/Pre Security   
  
**//Network Fundamentals**

/// What is Networking?

**IP – Internet protocol & MAC**

כל קטע ב-IP נקרא Octet ויש 4 קטעים.  
הפירוש של .MAC – Media Access Control   
מספר סידורי לכל מכשיר, מורכב מ6 חלקים כאשר שלושת החלקים הראשונים מייצגים את החברה ושלושת החלקים האחרונים מייצגים את המכשיר עצמו ( החלק הייחודי).  
ההפרדה בין רשת פנימית לחיצונית ( מכשירים מקבלים IP פנימי וככה משוחחים ביניהם, לא יכול להיות שני מכשירים תחת אותה רשת עם IP זהים).  
כאשר מכשיר 'יוצא' לתקשורת עם מכשירים ברשת אחרת, אופן הזיהוי שלו הוא דרך הIP של הרשת ללא תלות בIP הפנימי שלו.  
כלומר, שני מכשירים יזוהו כאותו IP ברשת החיצונית.

(דוגמא spoofed – כאשר מכשיר מעמיד פנים ככתובת MAC אחרת על מנת לקבל שירות (להיכנס לוויפיי, וכו )

**Ping**

כלי נוסף, ping מיועד כדי לקבוע את הביצועים של חיבור בין התקנים.   
למשל, אם החיבור קיים או אמין.  
Ping uses ICMP (Internet Control Message Protocol)

**Gobuster**

כלי בשם Gobuster הוא כלי שורת פקודה בקוד פתוח המשמש לאילוץ קבצים של ספריות וקבצים בשרתי אינטרנט. זה נועד בעיקר לעזור לזהות נתיבים, ספריות וקבצים נסתרים בשרת אינטרנט על ידי שימוש ברשימת מילים ושליחת בקשות HTTP לנקודות קצה שונות.  
לדוגמא: gobuster -u http://fakebank.com -w wordlist.txt dir

/// Intro to LAN

**Local Area Network (LAN) Topologies**  
  
Bus, Star, Ring  
  
Switch

A Primer on Subnetting

Network Address

Host Address

Default Gateway

**ARP Protocol**

פרוטוקול ARP מאפשר למכשיר לשייך את כתובת ה-MAC שלו לכתובת IP ברשת. כל מכשיר ברשת ישמור יומן של כתובות ה-MAC המשויכות להתקנים אחרים.

**The DHCP Protocol**

ניתן להקצות כתובות IP באופן ידני, על ידי הזנתן פיזית למכשיר, או באופן אוטומטי והנפוץ ביותר על ידי שימוש בשרת DHCP (Dynamic Host Configuration Protocol).

Discover -> Offer -> Request -> Ack

**Firewall**

יש שני סוגים : stateful ו stateless .

Filter entire connection - Stateful

Filter firewall inspects individual packets - Stateless

Two common types attacks :

* Trojan Horse is a program that shows one desirable function but hides a malicious function underneath. For example, a victim might download a video player from a shady website that gives the attacker complete control over their system.
* Ransomware is a malicious program that encrypts the user’s files. Encryption makes the files unreadable without knowing the encryption password. The attacker offers the user the encryption password if the user is willing to pay a “ransom.”

**SIEM**

Security Information and Event Management system that is used to aggregate security information in the form of logs, alerts, artifacts and events into a centralized platform that would allow security analysts to perform near real-time analysis during security monitoring.’

/// Extending Your Network

**Ip scanner.thm**

כלי לסריקת IP וזיהוי כתובות חשודות

**Web Application Security - common attacks**

1. Identification and Authentication Failure
2. Broken Access Control (IDOOR -are a type of access control vulnerability that arises when an application uses user-supplied input to access objects directly).
3. Injection
4. Cryptographic Failures

**Scanning tools**

1. Nmap - We used nmap to learn about the running services – i.e. ftp etc’. ftp 10.10.15.241.
2. Document Metadata - pdfinfo DOCUMENT.pdf
3. Photo EXIF Data - exiftool IMAGE.jpg

**//Linux – common commands**ls | cd | cat | pwd | find ( find name a.txt / \*.txt) | grep  
ls -a -> All the files (also hidden) ;   
man \*\*command\*\* -> manual of this command  
su \*\*user\*\* / su -l \*\*user\*\* -> change account

& This operator allows you to run commands in the background of your terminal.

&& This operator allows you to combine multiple commands together in one line of your terminal.

> This operator is a redirector - meaning that we can take the output from a command (such as using cat to output a file) and direct it elsewhere.

>> This operator does the same function of the > operator but appends the output rather than replacing (meaning nothing is overwritten).



/etc

ספריית שורש זו היא אחת מספריות השורש החשובות ביותר במערכת שלך. התיקיה etc (קיצור של etcetera) היא מיקום נפוץ לאחסון קבצי מערכת המשמשים את מערכת ההפעלה שלך.

/var

ספריית "/var", כאשר "var" הוא קיצור של נתונים משתנים, היא אחת מתיקיות השורש העיקריות שנמצאות בהתקנת לינוקס. תיקיה זו מאחסנת נתונים שאליהם ניגשים או נכתבים לעתים קרובות על ידי שירותים או יישומים הפועלים על המערכת. לדוגמה, קבצי יומן משירותים ויישומים פועלים נכתבים כאן (/var/log), או נתונים אחרים שאינם בהכרח משויכים למשתמש ספציפי (כלומר, מסדי נתונים וכדומה).

/root

בניגוד לספריית /home, תיקיית /root היא למעשה הבית של משתמש מערכת ה"שורש". אין שום דבר נוסף בתיקייה הזו מלבד ההבנה שזוהי ספריית הבית של משתמש ה"שורש". אבל, כדאי להזכיר שכן ההנחה ההגיונית היא שמשתמש זה ימצא את הנתונים שלו בספרייה כגון "/home/root" כברירת מחדל.

/tmp

זוהי ספריית שורש ייחודית שנמצאת בהתקנת לינוקס. קיצור של "זמני", ספריית /tmp היא נדיפה ומשמשת לאחסון נתונים שיש צורך לגשת אליהם רק פעם או פעמיים. בדומה לזיכרון במחשב שלך, לאחר הפעלה מחדש של המחשב, התוכן של תיקיה זו נמחק.  
מה שמועיל לנו ב-pentesting הוא שכל משתמש יכול לכתוב לתיקיה זו כברירת מחדל. כלומר ברגע שיש לנו גישה למכונה, היא משמשת מקום טוב לאחסון דברים כמו תסריטי הספירה שלנו.

**SSH and how Does it Work**

Secure Shell or SSH simply is a protocol between devices in an encrypted form. Using cryptography, any input we send in a human-readable format is encrypted for travelling over a network -- where it is then unencrypted once it reaches the remote machine, such as in the diagram below.

Command: ssh username@ip

VIM and Nano – edit easy tools.

Downloading Files – wget

Transferring Files From Your Host - SCP (SSH)

Serving Files From Your Host – WEB ; HTTP Server

Process in Linux  
ps ; ps aux ; ps aux | less  
crontab – automation programs  
asp – package mennagment

ADS  
זרם נתונים ראשי: כאשר אתה יוצר קובץ ב-Windows, יש לו זרם נתונים ראשי המכיל את תוכן הקובץ. אלה הנתונים שבדרך כלל אתה מתקשר איתם בעת פתיחה ועריכה של קבצים.

זרמי נתונים חלופיים: NTFS מאפשר לקבצים להיות זרמי נתונים נוספים, המכונה זרמי נתונים חלופיים. זרמים אלו יכולים להכיל מידע נוסף שאינו חלק מתוכן הקובץ הראשי. לעתים קרובות נעשה שימוש ב-ADS למטרות לגיטימיות, כגון אחסון מטא נתונים או מידע נוסף הקשור לקובץ.

Framework for pentest:

OSSTMM:   
 a. Telecommunications (phones, VoIP, etc.)

b. Wired Networks

c. Wireless communications

OWASP ZAP 2.12.0

1. focuses on the testing of web applications.

NIST Cybersecurity Framework 1.1:

1. organizations cybersecurity standards.

NCSC CAF :

1. organizations cybersecurity standards.

Black box, White box, Grey box Penetration Testing – how much the tester is given any information about the inner workings of the application or service.

/ Jr Penetration Tester

// Introduction to Web Hacking tryhackme

/// Pre + Content discovery

לבדוק source   
א. בדיקה האם יש שימוש בfarmeים (קישורים חיצוניים של אתרים) שאינם מעודכנים בגרסה האחרונה באתר הנבדק.

ב. זיהוי list / directories ע"י הpath של קבצי html / CSS and etc.’ ובדיקה עבור קבצים נוספים.

