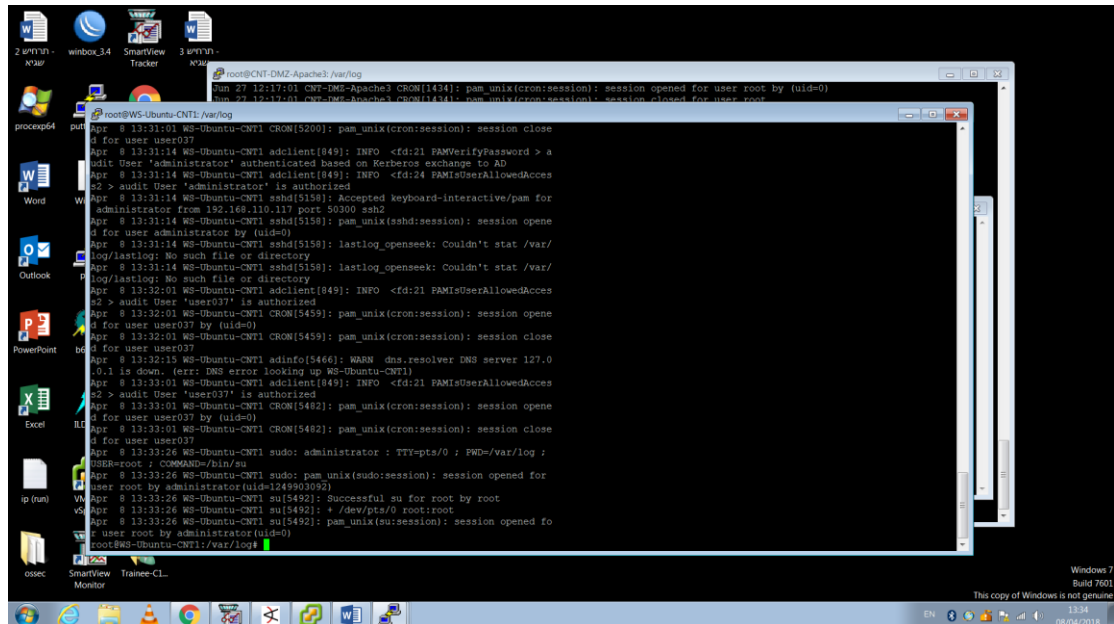
ניתן לראות שעמדה זו פונה לעוד שרתים בארגון שלנו באמצעות ה-ssh בצורה מחשודה (כי זה לא אמור לקרות).

מסקנה - עמדה זו פונה לכל השרתים שנפלו בארגון. (עמדה זו היא החשודה ביותר בשלב זה).

מכאן – צריך לחקור את העמדה על מנת לפענח דברים נוספים. (בדיקת היסטורית גלישה, היסטורית הורדות, קבצי לוגים, סריקת תהליכים וכו').

