פעולה לינארית = $f(x \oplus y) = f(x) \oplus f(y) + const$ - צופן המקיים תכונה זו קל לשבור אותו.

התקפת טבלה:
התוקף שומר אינפורמציה לגבי ההצפנה של בלוקי הכתב הגלוי ופיענוחם.
T_E - טבלת הצפנה, עבור הצפנת הבלוק M_i לבלוק כתב הסתר C_i המתאים לו.
T_D - טבלת פיענוח בלוקי כתבי הסתר C_i לבלוקי הכתב הגלוי M_i המתאים לו.

נשים לב שלאחר בניית הטבלה, אם התוקף אסף מספיק אינפורמציה על הבלוקים השונים, אזי התוקף יוכל להצפין ולפענח הודעות כרצונו, תוך שימוש בטבלאות שבנה.
התקפה זו אפשרית כאשר מס' הבלוקים האפשריים קטן, ולכן קל לאסוף אינפורמציה לגבי הבלוקים.

Known Plaintext Attack - בהתקפה מסוג זה לתוקף ידועים $M = M_1 M_2 ... M_n$ ו- $C = C_1 C_2 ... C_n$.
Chosen Plaintext Attack - אשר בה בנוסף התוקף יכול לבחור את ההודעה M שתוצפן ולקבל עבור הודעה M שבחר את ההצפנה C המתאימה לה.

חיפוש ממצה:
הגנה מפני חיפוש ממצה נדרוש שמספר המפתחות האפשריים יהיה גדול מאד- לדוגמא 2^{128} מפתחות אפשריים.

נתון כתב גלוי M וכתב סתר C.
<u>הנחה:</u> ההודעות המקוריות הן בעלות משמעות וניתן לזהות אותן.

- נעבור על כל המפתחות האפשרים ונראה איזה מפתח מצפין את M ל- C.
- אם אם יש יותר מאחד כזה, נניח שיש לנו דרך למצוא מי מהם הנכון (לדוגמא ע"י הודעה נוספת, עבורה יודעים את הכתב הגלוי וכתב הסתר)

התקפת שידור חוזר, *Replay Attack* - נניח שנשלחה ברשת הודעה חתומה / מוצפנת ע"י משתמש A ונניח שמתקיף E אינו מסוגל לזייף חתימה / הצפנה של A.
התקפה אפשרית היא: E יקליט את ההודעה המוצפנת / החתומה, וישדר אותה שוב ללא כל שינוי בריצה עתידית של הפרוטוקול.
<u>פיתרון:</u> הוספת מס' סידורי להודעה

התקפת האיש שבאמצע *MitM* - התוקף נכנס בין שני משתמשים ומשנה את ההודעות שמועברות בינהם.
<u>פיתרון:</u> מציאת קשר בין המפתח הפומבי לבעליו, למשל ע"י סרטיפיקטים.

sniffing* - התקפה ברמת ה *MAC - הקשבה למידע שלא מיועד אלינו, אך היא אפשרית רק ברשת מקומית.

***IP Spoofing* חד כיווני** - המטרה: התחזות.
זאת ע"י שינוי כתובת ה- *IP* של *host* (קל לביצוע).
ע"י זיוף כתובת ה- *IP* של השולח בחבילות הנשלחות, המתקיף *(A)* יכול לשכנע את המותקף שהחבילות שהוא שולח, מגיעות ממחשב אחר *(B)* , חבילות התשובה שהמחשב המותקף שולח לא יגיעו אל המחשב המתחזה, אלא אל מחשב *(B)*.

יתרונות:
- לעיתים זיהוי משתמש מתבצע על סמך כתובת *IP* בלבד.
- מניעת מעקב אחר התוקף האמיתי (בשילוב עם התקפה אחרת).
- שימושית כאשר מגבילים את כמות המשאבים עבור *IP* מסויים.

***IP Spoofing* דו כיווני -** המטרה: התחזות.
מאפשר למחשב מתחזה לשלוח חבילות מכתובת *IP* שלא שלו, ואף לקבל אליה תשובה (קשה מאד לביצוע, מצריך ביצוע שינויים ברשת (בנתבים), ולא רק במחשב המתחזה, בנוסף קשה מאד להתגוננות) - **במהלך הקורס התקפה זו תחשב כלא סבירה.**

***Syn Attack* -** מטרה: *DoS (Denial of Service)*
שולח המתקיף אל המותקף כמות גדולה מאד של חבילות ראשונות ב- *TCP sessions* במטרה להפיל את המותקף. למערכת ההפעלה תורים מיוחדים (וקטנים) המיועדים לשמירת נתונים של חיבורים באמצע לחיצת היד. שליחת כמות גדולה של חבילות, גורמת לניצול כל המקום בתורים אלו והמערכת לא יכולה לקבל חיבורים נוספים.

<u>הגנה אפשרית:</u> נגביל את מס' ה *sessions* המורשים עבור כתובת מקור *IP*.

