**המחלקה להנדסת תוכנה**

**פרויקט גמר תשע"ז**

**אבטחת נתבים ברשת בייתית**

**Securing wireless networks for home routers**

**ליאור ספיר liorsap1@gmail.com**

**יועד שירן shooki@gmail.com**

**מנחה אקדמי: ד'ר אללוף מרים.**

**אישור: תאריך:**

**רכזי הפרוייקטים: ד'ר אללוף מרים ו-ד'ר יגל ראובן.**

**אישור: תאריך:**

|  |  |  |
| --- | --- | --- |
| **#** | **מערכת** |  |
| **1** | **מאגר קוד** | **https://github.com/sh00ki/JCE-Project** |
| **2** | **יומן** | **https://trello.com/b/Ter3qxak/yoad-lior-project** |
| **3** | **ניהול פרוייקטים** | **Github** |

**מבוא**

היום, השימוש ברשתות אלחוטיות (WIFI) הופך להיות כל כך שכיח, יש צורך בנקיטת אמצי הגנה ואבטחה על הרשתות האלחוטיות. מעצם טיבעה, תקשורת אלחוטית מתאפיינת בכך שהמידע המשודר מהמחשב ואליו חשוף להאזנה. באמצעות כלים פשוטים, ניתן להאזין לתשדורת האלחוטית, מה שעלול להביא לחשיפה של מידע רגיש ומקורות שאין אנו רוצים לחשוף בפני אחרים.

בנוסף על כך, רשת אלחוטית בייתית או משרדית שאינה מאובטחת כיאות לעולם הגדול, חושפת את הנתב ומשתמשי הרשת הביתית עצמה, להאקרים וגורמים זרים המעוניינים לזרוע בה הרס, לגזול רוחב פס ולדלות אינפורמציה רגישה מבלי שמשתמשי הרשת ידעו על כך.

על מנת לפרוץ לנתב האלחוטי ולרשת הביתית, כל מה שנדרש הוא מחשב נייד, תוכנה וקצת סבלנות. לכן יש צורך חיוני ליישם מנגנוני אבטחה שיגנו על הרשת האלחוטית.

ברצוננו להדגיש כי אין אבטחה ברמה מושלמת לשום רשת אלחוטית. פורץ או גורם זר עיקש וחדור מטרה ברורה, עם אמצעים להשגתה, יוכל להתגבר על כל מנגנון אבטחה שניישם במהלך הפרויקט. עם זאת ובהנחה שעל המחשבים לא שמורים סודות חשובים ביותר, יישום האבטחה שנציג בפרויקט, יגביר את בטחון הרשת האלחוטית בבית ובמשרדים קטנים.

על מנת להבין את הבעיה שאנו רוצים להציב בפרויקט שלנו ולפתור אותה במהלך הפרויקט עלינו להבין את כל נושא אבטחת מידע הרלוונטיים עבור הפרויקט שלנו. הנושאים הרלוונטיים עבור הפרויקט שלנו אלה פרוטוקולים של תקשורת אלחוטית העובדים בתקן IEEE 802.11.

**IEEE 802.11** זהו אוסף סטנדרטים הנושאים את הרשת המקומית האלחוטית בתדרים 2.4Ghz, 5Ghz.

תקן זה מכיל שיטות אפנון לצורך העברת המידע ברשת האלחוטית ושימוש בפרוטוקול בסיסי שמשמש לרשת מקומית (LAN).

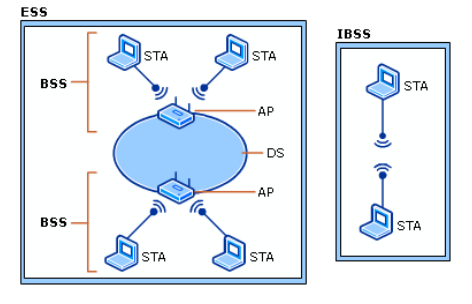
המבנה הבסיסי של 802.11 מכיל מספר מרכיבים עיקרים : תחנה (STA), נקודת גישה אלחוטית (AP), מערך שירות בסיסי עצמאי (IBSS), מערך שירות בסיסי (BSS), מערכת הפצה (DS), ומערכת מורחבת(ESS).

IBSS זה רשת אלחוטית המכילה לפחות שתי STA שעובדות ללא תלות במערכת הפצה (DS). רשת זאת נקראת Ad-hoc.