תחילת חקירת העמדה –

- קובץ auth.log של העמדה –

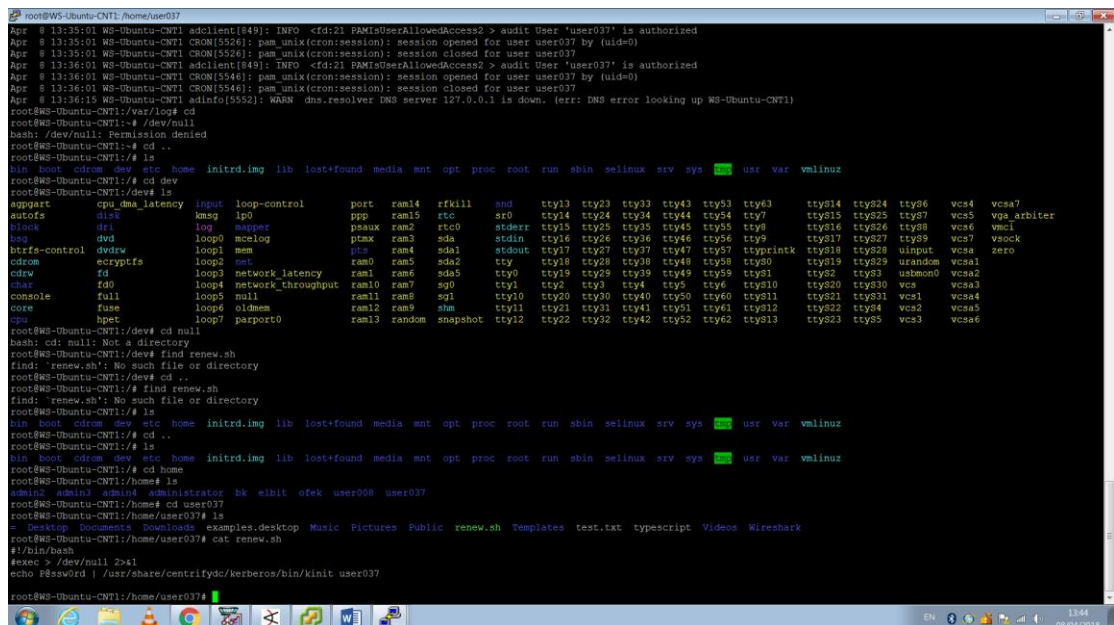


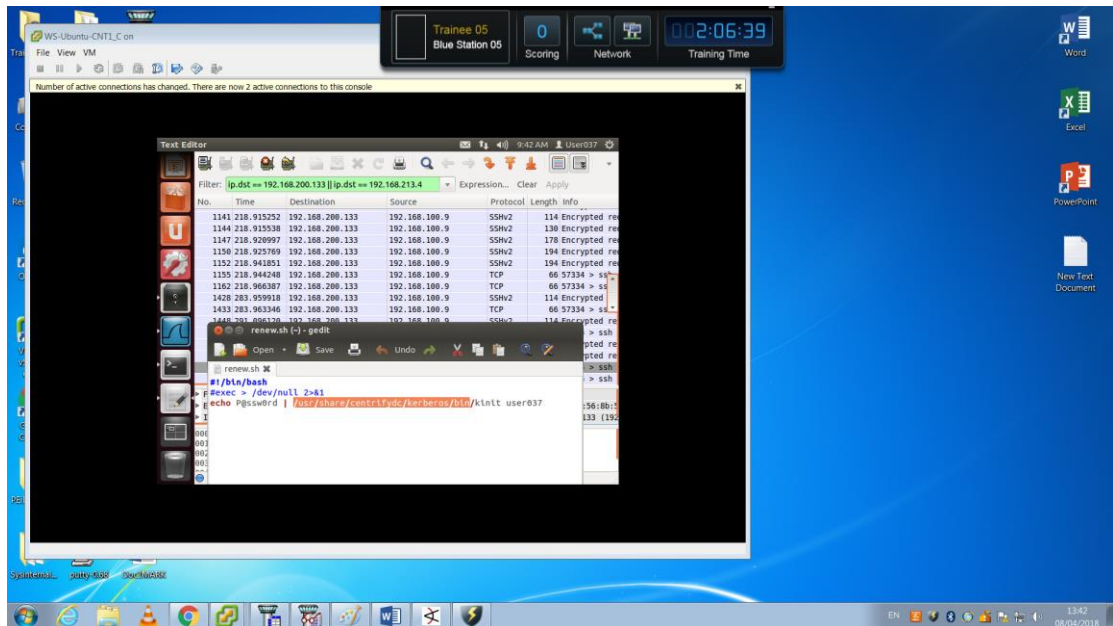
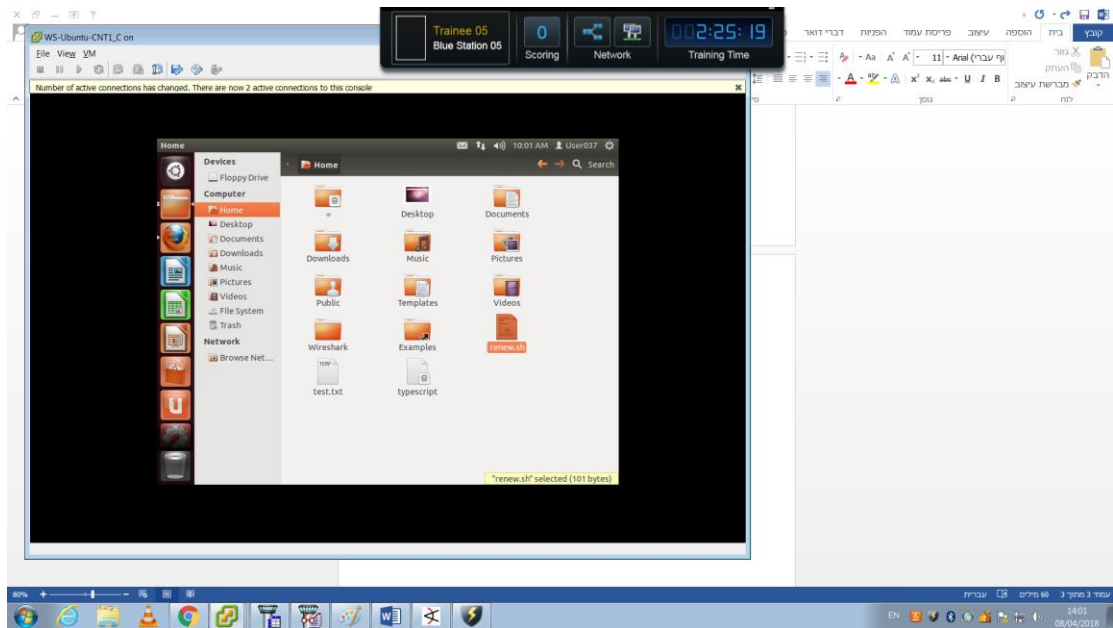
ניתן לראות פתיחות שיחה ואת סגירתם.