Syn Attack + IP Spoofing -
<u>הגנה אפשרית:</u> *cookies*.
התקפה אפשרית: ***Distributed DoS*.**

Cookies

הרעיון: עם תחילת ה- *session* נשלח ליוזם הקשר *cookie* שהיא מחרוזת בלתי ניתנת לחיזוי מראש. לאחר מכן נדרוש לקבל אותה חזרה מיוזם הקשר. אם הוא מחזיר אותה, הרי שה- *cookie* אכן הגיע ליוזם הקשר, משמע היוזם נמצא ב- *IP* שהוא טוען ששייך לו (אחרת, זיהינו *IP Spoofing*).

***DNS Poisoning* -** מטרה: התחזות, גרימת נזק.
מכיוון שהקישור בין שם ה- *domain* וכתובת ה- *IP* שלו דורשת גישה לשרתי *DNS*, ומכיוון שפרוטוקול ה- *DNS* אינו בטוח, ניתן להתקיף את קו התפר הזה.
הרעיון הבסיסי - התוקף ישלח חבילה המתחזה להיות התשובה משרת *DNS* (במקום שכתובת לוגית תהיה מוכרת כ- *IP* מסויים, תהייה מוכרת כ- *IP* אחר, הנקבע ע"י התוקף).

***RST Attack* -** המטרה: להפריע לחיבור *TCP*.
המתקיף שולח אל המותקף כמות גדולה של חבילות *TCP* במטרה לאתחל מחדש חיבור *TCP* קיים. ל- *TCP* קיים מנגונון לסגירת *session*, וכן מנגנון לאיתחול *session*.

- מכיוון שיש צורך לבצע *injection* של חבילה ל- *TCP*, יש לנחש את ה- *sequence number* ואת הפורט אליו יש לשלוח רת ההודעה. מאחר ו- *TCP* מאפשר הגעת חבילות לא בסדר שליחה, ה- *sequence number* שיש לנחש הוא בטווח מסויים ולא ערך אחד נתון (בנוסף מסתבר כי רוב הנתבים בוחרים *sequence number* הניתנים לחיזוי).
- לרוב אין לכך משמעות, אך אם לאיכות החיבור יש השלכה על הפרוטוקול שרץ מעל ל- *TCP*, הפרוטוקול עלול ל"הזדהם" (לדוגמא: *BGP4*)

***Ping Flood* -** סוג של *DoS*
פרוטוקול נוסף שקיים מעל שכבת ה *IP*, הינו פרוטוקול *ICMP*, פרוטוקול זה עוזר לשכבת ה- *IP*, אך נמצא מעט מעליה.

- לדוגמא, כאשר מריצים *ping* לשרת מרוחק, נשלחת הודעה *ICMP* מסוג *ping* ומחכים לתגובת *pong* ממנו.

שליחת הודעות אלה יוצרות עומס על חיבור האינטרנט של המותקף. אם החיבור של התוקף מהיר ושל המותקף איטי, והוא עונה על כל שאילתת *ping*, הרי שאפשר לאלץ אותו לענות על הרבה הודעות שכאלו.

- היום ישנם פחות שרתים שמחוברים בחיבור איטי, ובנוסף, מגבילים מענה להודעות *ping*.

Cookie Poisoning (Http)

- לשים לב אלו לא עוגיות אשר נועדו למנוע *IP Spoofing*, אלה עוגיות *http*, המאפשרים לשרת לשמור את המצב של הגולש באתר, על הכונן הקשיח של המשתמש, כך שגם אם השרת נופל, או התקשורת ניתקת, המידע עדיין קיים וזמין.
- העוגיות נשמרות אצל המשתמש, ולכן יכול לשנות את תוכנן וערכן של העוגיות.

התקפת מילון

בדר"כ משתמשים יעדיפו לבחור סיסמאות שיהיה להם קל לזכור (סיסמאות שכיחות הכוללות, שם פרטי או שם משפחה) הדבר מאפשר לתוקף לנצל תכונה זו של אנשים לצורך צמצום מרחב הסיסמאות שיש לנסות לצורך פריצה לחשבון.

חטיפת הקשר - *Session Hijacking*

בהתקפה זו המתקיף מחכה עד שתהליך אימות המשתמש החוקי יסתיים, ולאחר מכן מנתק אותו (ע"י שליחת הודעה למשתמש שהקשר נותק) וממשיך את ה - *session* במקום המשתמש החוקי. זוהי בעצם התקפת *MitM* על פרוטקולי בקרת כניסה.

<u>הגנה:</u> רצוי תוך כדי תהליך האימות להסכים על מפתח סודי וחד פעמי *k* (שנקרא *session key*). לאחר מכן, כל ה - *session* מוגן עם המפתח הנ"ל.

הערות נוספות

צפני בלוקים - התקפת טבלה - בלוק גדול
חתימות דיגיטאליות - שידור חוזר - הוספת מס' סידורי
חתימות עם *RSA*: זיוף אפשרי, תכונת הכפל, אסור להשתמש באותו מפתח לחתימה ולהצפנה