BSS זה רשת אלחוטית שמכילה AP אחד התומך בלפחות STA אחד. רשת זו נקראת רשת אלחוטית בסיסית.

כל STA ברשת BSS מתקשר דרך AP, וAP משמש כמתווך לרשת LAN הקווית בין תחנות STA שמחוברות מחוץ לרשת האלחוטית.

דוגמה לכך אפשר לראות באיור הבא:



חשוב לציין שצורת הגישה לרשת אלחוטית שונה מצורת הגישה לרשת קווית וזה כי לרשת אלחוטית חסר את הפרטיות המינימלית שמקבלים ברשת הקווית. בעוד שהרשת הקווית יכולה להיחשף רק למי שיש גישה פיזית לנקודת הרשת.

עקב חוסר הפרטיות ברשת האלחוטית נצטרך לספק אמצעי אבטחה שמחולק לשלושה חלקים:

1. אימות – כל STA מחויב להזדהות וליצור שייכות עם AP לפני שהוא משדר מידע, השייכות מתבצעת באמצעות שני קריטריונים:

- אימות במערכת פתוחה – זאת מערכת AP שמבצעת אימות ללא סיסמה וכולם יכולים להתחבר

- אימות במפתח משותף – זאת מערכת AP שמחויב במפתח משותף ועל ידי כך מקשה את ההתחברות לרשת האלחוטית עבור STA שאינו רצוי. כלומר כל אחד מחברי הרשת חייבים לדעת מהי סיסמת ההתחברות.

וזאת ע"י כך ש:

1. STA מנסה להתקשר לAP ע"י יצירת תקשורת (הקמת קשר).

2. ה-AP עונה לSTA ע"י מחרוזת רנדומלית כאתגר זיהוי.

3.STA מצפין את המחורזת בעזרת מפתח משותף שלו ומחזיר את המחרוזת לAP.

4.הAP מפענח את הקוד שקיבל בעזרת המפתח המשותף ובודק האם המחרוזת שהתקבלה זהה לזו ששלח בשלב 2, אם כן הSTA יקבל הודעה שהוא הצליח להתחבר ואחרת הוא יקבל הודעת שגיאה.

2. סינון – הAP מבצע יכולות לסנן תחנות שמתחברות אליו ע"י סינון כתובות MAC או סינון כתובות IP.

השיטות סינון ע"י MAC או IP יכולות להיות מיותרות כאשר האקר יכול לזייף את הכתובת MAC/IP וע"י כך להתחזות לגורם שמחובר לרשת. האקר עושה זאת ע"י האזנה למידע שעובד ברשת ודגימה של המידע הזה.

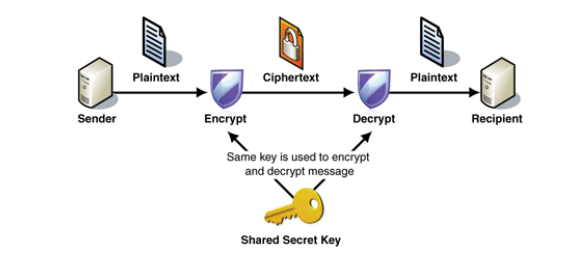
3. הצפנה – זהו תהליך שהAP הופך מסך מקורי למספר מוצפן.

הAP עושה זאת בשביל להבטיח פרטיות עבור הSTA שמחוברים ברשת האלחוטית ובשביל שהמידע אינו יגיע לגורם שאינו מחובר לרשת (מאזין).

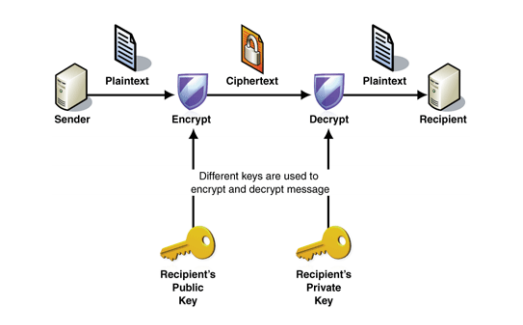
ההצפנה מתבצעת ע"י העברת המידע דרך נוסחת הצפנה מסוימת (מפתח).

קיום יש שני סוגים של הצפנות נפוצות.