- הסתכלנו על היסטוריית הדפדפן ומצאנו קובץ player.jar
- בנוסף, מצאנו קישור ל- Youtube (אולי הייתה כאן פעילות זדונית – Man in the middle

פתחנו WireShark והתחלנו להסניף את התקשורת בעמדה – לא ראינו שום דבר חשוד.

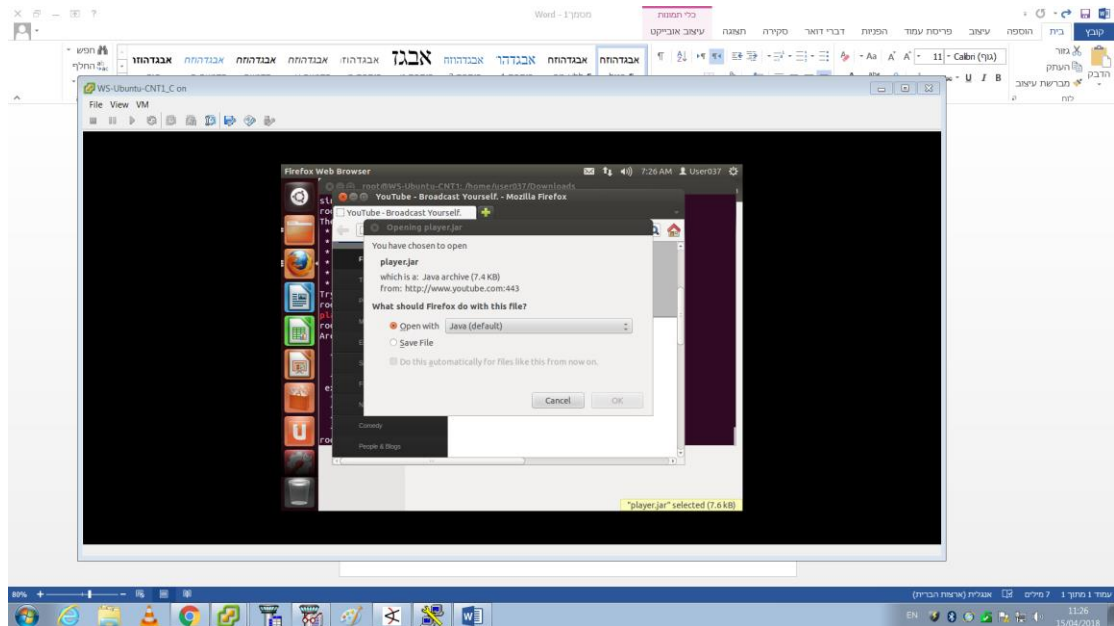
נכנסו לתיקיית home על העמדה וראינו שיש קובץ שם renew.sh.