ג. לבדוק Inspector - לא תמיד נראה את כל האלמנטים בעין או במקור ולא נרצה להיעזר ב source .  
כאשר יש תצוגה חסומה ננסה לבדוק את העיצוב CSS למשל ולשנות את הערך מ block ל none .

ד. להשתמש בdebugger - מאפשר לנו לעשות דיבג אין לתוכנית ולחקור פריטים או דברים שנעלמים במהלך טעינת העמוד.  
ה. להשתמש ב Network כדי לראות תנועות בדף כמו שליחת וקבלת בקשות ( Get , Post למשל)

robots.tx

קובץ robots.txt הוא מסמך שאומר למנועי חיפוש אילו דפים הם ואסור להופיע בתוצאות מנועי החיפוש שלהם או לאסור על מנועי חיפוש ספציפיים לסרוק את האתר לחלוטין. זה יכול להיות נוהג נפוץ להגביל אזורי אתרים מסוימים כך שהם לא יוצגו בתוצאות של מנוע החיפוש. דפים אלו עשויים להיות תחומים כמו פורטלי ניהול או קבצים המיועדים ללקוחות האתר. הקובץ הזה נותן לנו רשימה מצוינת של מיקומים באתר שהבעלים לא רוצים שנגלה כבודקי חדירה.

Manual Discovery – Favicon   
לפעמים כשמשתמשים ב-frameworks לבניית אתר, יישאר Favicon שהוא חלק מההתקנה, ואם מפתח האתר לא מחליף את זה באחד מותאם אישית, זה יכול לתת לנו רמז באיזו מסגרת נמצאת בשימוש

Sitemap.xml

בניגוד לקובץ robots.txt, המגביל את מה שסורקי מנועי החיפוש יכולים להסתכל עליו, קובץ sitemap.xml נותן רשימה של כל קובץ שבעל האתר מעוניין שיופיע במנוע חיפוש. אלה יכולים לפעמים להכיל אזורים באתר שקצת יותר קשים לניווט אליהם או אפילו רשימה של כמה דפי אינטרנט ישנים שהאתר הנוכחי כבר לא משתמש בהם אבל עדיין עובדים מאחורי הקלעים.

Header HTTP

כאשר אנו שולחים בקשות לשרת האינטרנט, השרת מחזיר כותרות HTTP שונות. כותרות אלו יכולות להכיל לפעמים מידע שימושי כגון תוכנת שרת האינטרנט ואולי שפת התכנות/סקריפטים הנמצאת בשימוש. בדוגמה למטה, אנו יכולים לראות ששרת האינטרנט הוא NGINX גרסה 1.18.0 ומריץ PHP גרסה 7.4.3. באמצעות מידע זה, נוכל למצוא גרסאות פגיעות של תוכנות בשימוש. נסה להריץ את פקודת ה-Curl להלן כנגד שרת האינטרנט, כאשר המתג -v מאפשר מצב מילולי, אשר יוציא את הכותרות (יכול להיות שיש משהו מעניין!). curl ip\_web -v  
  
Framework Stack

לאחר שהקמת את המסגרת של אתר אינטרנט, בין אם מהדוגמה של ה-favicon לעיל או על ידי חיפוש רמזים במקור הדף כגון הערות, הודעות זכויות יוצרים או קרדיטים, תוכל לאתר את אתר האינטרנט של המסגרת. משם, נוכל ללמוד עוד על התוכנה ומידע נוסף, ואולי יוביל לתוכן נוסף שנוכל לגלות.

Google Hacking / Dorking

Google Hacking / Dorking משתמש בתכונות המתקדמות של מנוע החיפוש של גוגל, המאפשרות לך לבחור תוכן מותאם אישית. אתה יכול, למשל, לבחור תוצאות מתוך שם דומיין מסוים באמצעות ה-site: filter, למשל (site:tryhackme.com) ואז תוכל להתאים את זה למונחי חיפוש מסוימים, למשל, המילה admin (אתר :tryhackme.com admin) אז זה יחזיר רק תוצאות מאתר tryhackme.com המכילות את המילה admin בתוכן שלו. ניתן לשלב גם מסננים מרובים. להלן דוגמה למסננים נוספים שתוכלו להשתמש בהם.

Wappalyzer

Wappalyzer (https://www.wappalyzer.com/) הוא כלי מקוון ותוסף דפדפן המסייע לזהות באילו טכנולוגיות משתמש אתר אינטרנט, כגון מסגרות, מערכות ניהול תוכן (CMS), מעבדי תשלומים ועוד, והוא יכול אפילו מצא גם מספרי גרסאות.

Wayback Machine

The Wayback Machine (https://archive.org/web/) הוא ארכיון היסטורי של אתרי אינטרנט שתחילתו בשנות ה-90 המאוחרות. אתה יכול לחפש שם דומיין, והוא יראה לך את כל הפעמים שהשירות גירד את דף האינטרנט ושמר את התוכן. שירות זה יכול לסייע בגילוי דפים ישנים שעשויים להיות עדיין פעילים באתר הנוכחי.

S3 Buckets

S3 Buckets הוא שירות אחסון שמסופק על ידי Amazon AWS, המאפשר לאנשים לשמור קבצים ואפילו תוכן סטטי של אתר אינטרנט בענן הנגיש באמצעות HTTP ו-HTTPS. הבעלים של הקבצים יכול להגדיר הרשאות גישה כדי להפוך קבצים לציבוריים, פרטיים ואפילו ניתנים לכתיבה. לפעמים הרשאות גישה אלו מוגדרות בצורה שגויה ומאפשרות בטעות גישה לקבצים שלא אמורים להיות זמינים לציבור. הפורמט של דלי S3 הוא http(s)://{name}.s3.amazonaws.com כאשר {name} נקבע על ידי הבעלים, כגון tryhackme-assets.s3.amazonaws.com. ניתן לגלות דלי S3 בדרכים רבות, כגון מציאת כתובות האתרים במקור העמוד של האתר, מאגרי GitHub, או אפילו אוטומציה של התהליך. שיטת אוטומציה נפוצה אחת היא שימוש בשם החברה ואחריו מונחים נפוצים כגון {name}-assets, {name}-www, {name}-public, {name}-private וכו'.

גילוי אוטומטי   
תהליך של שימוש בכלים לגילוי תוכן במקום לעשות זאת באופן ידני. תהליך זה הוא אוטומטי מכיוון שהוא מכיל בדרך כלל מאות, אלפי או אפילו מיליוני בקשות לשרת אינטרנט. בקשות אלו בודקות אם קיים קובץ או ספרייה באתר אינטרנט, מה שנותן לנו גישה למשאבים שלא ידענו על קיומם קודם לכן. תהליך זה מתאפשר באמצעות משאב הנקרא רשימות מילים.

ffuf

dirb - כתובת URL של dirb LINK/common.txt

Gobuster - gobuster dir --url URL -w LINK/common.txt

/// Subdomain Enumeration

ספירת תת-דומיינים היא התהליך של מציאת תת-דומיינים תקפים לדומיין, אך מדוע אנו עושים זאת? אנו עושים זאת כדי להרחיב את משטח ההתקפה שלנו כדי לנסות ולגלות נקודות תורפה פוטנציאליות נוספות.

נחקור שלוש שיטות ספירת תת-דומיינים שונות: Brute Force, OSINT (Intelligence בקוד פתוח) ומארח וירטואלי.  
  
\* המטרה של יומני שקיפות אישורים היא למנוע שימוש באישורים זדוניים שנעשו בטעות. נוכל להשתמש בשירות זה לטובתנו כדי לגלות תת-דומיינים השייכים לדומיין - crt.sh  
\* מנועי חיפוש

מנועי חיפוש מכילים טריליוני קישורים ליותר ממיליארד אתרים, שיכולים להיות משאב מצוין למציאת תת-דומיינים חדשים. שימוש בשיטות חיפוש מתקדמות באתרים כמו גוגל, כמו ה-site: filter, יכול לצמצם את תוצאות החיפוש. לדוגמה, "-site:www.domain.com site:\*.domain.com" יכיל רק תוצאות המובילות לשם התחום domain.com אך לא יכלול קישורים אל www.domain.com; לכן, הוא מציג לנו רק שמות תת-דומיין השייכים ל-domain.com.  
  