1. הצפנה סימטרית – הצפנה שבה אותו מפתח מושמש כדי להצפין את המידע. בהמשך נפרט על מנגון WEP שעובד על הצפנה סימטרית.



2. הצפנה אסימטרית – הצפנה שבה משתמשים במפתח אחד להצפנת המידע ובמפתח אחר לפענוח ההצפנה.(WPA)



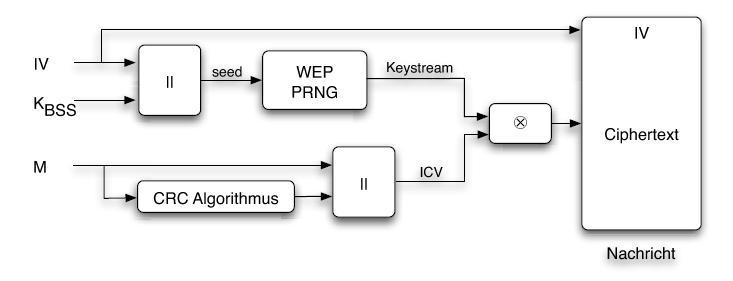
Wep – Wired Eqivalent Privacy – הוא אלגוריתם לאבטחת רשת נתונים אלחוטית העומדת בתקן IEEE 802.11. מנגנון אשר מאפשר הצפנה סימטרית. WEP משתמש במפתחות הצפנה באורכים של 40 bit או 104 bit. דרך פעולתו של האלגוריתם היא: הצפנת זרם סימטרי אשר לוקח את גוף הframe אשר מכיל את המידע הרצוי לשליחה. מעביר אותו דרך הצפנה שעליה נפרט בהמשך ומשדר אותו לאוויר. התחנה (משתמש קצה או AP) קולטת את המידע המשודר ומשתמשת באותה דרך הצפנה על מנת לפענח את המידע שנשלח.

WEP מצפין אך ורק את אותו הframe שבו מכיל מידע הרצוי לשליחה ולא את ה-header של חבילת המידע. כך משאיר את כתובת המען והנמען חשופים לכל מאזין. נרחיב על חולשותיו של הפרוטוקול בהמשך.

מנגנון האימות של WEP מורכז מ-4 הודעות בין המשתמש לבין הAP.

* ההודעה הראשונה – שליחת ACK התחברות לAP כלומר המשתמש מבקש להתחבר לרשת.
* ההודעה השנייה – הAP מקבל את הבקשה ושולח אתגר (Challenge) למשתמש שאם הוא יענה נכון על האתגר הוא יצליח להתחבר.
* ההודעה השלישית – המשתמש עונה לאתגר ושולח את המידע לAP.
* ההודעה הרביעית – במידה והתשובה נכונה הAP יורשה למשתמש להתחבר לרשת.

מנגנון ההצפנה של WEP משתמש באלגוריתם ההצפנה RC4, אשר פותח על ידי אותה חברה אשר פיתחה אלגוריתם הצפנה נוסף ה- RSA. כיום לWEP יש 3 צורות הצפנה ומה שמבדיל ביניהם הוא אורך המילה שעל הפרוטוקול להצפין. 64, 128ו- 256 ביט אורך מילה. כל אותם ההצפנות פועלות על אותו עקרון רק עם מימוש מעט שונה המבדיל אחד מהשני. נסביר את הרעיון כאן: נדגים זאת על ההצפנה בעלת אורך 64 תווים. כדי ליישם את הפרוטוקול WEP יש לייצר ווקטור אתחול שמטרתו להחזיק 2^24 סדרות תווים שונים אחד מהשני. כל רצף תווים בעל גודל 24 bit נקרא keystream. בנוסף לווקטור האתחול יש להחזיק מפתח קבוע באורך 40 ביט כאשר ל40 ביט של המפתח משורשר כל פעם של שליחה עוד 24 תווים שונים כל פעם מאותו ווקטור אתחול. כל פעם יש לנו תשדורת של 64 ביטים המכילים 40 ביט מפתח ועוד 24 תווים מהווקטור אתחול לפי רצף אשר מגיעים בתבנית קבועה וברצף מסוים, כל פעם 24 תווים אחרים לפי הסדר שנמצאים באותו הווקטור.