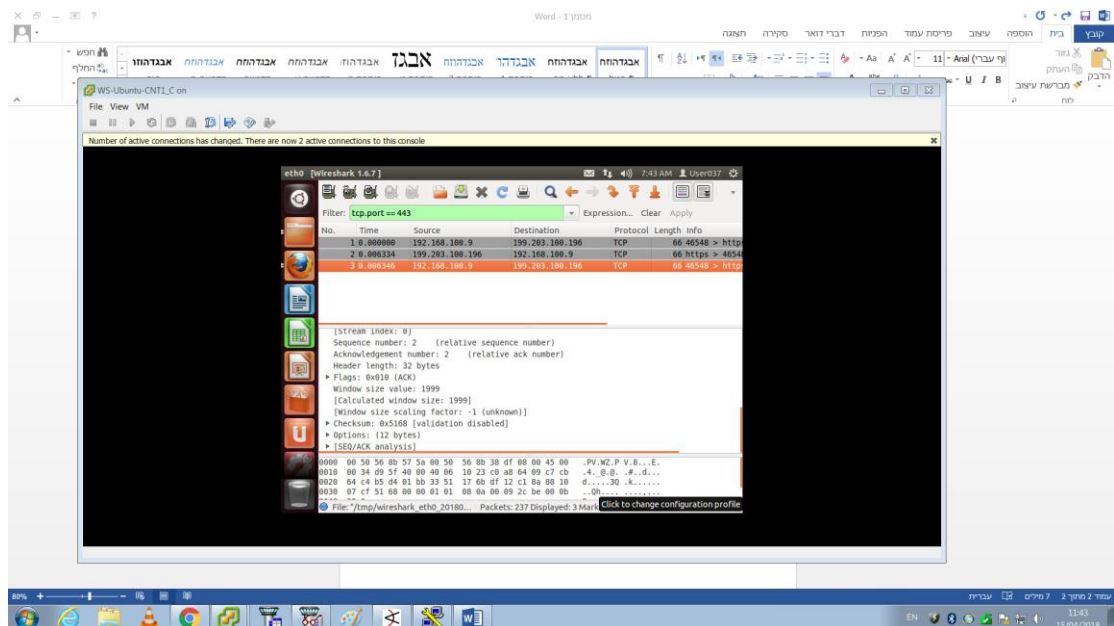
אנחנו יודעים שבקובץ הזה כתוב שכל stdout ו- stderr נשלחים אל איזשהו חור שחור .dev/null