\* DNS Bruteforce  
\* OSINT - Sublist3r  
\* Virtual Hosts – using ffuf

/// Authentication Bypass

Username Enumeration

1. תרגיל מועיל להשלמות הוא כאשר מנסים למצוא פרצות אימות הוא יצירת רשימה של שמות משתמש חוקיים, שבהם נשתמש בהמשך במשימות אחרות. (נשתמש בfuff)  
   נוכל להעזר באמצעי זה רק כאשר יש לנו פידבק על קיימות המשתמש ( “The account already exist” )

Brute Force

1. מתקפת כוח גס היא תהליך אוטומטי שמנסה רשימה של סיסמאות נפוצות נגד שם משתמש בודד או, כמו במקרה שלנו, רשימה של שמות משתמש

Logic Flaws

1. לפעמים תהליכי אימות מכילים פגמים לוגיים. פגם לוגי הוא כאשר הנתיב הלוגי הטיפוסי של יישום עוקף, עוקף או מטופל על ידי האקר. פגמים לוגיים יכולים להתקיים בכל אזור באתר, אבל אנחנו הולכים להתרכז בדוגמאות הקשורות לאימות במקרה זה.  
   למשל – עבור בקשת מייל ואז וידוא שם משתמש ושינוי הבקשות Post ו Get .

Cookie Tampering

1. בדיקה ועריכה של קובצי ה-cookie שהוגדרו על ידי שרת האינטרנט במהלך ההפעלה המקוונת שלך עשויות להביא לתוצאות מרובות, כגון גישה לא מאומתת, גישה לחשבון של משתמש אחר או הרשאות מוגברות. אם אתה צריך רענון על קובצי Cookie, בדוק את חדר HTTP In Detail במשימה 6.

/// IDOR

שינוי פרמטרים כמו user\_id וכו'. המיקום שלהם יכול להמצא ב -  
א. בשורת url  
ב. כערך מקודד ואז I) נבצע encode נשנה את ערכו ונקודד בחזרה או

II) נשתמש ב- <https://crackstation.net/> לחיפוש ערכים נפוצים

ג. יצירת 2 חשבונות והחלפת הערכים ביניהם – מצביע על פרצה

ד. שינוי ערכים ב- browser developer tools .

ה. התקפה המכונה כריית פרמטרים ( parameters mining)

///File Inclusion

ניצול פגיעויות של הכללת קבצים מקומיים (LFI), הכללת קבצים מרחוק (RFI) ומעבר ספריות.

﻿Path Traversal

הידועה גם כ-Directory Traversal, פגיעות אבטחת אינטרנט מאפשרת לתוקף לקרוא משאבי מערכת הפעלה, כגון קבצים מקומיים בשרת המריץ יישום. התוקף מנצל פגיעות זו על ידי מניפולציה וניצול לרעה של כתובת האתר של יישום האינטרנט כדי לאתר ולגשת לקבצים או ספריות המאוחסנות מחוץ לספריית הבסיס של היישום.

דוגמא - <http://webapp.thm/get.php?file=../../../../etc/passwd>

עוד קבצים נפוצים נמצא ב - <https://tryhackme.com/room/fileinc>

Local File Inclusion (﻿LFI)

זיהוי מספר התיקיות שנדרש ללכת בחזרה לroot . כלומר ../  
ביטול סיומת ע"י %00  
ביטול filterים ע"י שימוש בלכסון כפול.  
חזרה ל root כאשר התיקייה היא חלק מהארגומנטים.

Remote File Inclusion – RFI

Remote File Inclusion (RFI) היא טכניקה לכלול קבצים מרוחקים לתוך יישום פגיע.

הרמת שרת דרך הcmd ( python3 -m http.serer) ליצור קובץ עם הקוד המבוקש (למשל kill.txt) ולשלוח את זה דרך הפרמטר Get כ ?file=http://MY\_IP:8000/kill.txt

כלים שעבדנו איתם:  
א. שימוש ב- brupsuite לטובת שינוי בקשה מGet לPost  
ב. שינוי ערך ב- Cookies לטובת גישה כאיש אחר.

ג. זיהוי שגיאות והשמה של %00 במידת הצורך

///SSRF

SSRF ראשי תיבות של Server-Side Request Forgery. זוהי פגיעות המאפשרת למשתמש זדוני לגרום לשרת האינטרנט לבצע בקשת HTTP נוספת או ערוכה למשאב לפי בחירת התוקף.

כלומר, לשנות את כתובת הlink המבוקשת לכתובת זדונית. הדבר יכול להתבטא בכתובת url או בכתובת של תמונה למשל שנמצא בsource בכלל.  
שינויים אלו, חזרה לתיקייה והגעה לנתיב – מפורט בקישור של הדף.

/// Cross-site Scripting

Reflected XSS - מתרחש כאשר נתונים שסופקו על ידי המשתמש בבקשת HTTP נכללים במקור דף האינטרנט ללא כל אימות.

Stored XSS - כפי שהשם מסיק, מאוחסן באפליקציית האינטרנט (בבסיס נתונים, למשל) ואז מופעל כאשר משתמשים אחרים מבקרים באתר או בדף האינטרנט.

DOM Based XSS - הוא המקום שבו ביצוע JavaScript מתרחש ישירות בדפדפן מבלי שדפים חדשים נטענים או נתונים נשלחים לקוד backend. ביצוע מתרחש כאשר קוד JavaScript של האתר פועל על קלט או אינטראקציה של המשתמש. (למשל windows.location.hash )

Blind XSS - Blind XSS דומה ל-XSS מאוחסן (שאותו כיסינו במשימה 4) בכך שהמטען שלך מאוחסן באתר כדי שמשתמש אחר יוכל לצפות בו, אך במקרה זה, אינך יכול לראות את המטען פועל או להיות מסוגל לבדוק אותו נגד עצמך קודם כל.

-> כלי לבדיקת XSS Hunter Express .  
-> פוליגלוט XSS הוא מחרוזת טקסט שיכולה להימלט מתכונות, תגים ומסננים לעקוף הכל באחד. ( לדוגמא jaVasCript:/\*-/\*`/\*\`/\*'/\*"/\*\*/(/\* \*/onerror=alert('THM') )//%0D%0A%0d%0a//</stYle/</titLe/</teXtarEa/</scRipt/--!>\x3csVg/<sVg/oNloAd=alert('THM')//>\x3e )

אחרי זיהוי xss ניתן ליצור שרת המאזין לבקשה ולקחת את קבצי הcookies -  Practical Example (Blind XSS) בדף זה.

///Command Injection

הזרקת פקודה היא שימוש לרעה בהתנהגות של אפליקציה לביצוע פקודות במערכת ההפעלה, תוך שימוש באותן הרשאות שהאפליקציה במכשיר פועלת איתם. לדוגמה, השגת הזרקת פקודות על שרת אינטרנט הפועל כמשתמש בשם joe תבצע פקודות תחת משתמש ג'ו זה - ולכן תקבל את כל ההרשאות שיש לג'ו.  
ע"י ping, pwd, ls , ; , וסוגרים אלו ואחרים.

///SQL Injection

(Structured Query Language), המכונה בעיקר SQLi, היא התקפה על שרת מסד נתונים של יישומי אינטרנט הגורמת לביצוע שאילתות זדוניות. כאשר יישום אינטרנט מתקשר עם מסד נתונים באמצעות קלט ממשתמש שלא אומת כראוי, קיים פוטנציאל של תוקף שיכול לגנוב, למחוק או לשנות נתונים פרטיים ולקוחות וגם לתקוף את שיטות האימות של יישומי האינטרנט לפרטי או אזורי לקוחות. זו הסיבה שבגלל ש- SQLi היא אחת מפגיעות יישומי האינטרנט הוותיקות ביותר, היא גם יכולה להזיק ביותר.

// Burp suite

///The Basic

Proxy: ההיבט הידוע ביותר של Burp Suite, ה-Burp Proxy מאפשר לנו ליירט ולשנות בקשות/תגובות בעת אינטראקציה עם יישומי אינטרנט.

Repeater: תכונת הגיהוק השנייה הידועה ביותר - Repeater - מאפשרת לנו ללכוד, לשנות ולאחר מכן לשלוח שוב את אותה בקשה מספר פעמים. תכונה זו יכולה להיות בעלת ערך רב, במיוחד כאשר אנו צריכים ליצור מטען באמצעות ניסוי וטעייה (למשל ב-SQLi -- Structured Query Language Injection) או בעת בדיקת הפונקציונליות של נקודת קצה עבור פגמים.

Intruder: למרות שהקצב מוגבל בקהילת Burp, Intruder מאפשר לנו לרסס נקודת קצה עם בקשות. זה משמש לעתים קרובות עבור התקפות bruteforce או כדי לטשטש נקודות קצה.

decoder: למרות שהוא פחות בשימוש מהתכונות שהוזכרו קודם לכן, מפענח עדיין מספק שירות רב ערך בעת שינוי נתונים - או במונחים של פענוח מידע שנלכד, או קידוד מטען לפני שליחתו למטרה. בעוד שיש שירותים אחרים זמינים לעשות את אותה העבודה, לעשות זאת ישירות בתוך Burp Suite יכול להיות יעיל מאוד.