מספר חולשות קריטיות התגלו על ידי חוקרים והאקרים, מה שהוביל לכך שהפרוטוקול WEP ניתן לפריצה ובאמצעות כלי תוכנה זמינים ומתאימים תוך זמן קצר יחסית. כעת נסביר כיצד ניתן לנצל את חולשת ה-WEP לצורך הבנת המפתח המועבר דרך הפרוטוקול המדובר. פענוח של המידע המוצפן בעזרת הפרוטוקול נעשה באופן הבא: יש להאזין לכל חבילות המידע הנשלחות ברשת האלחוטית ולשמור את כל המידע שמועבר. לדוגמה ב- WEP של 64 bit יש להאזין 2^24 חבילות מידע ולדגום את כל רצפי התווים הנמצאים בווקטור. כאשר בחבילה הבאה מזהים רצף תווים כזה - keystream שכבר הגיע בעבר, ניצור keystream שהוא זהה לזה שנוצר בתהליך ההצפנה (במידה וגם המפתח הסודי זהה). Keystream זה והקוד (plaintext) עוברים פעולת XOR וכל מקבלים את המידע. על ידי גילוי keystream אחד, תוקף יכול לשדר מידע מוצפן ללא חשש או לפענח חבילת מידע הנשלחת ברשת האלחוטית בעזרת האזנה לאותה הרשת.

עקב חולשותיו של הפרוטוקול WEP, התכנסו חוקרים רבים ומהנדסים מההתאחדות של IEEE 802.11 והחליטו ולשפר את פרוטוקול האבטחה האמון על הרשת האלחוטית WIFI והגיעו לפיתוח תקן חדש IEEE 802.11i. אותה התאחדות הגיעה לפיתוח פרוטוקול אבטחת רשת חדש המיישם את אותו סטנדרט מוזכר הנקרא – WPA=Wi-Fi Protected Access.

הגרסה הראשונית של אותו הפרוטוקול הייתה להחליף את הפרוטוקול ה-WEP הישן. אותו פרוטוקול פותח בעקבות פריצות חמורות שנמצאו על ידי מומחים במערכת העובדת על הפרוטוקול הקודם.

מהו WPA – Wi-Fi Protected Access

הפרוטוקול משתמש באותו הרעיון כפי שהפרוטוקול הישן WEP אוחז אך עם מספר שינויים אשר מחזק את אבטחתו ומתגבר על החולשות של הפרוטוקול הקודם.

פרוטוקול WPA מביא שני שיפורים מפרוטוקול WEP:

1. הצפנת נתונים משופרת דרך פרוטוקול אמינות מפתח זמני שנקרא – TKIP-temporal key integrity protocol.

WPA משתמש ב-TKIP שהוא פרוטוקול מפתח זמני. הצפנת המפתחTKIP בפרוטוקול החליפה את המפתח הקטן יחסית אשר נמצא בWEP, בגודל של 40 [ביט](https://he.wikipedia.org/wiki/%D7%A1%D7%99%D7%91%D7%99%D7%AA), שאינו משתנה וקבוע. ה-TKIP הוא מפתח של 128 ביט לכל [חבילת מידע](https://he.wikipedia.org/wiki/%D7%97%D7%91%D7%99%D7%9C%D7%AA_%D7%9E%D7%99%D7%93%D7%A2) והוא יוצר באופן דינאמי ומשתנה מפתח חדש לכל חבילת מידע הנשלחת וכך מונע התנגשויות ורצף תווים חוזרים.

2. אימות המשתמש – בWEP היה תהליך של אימות ע"י 4 הודעות בין הSTA לAP.

ב-WPA יש הרחבה של תהליך האימות ע"י תהליך דומה שנקרא The 4 way handshake.

תהליך ה 4 way handshake הינו תהליך אימות בין STA לבין AP.

בתהליך זה מתבצע אימות שעונה על כל הקריטריונים הבאים :

* Authentication
* Access control
* Replay prevention
* Message modification detection
* Message privacy
* Key protection

תהליך ה 4 way handshake

הAP מוודא שהSTA הינו גורם אמין שמנסה להתחבר לרשת ע"י החזקת מפתח שנקרא Per Master Key(PMK) שאותו הוא יוצר מחדש עבור כל STA.