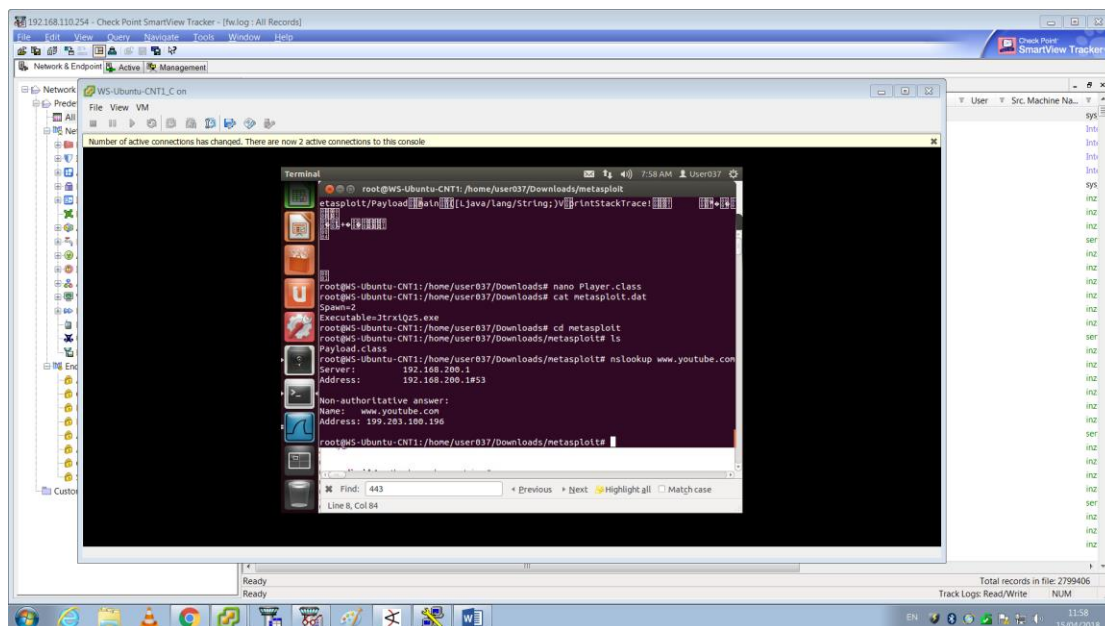
הנחה - הקובץ player.jar הוא קובץ זדוני ולכן נבדוק אותו ואת המקור שלו.
מקור הקובץ – בתהליך הגלישה של העובד לאתר youtube היה חלון שביקש הורדה של קובץ player, מאחר והמקור נראה אמין, בוצעה ההורדה והקובץ נשמר במחשב.



מכאן נשאלת השאלה – איך יכול להיות שמישהו גלש ל- youtube וקיבל בקשה להורדת קובץ מסוג זה?

תמונה שמציגה את התקשורת בין העמדה לתוקף.





מדובר בהתקפת **DNS Spoofing** (הרעלת DNS) – מתקפה אשר מזריקה נתונים שגויים של שמות שרתים וכתובות IP המקושרים אליהם, כל מערכת DNS, או אל זיכרון המטמון שלה. הדבר גורם להפניית בקשת דף אינטרנט מסוים אל אחר, כתוצאה מהחזרת כתובת IP שאינה משויכת לשרת המקורי.

מקור -

https://he.wikipedia.org/wiki/%D7%94%D7%A8%D7%A2%D7%9C%D7%AA_DNS

בארגון שלנו, ישנם 2 DNSים – אחד ב-DMZ, הוא ה-DNS שפונים אליו כדי לקבל את האתרים של הארגון. ה-DNS המהימן ללקוחות מבחוץ (top levels ו-roots מכירים אותו). ה-DNS השני נמצא ב-DC שלנו, כלומר ה-DC ברשת שלנו מתפקד גם כשרת DNS.

בתהליך ה-Poisoning מה שקורה הוא שהלקוח פונה לשרת DNS מקומי ומבקש אתר. אם השרת המקומי מכיר, מחזיר תשובה. אם הוא לא מכיר, נשלחת בקשה באופן רקורסיבי (בדרך כלל מה שקורה ב-Poisoning) לשרת ה-ROOT DNS (מתבסס על כך שבקשות DNS אינן מוצפנות ולכן התוקף יכול לראות את הבקשות בהנחה שלא נעשה שימוש ב-1.1.1.1), בחלון הזמן של הבקשה הרקורסיבית לשרת ה-ROOT, התוקף בעצם יכול להחזיר תשובה במקום שרת ה-ROOT DNS, ובגלל שבקשות DNS לא נבדקות, אז מקור התשובה לא נבדק והאתר שאליו מפנה התוקף נשמר בטבלאות של ה-Local DNS וזו בעצם התשובה המוחזרת ללקוח – אתר פיקטיבי זדוני.

השימוש באתר נפוץ – איך יכול להיות שה-DNS המקומי לא מכיר אותו?

