דו״ח מעבדה- תרחיש מס׳ _01_

פרטים :

מגיש : אוראל מרדכי

תאריך :

שם התרחיש : הפלת השרת דרך פרוטוקול SSH(22)

תהליך ההתקפה: תחילה, התוקף ניסה להתחבר לשרת של רשת FOXNEWS (172.16.100.21). לשרת עשו port scanning, ולאחר שעשו port scanning לשרת זיהו אצלנו אוזן קשובה. ברגע שמצאו אצלנו את האוזן הקשובה ניסו לעשות לו SSH, התקוף גילה שהפרוטוקול שפתוח הוא של SSH והתחיל לעשות brute-force attack של סיסמאות.

תהליך הזיהוי: מה שקרה בעצם שהסיסמא שלנו הייתה חלשה (P@ssw0rd), התוקף התחבר דרך SSH (משהו שבעייתי בפני עצמו), ואנחנו באמצעות ה- ArcSight ראינו את כתובת התוקף שמנסה לתקוף את השרת ובשלב זה חוקקנו חוק שהתוקף לא יוכל להיכנס חזרה. שמנו לב בנוסף, כי ה –Service של apache2 נופל כל הזמן והתקדמנו לשרת(כי הבנו שיש בעיה בשרת), התחברנו לשרת דרך putty ומשמה התחלנו תהליך הגנה.

תהליך הגנה:

ראשית , עברנו על הלוגים בשרת (על מנת לבדוק דברים יוצאי דופן),ובשלב מסויים עלינו על הפקודה "ps aux" שעזרה לנו למצוא את הקובץ שהתוקף השתיל בשרת כולל כתובת התוקף ושמות הקבצים שהושתלו כולל נתיב התיקייה.

שנית, הקבוצה השנייה השתמשה בפקודה crontab –l וראתה כי התוקף הזריק לשרת (ל- cronjobs) שתי פקודות קריטיות שגורמות ל- Service apache2 לעצור לפי תזמון מסויים.

#)באמצעות קובץ syslog שנמצא בתיקיית log היה ניתן לראות את הפקודות שהזריק את התוקף לשרת.

#)בנוסף בתיקיית log באמצעות הקובץ auth.log ושימוש ב- grep(לחיפוש מועיל של המידע הנחוץ ("199")), ניתן היה לראות כי התוקף היה 5 דקות מרגע כניסתו לשרת ועד להתנתקות שלו מהשרת.

תהליך הגנה מונעת:

1.חוקקנו חוק שהתוקף לא יוכל להיכנס שוב לשרת ולגשת אליו(היינו צריכים לסגור גם את נתיב היציאה שלו מהשרת אך לא עשינו זאת).

2.העברנו את הקבצים (shadow+passwd) לתיקייה אחרת בשם "tmp", ואז מתקיית "tmp" יצרנו תיקייה חדשה בשם "DO_NOT_TOUCH".

3.ראינו כי בתיקיית "bd" נמצאים הקבצים הנגועים (shadow+passwd).

4.נכנסו לראות מה התוקף עשה בפקודת shell שהוא הזריק לשרת, ואז היה ניתן לראות כי התוקף פתח תיקייה חדשה בתוך "tmp" שתקרא "bd" (mkdir) , לאחר מכן העתיק התוקף את 2 קבצי ה-py לתיקייה שיצר. לאחר העתקת הקבצים לתיקייה ביקש התוקף מהשרת לפתוח בפייתון את הקבצים שנמצאים בתיקיית "tmp" ולהפעיל אותם. מה שעזר לנו להבין מה הוא עשה במהלך הזמן שהיה בשרת ואת מה שביצע בדיוק.

5. לבסוף הוסבר לנו כי באמצעות הזזת הקובץ/מחיקת הקובץ (root) שרץ ב –cronjobs , היה ניתן להפסיק את פקודות ה -shell שהזריק התוקף לשרת ובכך לגרום ל - Service apache2 לעצור כל הזמן ולהחזיר אותה לפעולה.

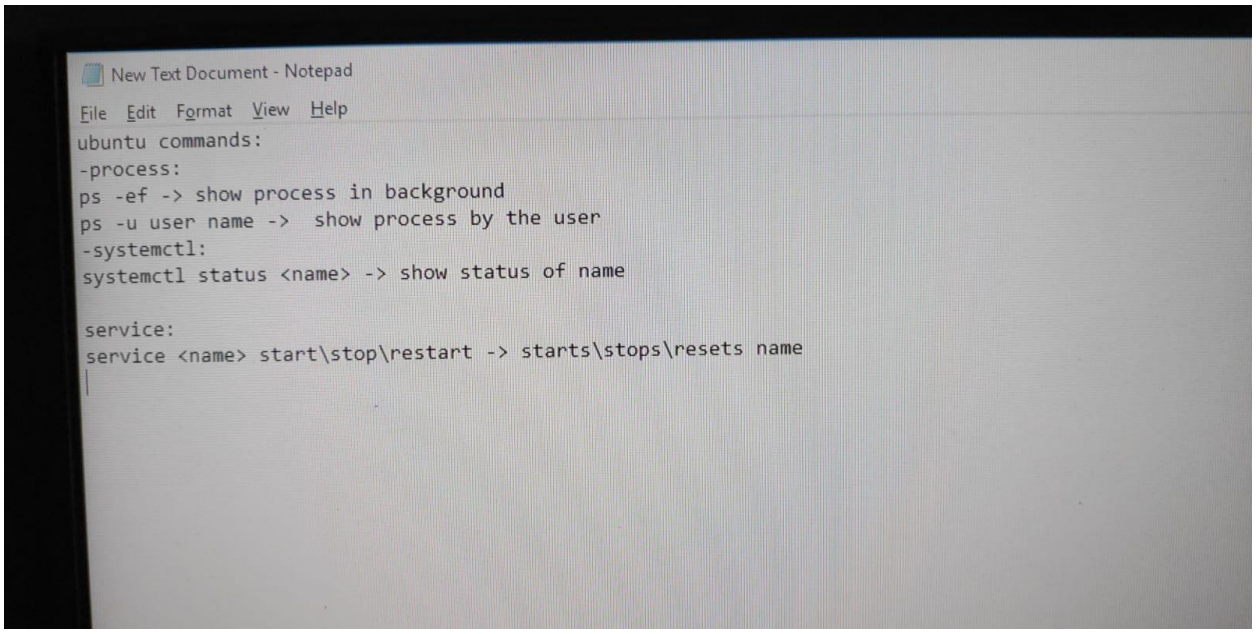
הפרצות באבטחת הארגון: סיסמא חלשה , שימוש בפרוטוקול ssh, ידע!!!

כלים שפיתחנו : אין

אופן עבודת הצוות: העבודה התחלקה בין כל אחד כמעט , אורם היה אחראי על חילוק התפקידים ושיתוף המידע מכל אחד. רע היה אחראי לחוקק חוקים. כל השאר היו אחראים לחפש בלוגים ולבדוק לגבי הקבצים שמפילים את השרת וזאת לאחר שחוקקנו חוק שהתוקף לא יכול להיכנס שוב(רק שלא חוקקנו חוק יציאה גם).

חוסרים/קשיים : שימוש בתוכנות , ידע בנושא, הכרת המערכת וסביבתה, פקודות בסיסיות .

תמונות שהעברנו בווטסאפ בזמן התרחיש :

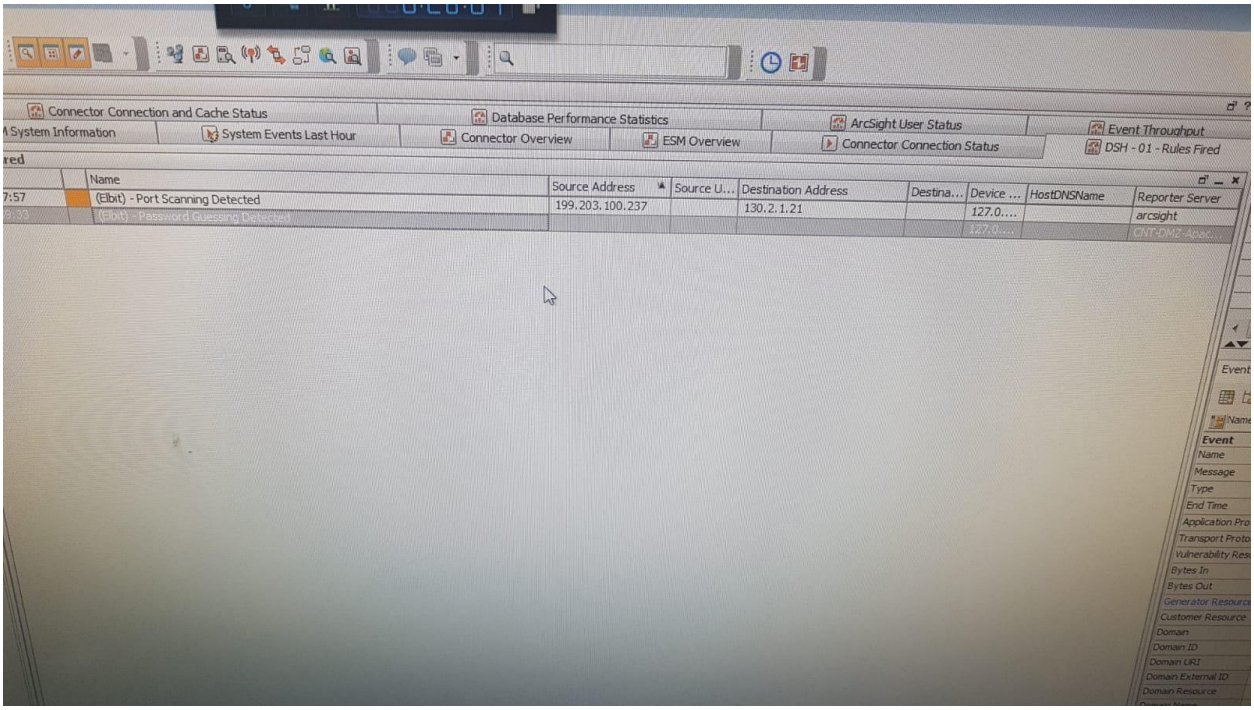


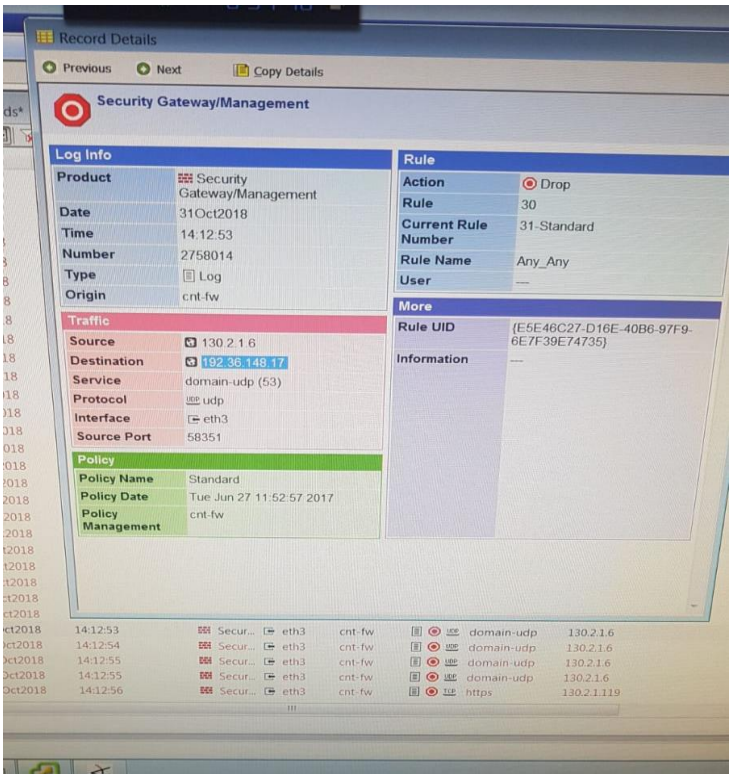
New Text Document - Notepad

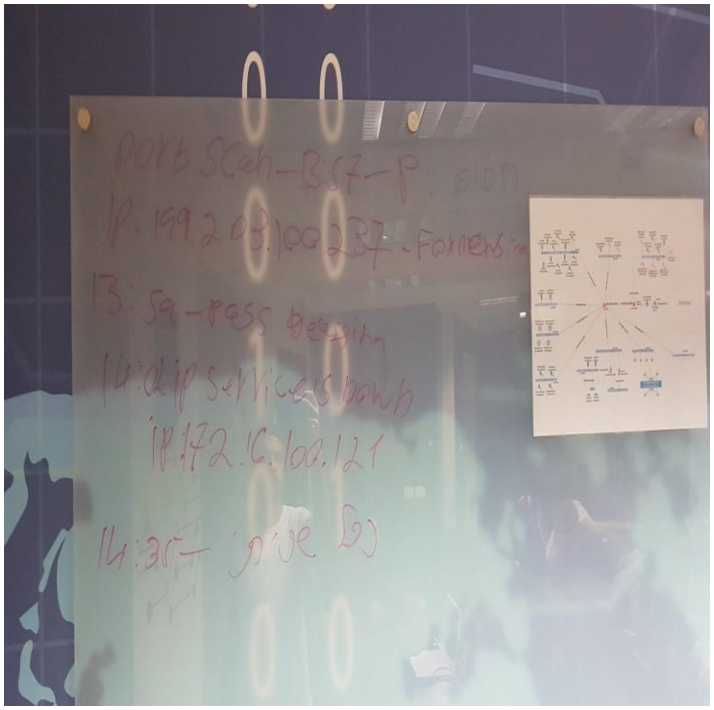
File Edit Format View Help

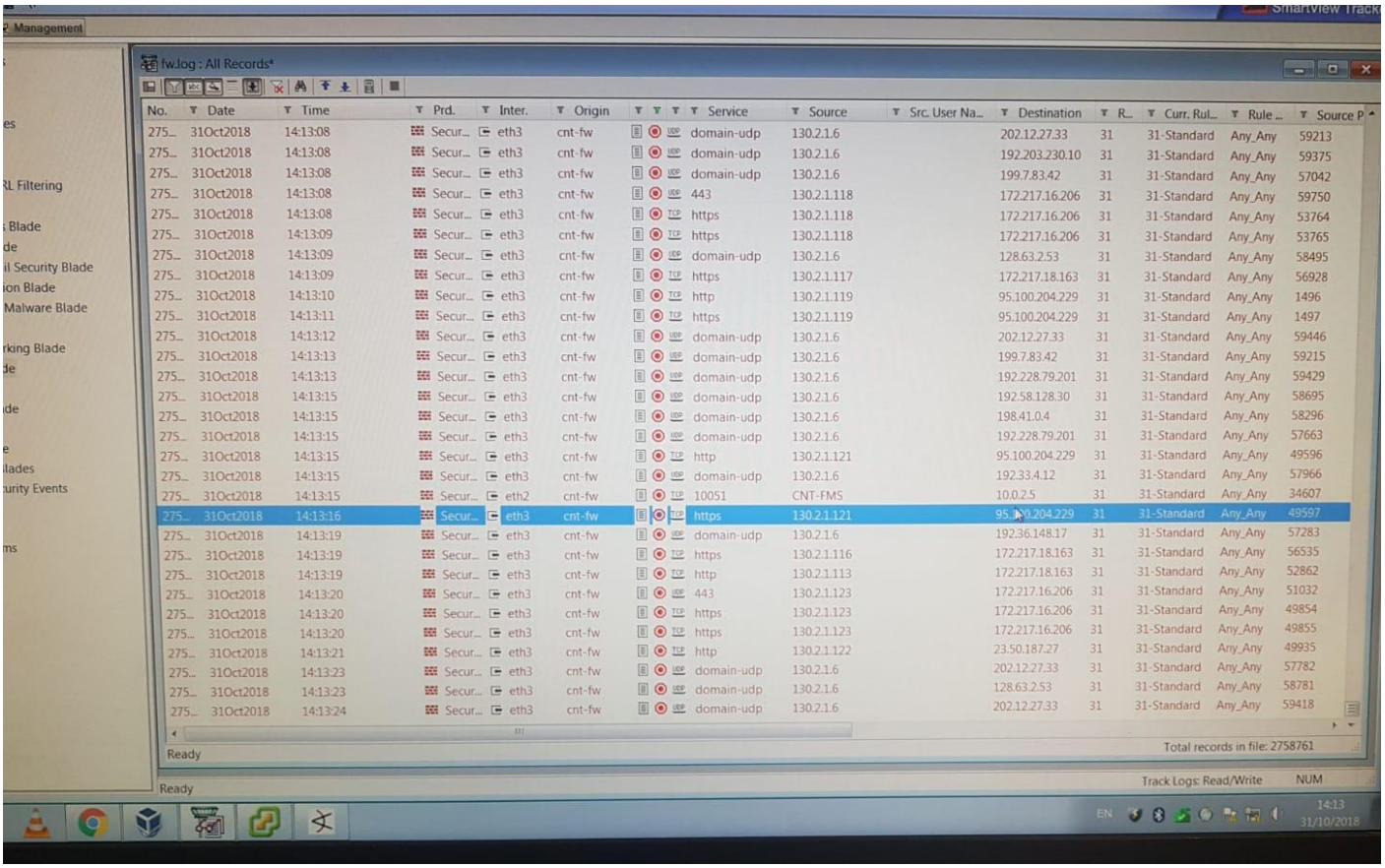
```
ubuntu commands:
-process:
ps -ef -> show process in background
ps -u user name ->  show process by the user
-systemctl:
systemctl status <name> -> show status of name

service:
service <name> start\stop\restart -> starts\stops\resets name
```




Record Details
Previous
Next
Copy Details
Security Gateway/Management
Log Info
Product Security Gateway/Management
Date 31Oct2018
Time 14:12:53
Number 2758014
Type Log
Origin cnt-fw
Rule
Action Drop
Rule 30
Current Rule Number 31-Standard
Rule Name Any_Any
User ---
Traffic
Source 130.2.1.6
Destination 192.36.148.17
Service domain-udp (53)
Protocol udp
Interface eth3
Source Port 58351
More
Rule UID {E5E46C27-D16E-40B6-97F9-6E7F39E74735}
Information ---
Policy
Policy Name Standard
Policy Date Tue Jun 27 11:52:57 2017
Policy Management cnt-fw




fw.log : All Records*
No. Date Time Prd. Inter. Origin Service Source Src. User Na... Destination R... Curr. Rul... Rule ... Source P
Total records in file: 2758761
Ready
Track Logs: Read/Write NUM