בשלב הראשון הAP שולח את הPMK לSTA (הודעה 1), כאשר הSTA מקבל את ההודעה מהAP הוא יוצר הודעה חדשה שנקראת Pairwise Transient Key (PTK), שהיא מכילה את כל הנתונים הבאים :

1. Authenticator nonce – זהו הnonce שהAP שלח בהודעה הראשונה.
2. Supplicant nonce – זהו הnonce של הSTA שהוא אשלח בהודעה השנייה.
3. Authenticator Address – הכתובת MAC של הAP שנשלחה בהודעה הראשונה.
4. Supplicant Address – הכתובת MAC של הSTA שתישלח בהודעה השנייה.

חוץ מה-4 הנתונים הPTK מחזיק 5 מפתחות שונים שנשתמש בהם בהמשך (64bytes), 5 המפתחות הינם:

1. KCK – Key Confirmation Key - מאפיין זה מכיל בתוכו את הסיסמה שנשלחת ברשת. (16bytes)
2. KEK – Key Encyption Key – מאפיין זה יצפין את הסיסמה שתישלח ברשת (הפריימים (מידע) של הeapol מכיל בתוכו KEK לצורך הצפנה) (16bytes)
3. Temporal Key - זהו מפתח זמני שישמש את הAP להצפין/לפענח את המידע שעובר בין הAP לSTA. (16bytes)
4. Temporal MIC 1 - נועד בשביל לחשב את האימות בתוך הAP האם הPTK נכון.
5. Temproal MIC 2 - נועד בשביל לחשב את האימות בתוך הSTA האם הEAPOL-Key נכון.

והוא שולח את כל המידע הזה לAP (הודעה 2 )

אם כל המידע של הודעה 2 נכון (אומת ע"י הAP) , נמשיך בתהליך להודעה 3, אחרת יפסיק התקשורת בין הSTA לAP.

ברגע שהAP מקבל את ההודעה השנייה מהSTA, הוא מבצע אימות של הדברים הבאים:

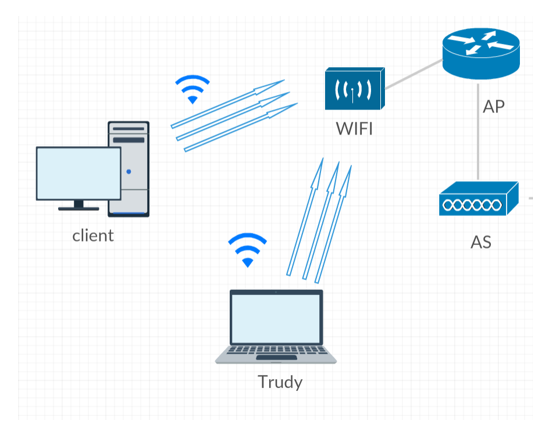
1. הוא בודק האם ניתן לגזור מחדש את ה PMK שהתקבל מההודעה הראשונה.
2. הוא בודק האם ה PMK נכון.
3. הוא בודק האם ה PTK שנשלח בהודעה השנייה נגזר נכון מהPMK של הסעיף השני.

ברגע שכל המידע נבדק ונמצא אמין הAP מתחיל ליצור GTK – Group Temporal Key שמכיל את פרטי האימות ומאפשר גישה לSTA. (הודעה 3)

במידה והSTA קיבל את ההודעה השלישית (GTK) הוא שולח ACK של התחברות לAP ומודיע לו שהGTK אושר והותקן על הSTA.

תהליך האימות החדש (The 4 way handshake) ותהליך ההצפנה החדשה (TKIP) שיפר את פרוטוקול WEP וגרם לכך שפרוטוקול WPA יהיה מאובטח.

תיאור הבעיה:



בשני הפרוטוקולים (WEP,WPA) נשארה דרך אחת להיכנס לרשת שלא נפתרה ע"י IEEE 802.11 והיא מתקפת brute force ע"י האזנה לפקטות האימות ופיענוח של תהליך האימות.

brute force – זאת מתקפה נאיבית כאשר צד ג' מנסה לנחש רצף של סיסמאות ע"י מילון סיסמאות גדול עד שהוא מנחש נכון את הסיסמה ונכנס לרשת.

החולשה של WPA (וגם WEP) הינה שניתן להאזין למידע ע"י כלים חינמיים ברשת ולדגום את המידע שעובר ברשת. למעשה לא נוכל לדעת יותר מדי עד שלא נפענח את המפתח של המידע שעובר ברשת. ולכן כל מה שנשאר לדעת הוא לדגום את פקטות האימות ולפענח את המפתח ובכך להאזין לשאר המידע שעובר ברשת ולעשות פעולות נוספות.