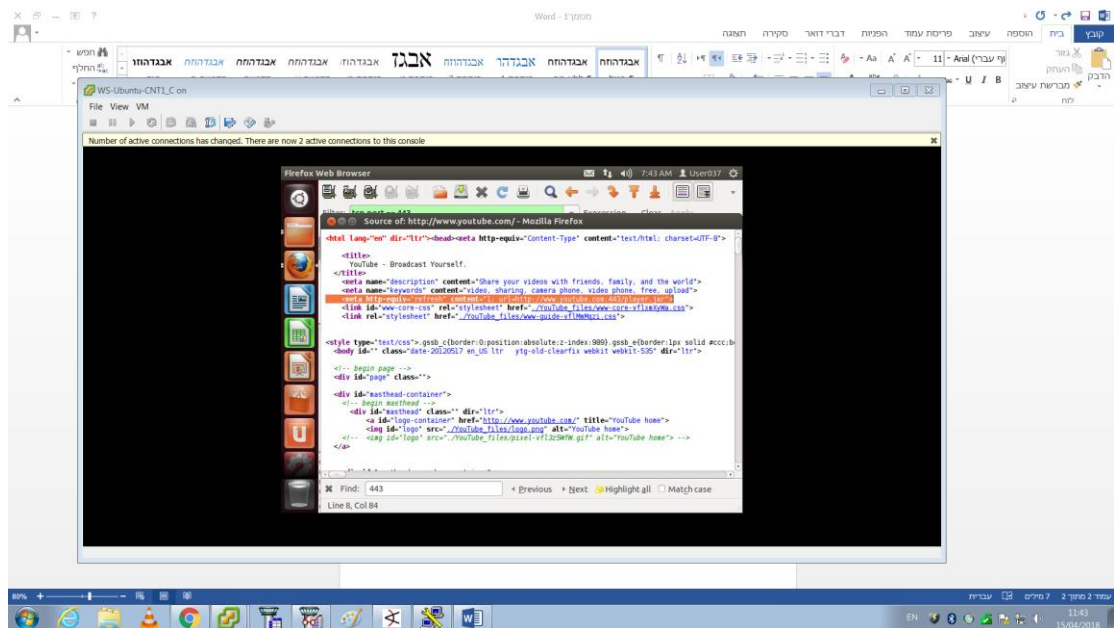
- 1 – שרתי ה-DNS מבצעים שאילתות כל כמה זמן לעדכון.
- 2 – אם ישנו Cache חשוב לדעת שגם להם יש תאריך תפוגה.

מלכתחילה היה קצת קשה לזהות את ה-DNS Poisoning דרך ה-DC, משום שהתקיפה לא באמת בוצעה בזמן אמת אלא הקבצים מראש היו שם למען דימוי התקיפה בתרחיש. אם נעשה nslookup מתוך הארגון ל-youtube.com נוכל לראות שהוא מבנה ל-IP של התוקף ולא ה-IP של youtube האמיתי. היתרון בהרעלת DNS הוא שה-URL נשאר בדיוק אותו דבר, ואז קשה יותר לעלות על זה. אך תקיפת הרעלת DNS פחות שכיחה היום.

מסקנות –

- העובד נכנס לאתר youtube מזויף שבו מושתלת הפניה לאתר הזדוני – 443 redirect והוא מוריד קובץ jar, קובץ הפעלה של Java.
- לא קפצה התראה ב-ArcSight מכיוון שהתוקף ניצל ידע על כך שיש שרת DNS בתוך הארגון ולכן לא היה צורך לבצע Port Scanning וכי' על מנת לגלות פרטים על השרת. אז לא קפצה התראה מאחר והכל היה נראה לגיטימי, פניה החוצה מלקוח – פניה לגיטימית.
- קבלת תשובה לגיטימית משרת ה-DNS המקורי, והורדת קובץ מאתר מוכר. במקרה הזה פשוט קיבלנו בקשה לקובץ שנראה לגיטימי למשתמש על המחשב ולכן הוא ביצע את ההורדה.