Comparer: כפי שהשם מרמז, Comparer מאפשר לנו להשוות שני חלקי נתונים ברמת מילה או בתים. שוב, זה לא משהו שייחודי ל-Burp Suite, אבל היכולת לשלוח פיסות נתונים (פוטנציאליות גדולות מאוד) ישירות לתוך כלי השוואה עם קיצור מקלדת בודד יכולה להאיץ את העניינים בצורה ניכרת.

רצף: בדרך כלל אנו משתמשים ב-Sequencer כאשר אנו מעריכים את האקראיות של אסימונים כגון ערכי קובצי Cookie של הפעלה או נתונים אחרים שנוצרו כביכול אקראיים. אם האלגוריתם אינו מייצר ערכים אקראיים מאובטחים, הדבר עלול לפתוח כמה דרכים הרסניות להתקפה.

\*שימוש ב- FoxyProxy בFirefox לטובת ניתוב האינטרנט לשרת הproxy.

\* פתרון באתרים המתמשים ב- TLS

Site map :מאפשרת לנו למפות את האפליקציות אליהן אנו מכוונים במבנה עץ. כל עמוד שנבקר בו יופיע כאן, מה שיאפשר לנו ליצור אוטומטית מפת אתר עבור היעד פשוט על ידי גלישה ברחבי אפליקציית האינטרנט. Burp Pro גם יאפשר לנו לעכביש את המטרות באופן אוטומטי (כלומר לחפש קישורים בכל עמוד ולהשתמש בהם כדי למפות כמה שיותר מהאתר נגיש לציבור באמצעות הקישורים בין הדפים); עם זאת, עם Burp Community, אנו עדיין יכולים להשתמש בזה כדי לצבור נתונים בזמן שאנו מבצעים את שלבי הספירה הראשוניים שלנו.

מפת האתר יכולה להיות שימושית במיוחד אם ברצוננו למפות ממשק API, שכן בכל פעם שאנו מבקרים בדף, כל נקודת קצה של ממשק API שמהם הדף מאחזר נתונים בזמן הטעינה יופיעו כאן.

Scope Settings: כבר ראינו את חלון הגדרות ההיקף - הוא מאפשר לנו לשלוט בהיקף היעד של Burp עבור הפרויקט

///Intruder

Intruder הוא כלי ה-fuzzing המובנה של Burp Suite. זה מאפשר לנו לקחת בקשה (בדרך כלל נתפסת ב-Proxy לפני שהועברה ל-Intruder) ולהשתמש בה כתבנית כדי לשלוח בקשות רבות נוספות עם ערכים שהשתנו מעט באופן אוטומטי. לדוגמה, על ידי לכידת בקשה המכילה ניסיון התחברות, נוכל להגדיר את Intruder להחליף את שדות שם המשתמש והסיסמה עבור ערכים מרשימת מילים, ובכך לאפשר לנו למעשה להפעיל את טופס ההתחברות.

ישנם ארבעה סוגי תקיפה זמינים:

Sniper - ייקח כל מטען בערכת מטען ויכניס אותו לכל עמדה מוגדרת בתורו. (רשימה של מילים ובכל פעם ישנה ערך אחד מתוך רשימת המילים).

Battering ram - שם את אותו מטען בכל עמדה ולא בכל עמדה בתורו. (אם יש שלוש מילים, יחליף עבור ערך אחד את כולם באותה הרצה).

Pitchfork - לוקח את הפריט הראשון מכל רשימה ומכניס אותם לבקשה, אחד לכל עמדה. לאחר מכן הוא חוזר על זה עבור הבקשה הבאה.

Cluster bomb - אחת לכל עמדה, עד למקסימום של 20; עם זאת, בעוד Pitchfork חוזר על כל סט מטען בו זמנית, Cluster bomb חוזר על כל סט מטען בנפרד, ומוודא שכל שילוב אפשרי של מטענים נבדק. (את כל הצירופים האפשריים). לוקח לא מעט זמן.

* דפים המכילים CRFS tokens יצטרכו אימות נוסף בעת הגשת הטפסים ולכן באופן ספציפי, נצטרך להגדיר "מאקרו" (כלומר קבוצה קצרה של פעולות חוזרות) שיבוצעו לפני כל בקשה. זה יתפוס קובץ Cookie ייחודי של הפעלה ואסימון התחברות תואם, ואז יחליף אותם בכל בקשה של ההתקפה שלנו.  
    
  Decoder – משתמש בכל מיני קידודים (גם לפענוח וגם להצפנה). בנוסף יש smart encoder.  
  בנוסף, שימוש בhashing

ישנן שתי שיטות עיקריות בהן אנו יכולים להשתמש כדי לבצע ניתוח אסימון עם Sequencer:

לכידה חיה היא הנפוצה יותר מבין שתי השיטות -- זוהי כרטיסיית המשנה המוגדרת כברירת מחדל עבור Sequencer. לכידת חי מאפשרת לנו להעביר בקשה ל-Sequencer, שאנו יודעים שתיצור עבורנו אסימון לניתוח. לדוגמה, ייתכן שנרצה להעביר בקשת POST לנקודת קצה התחברות ל-Sequencer, מכיוון שאנו יודעים שהשרת יגיב במתן קובץ Cookie. לאחר שהבקשה תועבר פנימה, נוכל לומר ל-Sequencer להתחיל לכידה חיה: לאחר מכן הוא יבצע את אותה בקשה אלפי פעמים באופן אוטומטי, ויאחסן את דגימות האסימונים שנוצרו לניתוח. לאחר שצברנו מספיק דגימות, אנו עוצרים את Sequencer ומאפשרים לו לנתח את האסימונים שנלכדו.

טעינה ידנית מאפשרת לנו לטעון רשימה של דגימות אסימונים שנוצרו מראש ישירות לתוך Sequencer לצורך ניתוח. שימוש בטעינה ידנית אומר שאנחנו לא צריכים לשלוח אלפי בקשות ליעד שלנו (שזה גם רועש וגם הרבה משאבים), אבל זה אומר שאנחנו צריכים להשיג רשימה גדולה של אסימונים שנוצרו מראש!

// Network security

///Passive Reconnaissance

3 כלים עיקרים שהתעסקנו איתם:   
whois לבצע שאילתות על שרתי WHOIS - בלינוקס, הפקודה whois היא כלי עזר המשמש לאחזור מידע על שמות דומיינים, כתובות IP וישויות הקשורות לרשת מרשמי שמות דומיינים ומרישום אינטרנט אזוריים (RIR). הוא מספק פרטים על הבעלות, תאריך הרישום, פרטי הקשר ונתונים רלוונטיים אחרים עבור דומיין או כתובת IP נתונה.

nslookup כדי לבצע שאילתות על שרתי DNS - nslookup הוא כלי שורת פקודה רב תכליתי לשאילתת מידע DNS ואבחון בעיות הקשורות ל-DNS. זהו משאב רב ערך עבור מנהלי רשת, מנהלי מערכות וכל מי שצריך לעבוד עם תצורות DNS ופתרון שמות דומיין.

dig - לשאילתות DNS מתקדמות יותר ופונקציונליות נוספת, אתה יכול להשתמש ב-dig.  
  
כלים יותר פרקטים –  
\* DNSDumpster (אפשר להשתמש בשירות מקוון המציע תשובות מפורטות לשאילתות DNS, כמו מציאת sub domain).  
\* Shodan.io הוא מנוע חיפוש פופולרי באינטרנט וכלי אבטחת סייבר המשמש לגילוי וניתוח מכשירים ומערכות המחוברים לאינטרנט. בניגוד למנועי חיפוש מסורתיים כמו גוגל, שמאנדקסים תוכן אינטרנט, שודן מתמקדת באינדקס מידע על המכשירים והמערכות המחוברות לאינטרנט.

///Active Reconnaissance

יש גם המון תוספים עבור Firefox וכרום שיכולים לעזור בבדיקות חדירה. הנה כמה דוגמאות:

\*\* FoxyProxy מאפשר לך לשנות במהירות את שרת ה-proxy שבו אתה משתמש כדי לגשת לאתר היעד. תוסף דפדפן זה נוח כאשר אתה משתמש בכלי כגון Burp Suite או אם אתה צריך להחליף שרתי פרוקסי באופן קבוע. אתה יכול להשיג את FoxyProxy עבור Firefox מכאן.

\*\* User-Agent Switcher and Manager נותן לך את היכולת להעמיד פנים שאתה ניגש לדף האינטרנט ממערכת הפעלה אחרת או דפדפן אינטרנט אחר. במילים אחרות, אתה יכול להעמיד פנים שאתה גולש באתר באמצעות אייפון כאשר למעשה אתה ניגש אליו מ-Mozilla Firefox.