אנו לא נוכל למנוע מההאקר את האזנה לרשת ולדגום את פקטות האימות אבל כן ננסה להוסיף/לשנות את דרך האימות ובכך למנוע מההאקר להיכנס לרשת.

בפרויקט זה נחקור ונרצה לאבטח רשת ביתית או משרדית קטנה ולנסות למנוע מתקפה מסוג brute force.

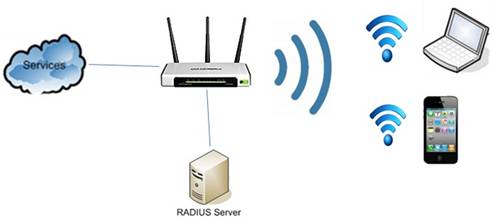
ברצוננו להבדיל בין שני סוגים של רשתות נתונים:

* Personal - רשת ביתית שמגדירם בתוך הנתב את הסיסמה של הרשת ואת סוג ההצפנה (WPA,WEP) ברשת זאת יש סיסמה **אחת** עבור כלל המשתמשים שמחוברים לרשת ועברו את שלב האימות.



* Enterprise – רשת משרדית שנועדה בשביל ארגונים גדולים (50 ומעלה משתמשים) ובה יש שרת RADIUS שמגדיר את האימות לרשת.

ברשת זאת ניתן לבנות דרך אימות חדשה/שונה חשוב לציין שרשת זאת מסובכת מאוד להקמה ומצריכה המון ידע על רשתות.



דרישות ואפיון הבעיה:

נתבונן כעט מנקודת המבט של המשתמש אשר ישתמש במערכת האבטחה.

הדרישה שלנו מהפיתוח ומהמחקר היא *מניעת האפשרות לבצע מתקפות מסוג brute force* וזה יצריך מאיתנו להתמודד עם התנאים הבאים:

* להתבסס על התקן IEEE 802.11 וזאת מהסיבה שרשתות ביתיות או משרדיות קטנות מתבססות ועובדות על תקן זה.
* התוכנה שנכתוב תצטרך להיות יעילה וקלה לשימוש עבור כל משתמש.
* התוכנה שנכתוב צריכה להיות בעלת יכולת אוטומטית ליכולת הפעלה וזיהוי ניסיון חדירה.
* נרצה לאפשר יכולת בקרה מרחוק על מצב הרשת ובמידת הצורך הכנסת גורם אנושי שיבצע את האימות שעליו נפרט בהמשך.

הבעיה מבחנת הנדסת תוכנה:

בפרויקט זה נעבוד מול תקן IEEE 802.11, אשר אוסף החוקים שלו צרובים בכל נתב בייתי/משרדי ולכן נצטרך להסתנכרן עם סביבת עבודה שעובדת עם התקן הזה ולכן נתכנת בסביבת עבודה Linux ונעבוד עם השפות (C,C++,Phtyon). ישנן מספר בעיות תכנותיות אשר אנו נאצלים לפתור:

1. שינוי של דרך האימות עבור AP בשביל למנוע את דגימת פקטות האימות של הEAPOL.
2. שאיבת נתונים בזמן אמת מAP לצורך זיהוי וניתור ניסיונות חדירה לרשת.