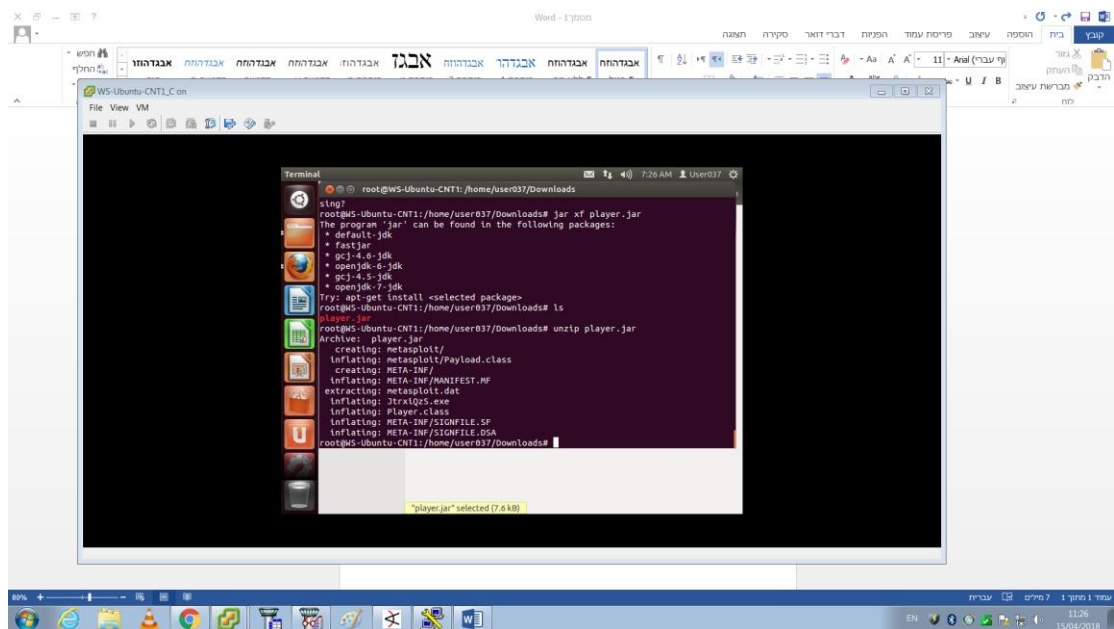
קוד ה-HTML של האתר אליו הגענו – (הרחבה על Java applet בהמשך)



ניתן לראות שבכל Refresh נתבקש להוריד את הקובץ הנ"ל.

לאחר שבדקנו את מקור הקובץ, כעת נרצה לראות שאכן הוא זדוני ובנוסף להבין מה פעולותיו.

בעזרת חילוץ קובץ ה-jar ניתן לראות את הקבצים המוכללים בו –



ניתן לראות כקובץ חשוד את **Metasploit** – פרויקט מטהספלויט הוא כלי המיועד למבדקי חדירה, מכיל בתוכו מאגר נתונים ופגיעויות נגד מערכות הפעלה, מערכות אנטי וירוס ותוכנות שונות.

ניצול נכון של מידע זה עשוי להוות דלת אחורית להתחברות אל מחשב היעד.

הפרויקט מבוסס קוד פתוח ומקבל תרומות והצעות עריכה מקהילת המשתמשים באמצעות אתר GitHub, ההצעות נבדקות על ידי צוות המורכב מעובדי Rapid7 וחברי קהילה בכירים.

מקור - <https://he.wikipedia.org/wiki/Metasploit>

בנוסף, המערכת מחזיקה אוסף של Exploits מוכרים ומוכנים לשימוש.

בעיקרון, הרעיון של מערכת זו הוא שימוש לטובה – כלומר למטרות בדיקה שרשת מוגנת ולראות אילו Exploits מצליחים לתקוף את המחשבים ברשת כלשהי, אך כמובן מנוצלים גם לרעה ע"י תוקפים.

Java Applet – תוכנית שהקריאה להפעלתה נעשית מתוך מסמך HTML אשר מוצג באמצעות דפדפן.

ה- Applet – יישומון, מהווה, בדרך כלל, תכנית קטנה שבולטת בתכונותיה הגרפיות.

איננו תכנית עצמאית וכדי להפעילו חייבים לשלבו במסמך HTML.

עצם הצפייה בדף ה- HTML באמצעות הדפדפן גורמת להפעלת ה- Applet.

מכאן ניתן להסיק איך הקובץ jar פועל ואיך הוא מתקשר לכניסה לאתר Youtube הזדוני.

בנוסף, ניתן לבצע reverse engineer לקבצי Java (ישנם אתרים שעושים זאת) וכך לפענח מה קבצים ההרצה עושים בעזרת הסתכלות על קוד המקור.

לסיכום, הקובץ Jar זדוני והוא הגורם לכך שהשרתים החיוניים בארגון נופלים אחד אחרי השני.

תהליך הגנה :

בתהליך ההגנה, הבנו שאכן עמדת העובד היא העמדה המותקפת באמצעות זיהוי של הורדת הקובץ הזדוני וחקירתו.

לכן, נצטרך להחזיר את כתובת youtube לכתובת הרגילה והמוכרת ובנוסף לבצע הסרה של הקבצים הזדוניים בעמדה.

תהליך הגנה מונעת :

בתהליך זה יש כמה דברים שכדאי לעשות בארגון כדי למנוע תקיפה כזו –

- 1 – תדריך העובדים בארגון – לעשות מדי פעם תרחישי תקיפה ולראות איך יגיבו העובדים ומשם להסיק מסקנות שיעזרו בהמשך. (כמובן, אזהרות מפני מתקפות מסוג זה).
- 2 – אנטי וירוס בעמדות העובדים שיבדוק את הקבצים שהורדו למחשב (לדוגמא, Intezer – מערכת המאפשרת נראות של כל התוכנות והקבצים הפועלים בארגון. בנוסף, יודעת לזהות באמצעות בדיקת DNA של הקובץ האם הוא קובץ זדוני או קובץ תקין. ניתן לקרוא עוד על Intezer כאן - <https://www.intezer.com/>)
- 3 – הגבלת הורדת קבצים בעמדות העובדים.
- 4 – כתיבת חוק שעמדת עובד פונה ב- SSH ל- Zenoss לדוגמא, ה- ArcSight יודיע על כך. (כמובן, גם בתקשורת עם שרתים שונים, כגון, Mail, Sql וכדומה. (בנוסף, חסימת עמדה במידה וכן הצליחה לפנות לשרת חיוני בארגון)
- 5 – בדיקת הפניה נכונה לאתר – מה שניתן לעשות על מנת לוודא שאכן אנו מופנים לאתר המקורי הוא ביצוע 2 בקשות (קרובות במיקום הפיזי על מנת שנקבל את אותו אייפי לאתר – משתנה בדרך כלל לפי מיקום גאוגרפי) והשוואה בין התוצאות של ההפניות. משווים שני מקורות כי רוב הסיכויים שלא שני המקומות הותקפו ולכן זה יכול לעזור מאוד לגלות את זה (לדוגמא, בקשה מ- google ובקשה מהרשת שלנו).

הפרצות באבטחת הארגון

ראה סעיף "תהליך הגנה מונעת".

ובנוסף, הארגון אינו משתמש בשירות ה-DNS החדש והחינמי 1.1.1.1 אשר לא שומרת את כתובת האיפי שלך ומאפשרת איסוף מידע.

איך להשתמש ב-DNS החדש? בקישור הבא -

[/https://www.geektime.co.il/cloudflares-new-dns](https://www.geektime.co.il/cloudflares-new-dns)

כלים שפיתחנו

אין ברשותנו כרגע את הידע לפתח כלים.

אופן עבודת הצוות

בתרחיש זה, מכיוון שהיו הרבה נפילות של שרתים בארגון, חילקנו את בדיקת השרתים הללו על ידי כך שבדקנו במקביל את קבצי הלוגים של השרתים הנופלים לפני שהגענו למסקנה שהגורם המרכזי הוא עמדת העובד וכך חסכנו בזמן יקר.

חוסרים/קשיים

קושי אחד שהיה הוא שלא הייתה אפשרות לצפות לכך שהתוקף שינה את כתובת האתר youtube לאתר זדוני.