\*\* Wappalyzer מספק תובנות לגבי הטכנולוגיות המשמשות באתרים שבהם ביקרת. תוסף כזה שימושי, בעיקר כאשר אתה אוסף את כל המידע הזה בזמן הגלישה באתר כמו כל משתמש אחר. צילום מסך של Wappalyzer מוצג להלן. אתה יכול למצוא את Wappalyzer עבור Firefox כאן.

הצגנו את ping, traceroute, telnet ו-nc כדי לאסוף מידע על הרשת, המערכת והשירותים.  
ping - המטרה העיקרית של הפינג היא לבדוק האם אתה יכול להגיע למערכת המרוחקת ושהמערכת המרוחקת יכולה להגיע אליך בחזרה. במילים אחרות, בתחילה, זה שימש לבדיקת קישוריות רשת; עם זאת, אנו מתעניינים יותר בשימושים השונים שלה: לבדוק אם המערכת המרוחקת מקוונת.

במילים פשוטות, הפקודה ping שולחת מנה למערכת מרוחקת, והמערכת המרוחקת עונה. כך ניתן להסיק שהמערכת המרוחקת מקוונת ושהרשת פועלת בין שתי המערכות.  
traceroute - הפקודה traceroute עוקבת אחר המסלול שעברו החבילות מהמערכת שלך למארח אחר. המטרה של traceroute היא למצוא את כתובות ה-IP של הנתבים או ה-hips שחבילה עוברת כשהיא עוברת מהמערכת שלך למארח היעד. פקודה זו חושפת גם את מספר הנתבים בין שתי המערכות. זה מועיל מכיוון שהוא מציין את מספר הקפצות (נתבים) בין המערכת שלך למארח היעד. עם זאת, שים לב שהמסלול שלוקח החבילות עשוי להשתנות מכיוון שרבים מהנתבים משתמשים בפרוטוקולי ניתוב דינמיים המותאמים לשינויים ברשת.  
אין דרך ישירה לגלות את הדרך מהמערכת שלך למערכת יעד. אנו סומכים על ICMP כדי "להערים" על הנתבים לחשוף את כתובות ה-IP שלהם. אנו יכולים להשיג זאת על ידי שימוש ב-Time To Live (TTL) קטן בשדה כותרת ה-IP. למרות שה-T ב-TTL מייצג זמן, TTL מציין את המספר המקסימלי של נתבים/כפיצות שחבילה יכולה לעבור דרכם לפני שהיא נפלטה; TTL אינו מספר מקסימלי של יחידות זמן. כאשר נתב מקבל חבילה, הוא מוריד את ה-TTL באחד לפני שהוא מעביר אותו לנתב הבא. האיור הבא מראה שבכל פעם שחבילת ה-IP עוברת דרך נתב, ערך ה-TTL שלה מופחת ב-1. בתחילה, היא משאירה את המערכת עם ערך TTL של 64; הוא מגיע למערכת היעד עם ערך TTL של 60 לאחר שעבר דרך 4 נתבים.  
Telent - פרוטוקול TELNET (Teletype Network) פותח בשנת 1969 כדי לתקשר עם מערכת מרוחקת באמצעות ממשק שורת פקודה (CLI). לפיכך, הפקודה telnet משתמשת בפרוטוקול TELNET לניהול מרחוק. יציאת ברירת המחדל שבה משתמש telnet היא 23. מנקודת מבט אבטחה, telnet שולחת את כל הנתונים, כולל שמות משתמש וסיסמאות, בטקסט ברור. שליחת טקסט ברור מקלה על כל מי שיש לו גישה לערוץ התקשורת לגנוב את אישורי הכניסה. החלופה המאובטחת היא פרוטוקול SSH (Secure SHell).

עם זאת, לקוח telnet, עם הפשטות שלו, יכול לשמש למטרות אחרות. בידיעה שלקוח telnet מסתמך על פרוטוקול TCP, אתה יכול להשתמש ב-Telnet כדי להתחבר לכל שירות ולתפוס את הבאנר שלו. באמצעות telnet 10.10.100.137 PORT, אתה יכול להתחבר לכל שירות הפועל ב-TCP ואפילו להחליף כמה הודעות אלא אם כן הוא משתמש בהצפנה.

נניח שאנו רוצים לגלות מידע נוסף על שרת אינטרנט, מאזין ביציאה 80. אנו מתחברים לשרת ביציאה 80, ולאחר מכן אנו מתקשרים באמצעות פרוטוקול HTTP. אתה לא צריך לצלול לתוך פרוטוקול HTTP; אתה רק צריך להנפיק GET / HTTP/1.1. כדי לציין משהו אחר מלבד דף האינדקס המוגדר כברירת מחדל, אתה יכול להנפיק GET /page.html HTTP/1.1, אשר יבקש את page.html. כמו כן, ציינו לשרת האינטרנט המרוחק שאנו רוצים להשתמש ב-HTTP גרסה 1.1 לתקשורת. כדי לקבל תגובה חוקית, במקום שגיאה, עליך להזין ערך כלשהו עבור המארח המארח: דוגמה וללחוץ על enter פעמיים. ביצוע שלבים אלה יספק את דף האינדקס המבוקש.

Netcat - פשוט nc יש יישומים שונים שיכולים להיות בעלי ערך רב לפנטסטר. Netcat תומך גם בפרוטוקולי TCP וגם בפרוטוקולי UDP. זה יכול לתפקד כלקוח שמתחבר ליציאת האזנה; לחלופין, הוא יכול לפעול כשרת שמאזין ביציאה לבחירתך. לפיכך, זהו כלי נוח שאתה יכול להשתמש בו כלקוח או שרת פשוטים באמצעות TCP או UDP.

ראשית, ניתן להתחבר לשרת, כפי שעשית עם Telnet, כדי לאסוף את הבאנר שלו באמצעות nc 10.10.100.137 PORT, שהוא די דומה ל-telnet 10.10.100.137 PORT הקודם שלנו. שים לב שייתכן שתצטרך ללחוץ על SHIFT+ENTER אחרי שורת GET.  
  
סיכום:  
בחדר זה, כיסינו כלים רבים ושונים. קל לחבר כמה מהם יחד באמצעות סקריפט מעטפת כדי לבנות סורק רשת ומערכת פרימיטיבי. אתה יכול להשתמש ב-traceroute כדי למפות את הנתיב למטרה, פינג כדי לבדוק אם מערכת היעד מגיבה ל-ICMP Echo, ו-telnet כדי לבדוק אילו יציאות פתוחות וניתנות לגישה על ידי ניסיון להתחבר אליהן. סורקים זמינים עושים זאת ברמות מתקדמות ומתוחכמות הרבה יותר, כפי שנראה בארבעת החדרים הבאים עם nmap.

///Nmap Live Host Discovery

סיכום של מה שנעשה - אנו מציגים את הגישות השונות בהן משתמשת Nmap כדי לגלות מארחים חיים. בפרט, אנו מכסים:

סריקת ARP: סריקה זו משתמשת בבקשות ARP כדי לגלות מארחים חיים  
סריקת ICMP: סריקה זו משתמשת בבקשות ICMP כדי לזהות מארחים חיים  
סריקת ping TCP/UDP: סריקה זו שולחת מנות ליציאות TCP ויציאות UDP כדי לקבוע מארחים חיים.

אנחנו גם מציגים שני סורקים, arp-scan ו-masscan, ומסבירים כיצד הם חופפים לחלק מגילוי המארח של Nmap.

Network Segment - זהו אוסף של מחשבים המקושרים יחד באמצעות "מדיום" משותף.  
במקרה זה, ה"מדיום" יכול להיות מתג Ethernet או נקודת גישה ל-WiFi.  
Physical connection  
Subnetwork - היא בדרך כלל המקבילה למקטע רשת אחד או יותר המקושרים יחד ומוגדרים לשימוש באותו נתב ברשת IP.

ARP חבילה שעוברת רק תחת אותו subnetwork, ולא תעבור דרך מקטעים שונים.  
חבילה ששולחת לכל המכשירים ברשת בבקשה לקבל בחזרה את כתובת הMAC של המכשיר.  
1) ARP משכבת קישור - בשימוש נפוץ עבור Man-In-Middle-Attack (זיוף ARP והרעלת ARP)

2) ICMP מ- Network Layer - בשימוש נפוץ בהתקפות DDoS

3) TCP משכבת התחבורה - הצפה SYN, איפוס TCP, התקפות חטיפת הפעלה של TCP

4) UDP מ-Transport Layer - משמש גם בהתקפות DDoS, שבהן התוקף מכוון ומציף יציאות אקראיות במארח עם מנות IP המכילות מנות User Datagram Protocol (UDP)  
  
פינג למערכת באותה רשת דורש שאילתת ARP לפני ה-ICMP Echo.  
למרות העובדה ש-TCP ו-UDP הם שכבות תחבורה, סורק יכול לשלוח חבילה שנבנתה במיוחד ליציאות TCP או UDP נפוצות כדי לראות אם המטרה מגיבה.

אסטרטגיה זו יעילה, במיוחד כאשר ICMP Echo מושבת.

1. When a privileged user tries to scan targets on a local network (Ethernet), Nmap uses ARP requests.

- A privileged user is root or a user who belongs to sudoers and can run sudo.

2. When a privileged user tries to scan targets outside the local network, Nmap uses ICMP echo requests, TCP ACK (Acknowledge) to port 80, TCP SYN (Synchronize) to port 443, and ICMP timestamp request.

3. When an unprivileged user tries to scan targets outside the local network, Nmap resorts to a TCP 3-way handshake by sending SYN packets to ports 80 and 443.

אנו משתמשים בפקודה nmap -PM -sn MACHINE IP/24 כדי לנסות למצוא מארחים חיים באמצעות שאילתות מסיכת כתובת ICMP.

למרות העובדה שאנו יודעים שלפחות שמונה מארחים נמצאים באינטרנט על סמך סריקות קודמות, סריקה זו לא הניבה אף אחת.

הסיבה לכך היא שמערכת היעד או חומת האש במסלול מונעים שליחת מנות ICMP מסוג זה. כתוצאה מכך, לימוד טכניקות מרובות כדי להגיע לאותה תוצאה הוא קריטי. אם סוג אחד של חבילה נאסר, אנו עשויים תמיד לנסות אחרת כדי ללמוד על רשת היעד והשירותים.

///Nmap Basic Port Scans

בפרק הקודם ראינט כיצד ניתן לבצע סריקה ולזהות מכשירים נוספים ברשת שלי וברשתות נוספות.  
כעת, לאחר שזיהינו מכשירים נוספים נרצה לבצע סריקה ולמצוא כניסות, portים פתוחים.  
  
TCP : אנו מעוניינים ללמוד האם יציאת ה-TCP פתוחה, ולא ליצור חיבור TCP. מכאן שהחיבור נקרע ברגע שמצבו מאושר על ידי שליחת RST/ACK. אתה יכול לבחור להפעיל את סריקת TCP connect באמצעות -sT.  
  
משתמשים ללא הרשאות מוגבלים לסריקת חיבור. עם זאת, מצב הסריקה המוגדר כברירת מחדל הוא סריקת SYN, והוא דורש משתמש בעל הרשאה (שורש או sudoer) כדי להפעיל אותו. סריקת SYN אינה צריכה להשלים את לחיצת היד התלת-כיוונית של TCP; במקום זאת, הוא מנתק את החיבור ברגע שהוא מקבל תגובה מהשרת. מכיוון שלא יצרנו חיבור TCP, זה מקטין את הסיכוי שהסריקה תירשם. אנו יכולים לבחור סוג סריקה זה באמצעות האפשרות -sS. האיור שלהלן מראה כיצד פועלת סריקת TCP SYN מבלי להשלים את לחיצת היד התלת-כיוונית של TCP.  
  
 : UDPהוא פרוטוקול חסר חיבור, ומכאן שהוא אינו דורש לחיצת יד כלשהי ליצירת חיבור. אנחנו לא יכולים להבטיח ששירות שמאזין ביציאת UDP יגיב למנות שלנו. עם זאת, אם חבילת UDP נשלחת ליציאה סגורה, מוחזרת שגיאה בלתי ניתנת להשגה של יציאת ICMP (סוג 3, קוד 3). אתה יכול לבחור בסריקת UDP באמצעות האפשרות -sU; יתר על כן, אתה יכול לשלב אותו עם סריקת TCP נוספת.

האיור הבא מראה שאם אנו שולחים חבילת UDP ליציאת UDP פתוחה, לא נוכל לצפות לתשובה כלשהי בתמורה. לכן, שליחת חבילת UDP ליציאה פתוחה לא תספר לנו כלום.

///Nmap Advanced Port Scans

סריקות נוספות (מתקדמות יותר) למציאת portים פתוחים.  
3 הסריקות הראשונות מפעילות flags שונים ב TCP package.

Null Scan – ללא שום דגל דולק

FIN Scan – דגל אחד דולק

Xmas Scan – כל שלושת הדגלים דולקים.

Maimon Scan - בסריקה זו נקבעים סיביות FIN ו-ACK. המטרה צריכה לשלוח חבילת RST כתגובה. עם זאת, מערכות מסוימות הנגזרות מ-BSD מפילות את החבילה אם מדובר ביציאה פתוחה החושפת את היציאות הפתוחות. סריקה זו לא תעבוד על רוב היעדים שנתקלים בהם ברשתות מודרניות; עם זאת, חשוב שנכיר סריקה זו למקרים עתידיים.

ACK Scan - בואו נתחיל עם סריקת TCP ACK. כפי שהשם מרמז, סריקת ACK תשלח חבילת TCP עם סט דגל ה-ACK. השתמש באפשרות -sA כדי לבחור סריקה זו. כפי שאנו מראים באיור למטה, היעד יגיב ל-ACK עם RST ללא קשר למצב הנמל. התנהגות זו מתרחשת מכיוון שחבילת TCP עם ערכת דגל ACK צריכה להישלח רק בתגובה לחבילת TCP שהתקבלה כדי לאשר את הקבלה של נתונים מסוימים, בניגוד למקרה שלנו. לפיכך, סריקה זו לא תספר לנו אם יציאת היעד פתוחה בהגדרה פשוטה.  
סוג זה של סריקה יעזור אם יש חומת אש מול המטרה. כתוצאה מכך, על סמך אילו מנות ACK הביאו לתגובות, תלמד אילו יציאות לא נחסמו על ידי חומת האש. במילים אחרות, סוג זה של סריקה מתאים יותר לגילוי ערכות כללים ותצורה של חומת אש.

Window Scan - סריקה דומה נוספת היא סריקת חלון TCP. סריקת חלון TCP כמעט זהה לסריקת ACK; עם זאת, הוא בוחן את שדה TCP Window של מנות ה-RST המוחזרות. במערכות ספציפיות, זה יכול לחשוף שהיציאה פתוחה. אתה יכול לבחור סוג סריקה זה עם האפשרות -sW. כפי שמוצג באיור למטה, אנו מצפים לקבל חבילת RST בתגובה לחבילות ה-ACK ה"לא מוזמנים" שלנו, ללא קשר אם היציאה פתוחה או סגורה.

באופן דומה, הפעלת סריקת חלון TCP מול מערכת לינוקס ללא חומת אש לא תספק מידע רב. כפי שאנו יכולים לראות בפלט המסוף למטה, תוצאות סריקת החלון מול שרת לינוקס ללא חומת אש לא נתנו מידע נוסף בהשוואה לסריקת ACK שבוצעה קודם לכן.

עם זאת, כפי שהיית מצפה, אם נחזור על סריקת חלון ה-TCP שלנו מול שרת מאחורי חומת אש, אנו מצפים לקבל תוצאות משביעות רצון יותר. בפלט המסוף המוצג להלן, סריקת חלון ה-TCP הצביעה על כך ששלוש יציאות מזוהות כסגורות. (זה בניגוד לסריקת ACK שסימנה את אותן שלוש יציאות כבלתי מסוננות.) למרות שאנו יודעים ששלוש היציאות הללו אינן סגורות, אנו מבינים שהן הגיבו אחרת, מה שמצביע על כך שחומת האש אינה חוסמת אותן

Custom Scan – סריקה מותאמת אישית.

Spoofing IP , Spoofing MAC, Decoy Scan- דרך לבצע בדיקת פורטים פתוחים לא מהIP שלנו, ע"י שינוי הIP בעת שליחת ההודעה או שליחה ממספר רב של מכשירים (באותו אופן ניתן לבצע זאת על מספרי MAC).

Fragmented Packets - מערכת זיהוי חדירה (IDS) בודקת חבילות רשת עבור דפוסי התנהגות נבחרים או חתימות תוכן ספציפיות. זה מעלה התראה בכל פעם שכלל זדוני מתקיים. בנוסף לכותרת ה-IP ושכבת התעבורה, IDS יבדוק את תוכן הנתונים בשכבת התעבורה ויבדוק אם הוא תואם לדפוסים זדוניים כלשהם. כיצד תוכל להפחית את הסיכוי שחומת אש/IDS מסורתית תזהה את פעילות ה-Nmap שלך? לא קל לענות על זה; עם זאת, בהתאם לסוג חומת האש/IDS, ייתכן שתפיק תועלת מחלוקת החבילה למנות קטנות יותר.  
מנות מפוצלות Nmap מספקת את האפשרות -f לפצל מנות. לאחר הבחירה, נתוני ה-IP יחולקו ל-8 בתים או פחות. הוספת עוד -f (-f -f או -ff) תפצל את הנתונים ל-16 קטעי בתים במקום 8. אתה יכול לשנות את ערך ברירת המחדל באמצעות ה- --mtu; עם זאת, עליך לבחור תמיד בכפולה של 8.

Idle/Zombie Scan - זיוף של כתובת ה-IP המקור יכולה להיות גישה מצוינת לסריקה בגניבה. עם זאת, זיוף יעבוד רק בהגדרות רשת ספציפיות. זה מחייב אותך להיות במצב שבו אתה יכול לעקוב אחר התנועה. בהתחשב במגבלות אלו, זיוף כתובת ה-IP שלך עשוי להועיל מעט; עם זאת, אנו יכולים לתת לו שדרוג עם סריקה סרק.  
סריקת הסרק, או סריקת זומבים, דורשת מערכת סרק המחוברת לרשת שאיתה תוכל לתקשר. באופן מעשי, Nmap יגרום לכל בדיקה להיראות כאילו מגיעה מהמארח הבטל (זומבי), ואז היא תבדוק אינדיקטורים אם המארח הבטל (זומבי) קיבל תגובה כלשהי לבדיקה המזויף. זה מושג על ידי בדיקת ערך זיהוי ה-IP (מזהה IP) בכותרת ה-IP. אתה יכול להריץ סריקת סרק באמצעות nmap -sI ZOMBIE\_IP 10.10.159.147, כאשר ZOMBIE\_IP היא כתובת ה-IP של המארח הבלתי פעיל (זומבי).



.

//Threat Modelling

//

1. צופן קיסר וצפני הזזה אחרים.   
   על אף מורכבות הצופן ניתן להיעזר בשכיחות אותיות על מנת לזהות את הקידוד של האות.  
   אתר ניסיון לצבירת צופן – quipqiup
2. הצפנה סימטרית - קיימות תוכנות רבות להצפנה סימטרית. נתמקד בשניים, שנמצאים בשימוש נרחב גם להצפנה אסימטרית: א. GNU Privacy Guard ב. OpenSSL Project
3. הצפנה אסימטרית - הצפנה אסימטרית מאפשרת להחליף הודעות מוצפנות ללא ערוץ מאובטח; אנחנו רק צריכים ערוץ אמין. בערוץ אמין, אנו מתכוונים בעיקר לשלמות הערוץ ולא בסודיות.

כאשר משתמשים באלגוריתם הצפנה א-סימטרי, היינו יוצרים זוג מפתחות: מפתח ציבורי ומפתח פרטי. המפתח הציבורי משותף עם העולם, או ליתר דיוק, עם האנשים שרוצים לתקשר איתנו בצורה מאובטחת. המפתח הפרטי חייב להישמר בצורה מאובטחת, ולעולם אסור לנו לתת לאף אחד לגשת אליו. יתרה מכך, לא ניתן לגזור את המפתח הפרטי למרות הכרת המפתח הציבורי.

איך עובד צמד המפתחות הזה?

אם הודעה מוצפנת במפתח אחד, ניתן לפענח אותה במפתח השני. במילים אחרות:

אם אליס מצפינה הודעה באמצעות המפתח הציבורי של בוב, ניתן לפענח אותה רק באמצעות המפתח הפרטי של בוב.

הפוך, אם בוב מצפין הודעה באמצעות המפתח הפרטי שלו, ניתן לפענח אותה רק באמצעות המפתח הציבורי של בוב.  
הצגנו גם את RSA – פירוק למספרים ראשונים והצפנה.

1. דיפי-הלמן - הוא אלגוריתם הצפנה א-סימטרי. זה מאפשר החלפת סוד בערוץ ציבורי. נדלג על הרקע האריתמטי המודולרי ונספק דוגמה מספרית פשוטה. נצטרך שתי פעולות מתמטיות: הספק ומודולוס. xp, כלומר, x מוגדל בחזקת p, הוא x מוכפל בעצמו p פעמים. יתר על כן, x mod m, כלומר, x modulus m, הוא שאר החלוקה של x ב-m.  
   למרות שמאזן למד את הערכים של q, g, A ו-B, הם לא יוכלו לחשב את המפתח הסודי שאליס ובוב החליפו. השלבים לעיל מסוכמים באיור למטה.
2. פונקציית גיבוב ( Hash) – חד כיווני, מאפשר זיהוי של קלטים שהשתנו אפילו באות.

HMAC הוא סוג מסוים של קוד אימות הודעות (MAC) המספק דרך לאמת הן את האותנטיות והן את שלמות ההודעה או הנתונים. נעשה בו שימוש נפוץ בהצפנה ובפרוטוקולי אבטחה כדי להבטיח שהנתונים לא טופלו במהלך השידור.

קוד אימות הודעה (MAC):

MAC הוא פיסת מידע קצרה, לרוב מחרוזת באורך קבוע, שנוצרה מהודעה ומפתח סודי.

מטרת MAC היא לאשר שההודעה לא שונתה (שלמות) ושהיא מגיעה ממקור לגיטימי (אימות).

1. תשתית מפתח ציבורי (PKI):PKI היא מסגרת של מדיניות ונהלים המשמשים לניהול מפתחות דיגיטליים (מפתחות ציבוריים ופרטיים) ותעודות דיגיטליות. הוא מספק את הבסיס לתקשורת מאובטחת ונמצא בשימוש נרחב למשימות כמו הצפנה, חתימות דיגיטליות ואבטחת תקשורת רשת.  
   רשויות אישורים (CA): רשויות אישורים הן ארגונים מהימנים המנפיקים אישורים דיגיטליים. הם מאמתים את זהות מבקשי האישורים לפני הנפקת אישורים. CAs ידועים כמו DigiCert, GlobalSign ו-Let's Encrypt זוכים להכרה רחבה ומהימנה.
2. ניהול ושמירת סיסמאות בדטה בייס – רגיל (הכי גרוע), אחרי hashing (יותר מאובטח) , אחרי hashing עם salt (הכי מאובטח).

//

MITER ATT&CK (טקטיקות יריבות, טכניקות וידע משותף) היא מסגרת ובסיס ידע המשמשים באבטחת סייבר כדי לתאר את הטקטיקות, הטכניקות והנהלים השונים (TTPs) בהם משתמשים יריבים לביצוע התקפות סייבר. זהו משאב מקיף המסייע לארגונים להבין ולהגן מפני איומי סייבר בצורה יעילה יותר.  
להלן פירוט של מרכיבי המפתח של MITER ATT&CK:  
טקטיקות יריבות: אלו הן יעדים או יעדים ברמה גבוהה שיריבי סייבר שואפים להשיג במהלך מתקפה. MITER ATT&CK מגדיר מספר טקטיקות, כגון גישה ראשונית, הוצאה להורג, התמדה, הסלמה של הרשאות, התחמקות מהגנה, גישה לאישורים, גילוי, תנועה לרוחב, איסוף, הסתננות והשפעה. כל טקטיקה מייצגת שלב או היבט אחר של מתקפת סייבר.

טכניקות: טכניקות הן השיטות או הגישות הספציפיות בהן משתמשים יריבים כדי להשיג את המטרות הקשורות לכל טקטיקה. לדוגמה, אם ניקח בחשבון את טקטיקת ה"ביצוע", יכולות להיות טכניקות שונות שתוקף עשוי להשתמש בהן, כגון ביצוע קוד באמצעות קובץ מצורף זדוני או ניצול של פגיעות.

LUKS

ישנן מערכות תוכנה וכלים שונים המספקים הצפנה למערכות לינוקס. מכיוון שהפצות לינוקס מודרניות רבות נשלחות עם LUKS (Linux Unified Key Setup), בואו נסקור את זה ביתר פירוט.

SMB הוא פרוטוקול שיתוף קבצים המנוצל על ידי האקרים בטבע. הפרוטוקול משמש בעיקר לשיתוף קבצים ברשת; לכן, עליך להשבית את הפרוטוקול אם המחשב שלך אינו חלק מרשת על ידי הוצאת הפקודה הבאה ב- PowerShell.

הגנה על מערכת שמות דומיין מקומית (DNS)

מערכת שמות הדומיין (DNS) היא מערכת שמות המתרגמת את שמות הדומיין המלאים (FQDN) לכתובות IP. אם התוקף ממקם את עצמו באמצע, הוא עלול ליירט ולתפעל בקשות DNS ולהפנות אותן למערכות הנשלטות על ידי תוקף מאחר שתשובות DNS אינן מאומתות ואינן מוצפנות.

//Initial access //

Red Team Recon   
פקודות שימושיות:  
whois – מספק שירותי ספק, רושם, פרטי יצירת קשר ועוד.  
nslookup – מיפוי כתובת IP לשם דומיין.

traceroute – מספק את החיבורים בין הראוטר לטרגט (דרך אותם מערכות שעבר).

כלים:  
\*\* shodan ו Censys המאפשרים סריקה רחבה עבור מכשירים ושירותים באינטרנט.  
\*\* Recon-ng וmaltego הם שניהם כלי סיור רבי עוצמה המשמשים אנשי אבטחה, חוקרים ובוחני חדירה לאיסוף מידע על מטרות, כולל ארגונים, יחידים ורשתות.

Weaponization

קישור לקובץ הסברים והרחבות - /home/danshl/Desktop/reverse\_shell/Windows/initial\_access - office&htma  
קישור לגיט לקובץ reverse shell ללא חסימה של מערכת הפעלה ווינדווס - https://github.com/cym13/vbs-reverse-shell

Passwords

\* סיסמאות דיפולטיביות נפוצות - <https://default-password.info/>  
\* יצירת קובץ סיסמאות מסטרינגים או קייבורדים באתר מסוים -   
 cewl -w list.txt -d 5 -m 5 http://thm.labs

\*username\_generator - יצירת פרמוטציות בהינתן מילים ע"י . / וכו’ – לטובת נסיון כניסת סיסמאות.

\* crunch יצירת קובץ סיסמאות עפ"י קריטריונים מסוימים.

\*\* Offline Attacks - \* שימוש ב hashcat לפיצוח סיסמאות מגובבות.  
 \* שימוש בhashid לזיהוי סוג הגיבוב.

\*\*Brute force: command for hydra -   
 \* hydra -l ftp -P passlist.txt ftp://10.10.x.x

\* support – ftp,ssh, smtp

Phising  
שימוש ב Gophish. עיון בתיקיית Phising.  
  
  
--------------חורי אבטחה בAPI------------------

\* Broken Object Level Authorisation (BOLA) – אפשרות שינוי הקלט וקבלת ערכים שונים.  
למשל שינוי הID והגעת למשתמש אחר ללא שום ביצוע Authorizaion-Token או בדיקת אימות אחרת.

\* Broken User Authentication (BUA) - אימות משתמש שבור (BUA) משקף תרחיש שבו נקודת קצה API מאפשרת לתוקף לגשת למסד נתונים או לרכוש הרשאה גבוהה יותר מזו הקיימת. הסיבה העיקרית מאחורי BUA היא יישום לא חוקי של אימות כמו שימוש בשאילתות דוא"ל/סיסמה שגויות וכו', או היעדר מנגנוני אבטחה כמו כותרות הרשאות, אסימונים וכו'.

\* Excessive Data Exposure – עודף נתונים שמגיע (לפעמים דרך הAPI נוכל לראות מה שאין בפרונט עקב סינון מצד הפרונטנד ואי סינון מצד הבקאנד).

\* Lack of Resources & Rate Limiting - חוסר במשאבים והגבלת קצב פירושה שממשקי API אינם אוכפים כל הגבלה על תדירות המשאבים המבוקשים של הלקוחות או על גודל הקבצים, מה שמשפיע קשות על ביצועי שרת ה-API ומוביל ל-DoS (מניעת שירות) או אי-זמינות של שֵׁרוּת. שקול תרחיש שבו מגבלת API לא נאכפת, ובכך לאפשר למשתמש (בדרך כלל פולש) להעלות מספר קבצי GB בו-זמנית או לבצע כל מספר של בקשות בשנייה. נקודות קצה כאלה של API יגרמו לניצול יתר של משאבים ברשת, אחסון, מחשוב וכו'.

\* Broken Function Level Authorisation - אישור רמת פונקציה שבור משקף תרחיש שבו משתמש בעל הרשאות נמוכה (למשל, מכירות) עוקף את בדיקות המערכת ומקבל גישה לנתונים סודיים על ידי התחזות למשתמש בעל הרשאות גבוהות (אדמין). שקול תרחיש של מדיניות בקרת גישה מורכבת עם היררכיות, תפקידים וקבוצות שונות והפרדה מעורפלת בין פונקציות רגילות וניהוליות המובילות לפגמי הרשאה חמורים. על ידי ניצול הבעיות הללו, הפולשים יכולים לגשת בקלות למשאבים הלא מורשים של משתמש אחר או, המסוכן ביותר - לפונקציות הניהוליות.

\* Mass Assignment - הקצאת המונים משקפת תרחיש שבו נתונים בצד הלקוח קשורים אוטומטית לאובייקטים בצד השרת או למשתני מחלקה. עם זאת, האקרים מנצלים את התכונה על ידי הבנת ההיגיון העסקי של האפליקציה ושליחת נתונים בעלי מבנה מיוחד לשרת, רכישת גישה מנהלתית או הכנסת נתונים מבולבלים.   
  
\* Security Misconfiguration - תצורת אבטחה שגויה מתארת הטמעה של בקרות אבטחה שגויות ותצורה גרועות שמעמידות את האבטחה של כל ה-API על כף המאזניים. מספר גורמים עלולים לגרום לתצורת אבטחה שגויה, כולל תצורת ברירת מחדל שגויה/לא שלמה, אחסון ענן נגיש לציבור, שיתוף משאבים חוצי מקור (CORS) והודעות שגיאה המוצגות עם נתונים רגישים. פולשים יכולים לנצל את התצורות השגויות הללו כדי לבצע סיור מפורט ולקבל גישה לא מורשית למערכת.  
  
\* Injection - התקפות הזרקה הן כנראה בין התקפות ה-API/האינטרנט הוותיקות ביותר ועדיין מבוצעות על ידי האקרים על יישומים מהעולם האמיתי. פגמים בהזרקה מתרחשים כאשר קלט המשתמש אינו מסונן ומעובד ישירות על ידי API; ובכך לאפשר לתוקף לבצע פעולות API לא מכוונות ללא אישור. הזרקה עשויה להגיע משפת שאילתות מבנה (SQL), פקודות מערכת הפעלה (OS), שפת סימון מתרחבת (XML) וכו'.

\* Improper Assets Management - ניהול נכסים בלתי הולם מתייחס לתרחיש שבו יש לנו שתי גרסאות של API זמינות במערכת שלנו; בואו נקרא להם APIv1 ו-APIv2. הכל הועבר לחלוטין ל-APIv2, אבל הגרסה הקודמת, APIv1, עדיין לא נמחקה. בהתחשב בכך, אפשר לנחש בקלות שלגרסה הישנה יותר של ה-API, כלומר, APIv1, אין את תכונות האבטחה המעודכנות או העדכניות ביותר. שפע של תכונות מיושנות אחרות של APIv1 מאפשרות למצוא תרחישים פגיעים, שעלולים להוביל לדליפת נתונים והשתלטות על השרת באמצעות מסד נתונים משותף בין גרסאות API. זה בעצם על אי מעקב נכון אחר נקודות קצה API. הסיבות הפוטנציאליות יכולות להיות תיעוד API לא שלם או היעדר תאימות למחזור החיים של פיתוח תוכנה. מלאי API מתוחזק כהלכה ועדכני ותיעוד מתאים הם קריטיים יותר מבקרת אבטחה מבוססת חומרה עבור ארגון.

\* Insufficient Logging & Monitoring - רישום וניטור לא מספיק משקף תרחיש שבו תוקף מבצע פעילות זדונית בשרת שלך; עם זאת, כאשר אתה מנסה לעקוב אחר ההאקר, אין מספיק ראיות זמינות עקב היעדר מנגנוני רישום וניטור. מספר ארגונים מתמקדים רק ברישום תשתיות כמו אירועי רשת או רישום שרת, אך חסרים רישום וניטור API. מידע כמו כתובת ה-IP של המבקר, נקודות הקצה שהגישה אליהם, נתוני קלט וכו', יחד עם חותמת זמן, מאפשרים לזהות דפוסי התקפת איומים. אם מנגנוני רישום לא קיימים, יהיה זה מאתגר לזהות את התוקף ואת פרטיו. כיום, מסגרות האינטרנט העדכניות ביותר יכולות לרשום באופן אוטומטי בקשות ברמות שונות כמו שגיאה, ניפוי באגים, מידע וכו'. ניתן לרשום שגיאות אלו במסד נתונים או בקובץ או אפילו להעביר לפתרון SIEM לניתוח מפורט.