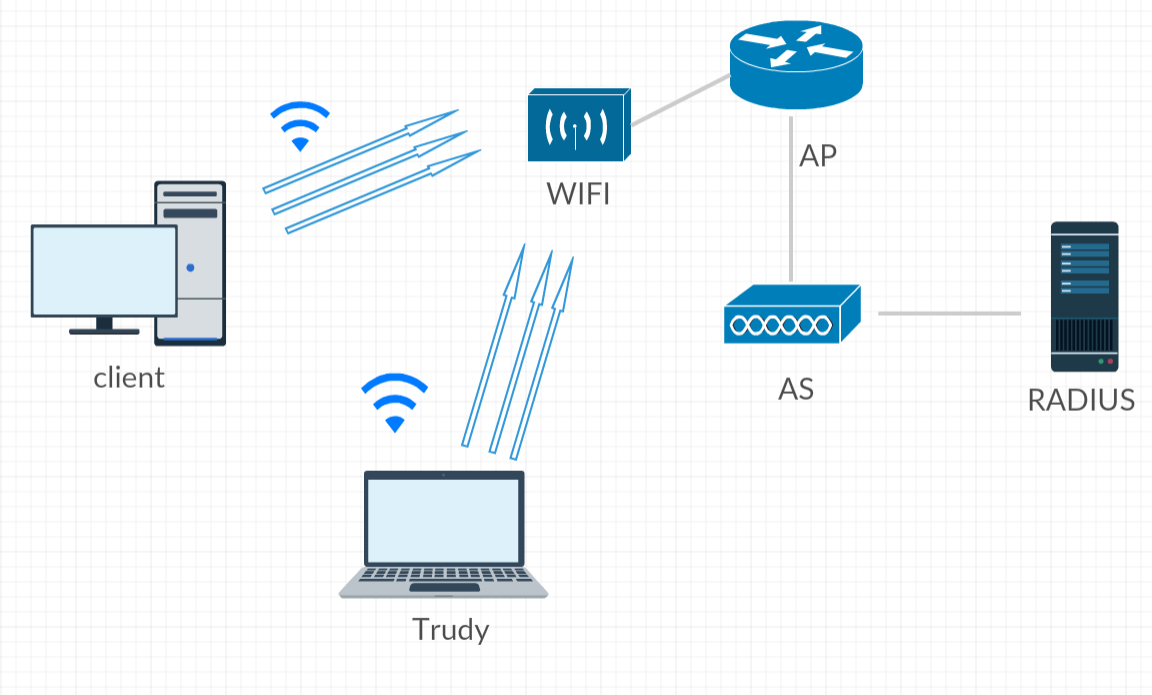
* ישנם המון סוגים של נתבים ונצטרך לדעת להתמודד עם שאיבת מידע לתוך אפליקציה בשביל שהמשתמש יוכל לבקר את המערכת.

1. הימנעות מפגיעה במשתמשים אשר מורשים גישה לAP אבל אינם יכולים להתחבר כתוצאה משילוב הפיתוח שלנו, נצטרך לפתח מערכת לומדת אשר מזהה ומכירה כל משתמש שמתחבר אליה.

קיום קיימים מספר פרוטוקולים שניתן להיעזר בהם (DHCP,NAT,VLAN) ולהשתמש בהם בפרויקט.

1. המערכת תהיה צריכה להיות קלה כמה שיותר יעילה בשביל שלא תעכב את תעבורת הרשת, המערכת בנוסף צריכה להיות קלה להתקנה ושקופה למשתמש.

תיאור הפתרון בשלבים: (נכון לעכשיו)

****

התרשים מתאר את ארכיטקטורת המערכת ומורכב מכמה חלקים:

* משתמש (client)
* האקר (Trudy) – מי שמאיים על תעבורת הנתונים של הרשת.
* WIFI – רכיב חומרתי שנמצא בתוך הAP והוא נועד לצורך שליחה וקבלת נתונים מהמשתמשים ברשת הפנימית והחוצה.
* AP – נתב
* AS – רכיב פנימי בתוך הנתב שהוא זה שמבצע את האימות בין המשתמש לAP
* RADIUS – זהו שרת שנועד לקבוע את צורת האימות שהAS יעבוד.

תיאור הפתרון המוצע בשלב הראשוני:

* פיתוח שרת RADIUS תוכנתי לצורך חיקוי תהליך ה4 way handshake ובהתאם נוכל לקבוע איזה אימות אנו נבצע.
* בניית אפליקציה שתתממשק עם הנתב הביתי לצורך בקרה ופיקוח על הרשת
* התאמה עבור כל רשת ביתית.

תיאור הכלים המשמשים לפתרון:

* נתב עם יכולת להתממשק עם שרת RADIUS. (אנו ניקח נתב בייתי מדגם D-link 615).
* מחשב עם מערכת Linux לצורך פיתוח שרת הRADIUS.
* מחשב שיבצע טסטים על תהליך האימות וההתחברות לנתב.

טבלת סיכונים

|  |  |  |  |
| --- | --- | --- | --- |
|  | הסיכון | חומרה | מענה אפשרי |
| 1 | אי עמידה בזמנים | 5/5 | הוספת שעות נוספות |
| 2 | תלות בשותף | 3/5 | ניסיון לקבוע לימים קבועים במשך השבוע ולנסות לפצל את העבודה |
| 3 | אי סנכרון בין שרת הRADIUS לAP | 2/5 | מחקר על שרתי RADIUS בחברות גדולות |
| 4 | חוסר ידע בתקשורת בתקן IEEE 802.11 | 2/5 | קריאה מרובה על רשתות WIFI בתקן IEEE 802.11 |
| 5 | פתרון לא יעיל שעלול לגרום לעיכוב בהגשת הפרויקט | 2/5 | לנסות לחפש פתרון שהומצא ולהתאימו לפרויקט ולא להמציא את הגלגל |
| 6 | פגיעה בחומרה של הנתב לאחר צריבת קוד | 3/5 | לבצע המון טסטים ולקרוא חומר על צריבה נכונה לנתב |
| 7 | שאיבת הנתונים לאפליקציה | 2/5 | מחקר על אפליקציות ששואבות נתונים משרתים |
| 8 | חוסר ידע באנדרואיד | 2/5 | ללמוד אנדרואיד בלי שום קשר לקורס במכללה |
| 9 | עיכוב בפיתוח של השרת RADIUS לצורך סנכרון המידע באפליקציה | 1/5 | לנסות להתחיל להקים את השרת בשלבים המוקדמים של הפרויקט |
| 10 | חוסר ניסיון בצריבה ובתכנות על חומרה | 1/5 | קריאה בפורומים עוד בשלב מוקדם של הפרויקט |

רשימת ספרות:

<http://etutorials.org/Networking/>

ספר שמנתח את ההיסטוריה של תקשורת wifi עד היום.

<https://en.wikipedia.org/wiki/Brute-force_attack>

הסבר על מתקפות brute fouce.

<http://www.tp-link.com/en/FAQ-500.html>

הבדלים בין רשת WPA-Personal לבין WPA-Enterprise.

<https://en.wikipedia.org/wiki/Wi-Fi_Protected_Access>

הסבר על פרוטוקול WPA.

<https://mrncciew.com/2014/08/19/cwsp-4-way-handshake/>

מסביר על תהליך 4 way handshake.

תכנון הפרויקט

|  |  |
| --- | --- |
| 8.8.16 | שיחת פתיחה על הפרויקט – פיתוח הרעיון |
| 1.10.16 | התחלת ביצוע מחקר על פרוטוקול EAPOL, מציאת חולשת ה4 way handshake עבור WPA-Personal. |
| 31.10.16 | הגשת הרעיון מחדש לאחר מחקר. |
| 14.11.16 | סיכום מאמר והמשך שיחה על רעיון הפרויקט |
| 4.12.16 | הגשת הצעה |
|  | שיחה על ההצעה |
|  | הגשת אב טיפוס |
|  | שיחה על האב טיפוס |
|  | בניית מצגת להצגה החולשות והפתרונות לרעיון |
|  | הקמת שרת RADIUS |
|  | הקמת אפליקציה |
|  | בדיקות תוכנה |
|  | מסירה |
|  | העברה |

|  |  |
| --- | --- |
|  | תיאור |
| 1 | מחקר על Security wireless:   1. מהם סוגי האבטחה שקיימים 2. חולשות וחוזקות של כל אחד מהם 3. הבנת החולשות ומציאת פתרונות |
| 2 | פיתוח שרת RADIUS   1. לדמות את תהליך ה4 way handshake 2. לחקור בארגונים גדולים איך הם עובדים ולנסות לשלב את זה ברשתות קטנות |
| 3 | פיתוח אפליקציה   1. לפתח בקרה וניהול רשתות wife מרחוק 2. לפתח מערכת עדכונים ששולחת הודעות log |
| 4 | שילוב שרת פיתוח עם האפליקציה   1. סנכרון של המידע, ניתוח והעברת החומר לידי המשתמש |
| 5 | הרצות חיות ועשיית מבדקים לכל אורך הפרויקט   1. בדיקת יעילות – האם המערכת לא מעטה את תעבורת הרשת. 2. בדיקת ההצפנה – האם אלגוריתם ההצפנה חזק או שצריך להשתמש ביותר ביטים/אלגוריתם אחר (RC4) 3. בדיקת שליחת הנתונים לאפליקציה 4. האם ניתן עדיין לבצע את המתקפה אחרי פיתוח השרת. |

נספחים:

