**המחלקה להנדסת תוכנה**

**פרויקט גמר תשע"ז**

**אבטחת נתבים ברשת ביתית(fm)**

**Securing wireless networks for home routers**

**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**](https://github.com/sh00ki/JCE-Project) |
| **2** | **יומן** | <https://trello.com/b/yFEnQM5i/final-project> |
| **3** | **ניהול פרוייקטים** | **Github** |
| **4** | **סרטון אב-טיפוס** | <https://drive.google.com/file/d/0B2j2ru9fqqVhYVdobmp3TENxV3M/view?usp=sharing> |

**תקציר**

*רקע כללי:*

בעידן של היום כל רשת ביתית המבוססת Wi-Fi בתקן 802.1X חשופה למתקפת brute force לצורך פיענוח הסיסמה לרשת.

ברגע שיש לנו את הסיסמה לרשת, ניתן לבצע האזנות, סריקות, מניעת שירות ופריצה לכל המחשבים ברשת וזאת בגלל שנתבים בייתים מבצעים תהליך אימות של סיסמה בלבד ללא פרמטרים נוספים.

לגורם ג' שאין לו גישה לרשת אבל השיג את הגישה ע"י האזנה וניתוח המידע שעובר ברשת.(מפורט בפרויקט איך ניתן לעשות את זאת) יש לו היום מספיק כלים שבמידה והוא יאומת ע"י הנתב להיכנס לרשת (בעזרת סיסמה) הוא יכול לדעת כל מה שעובר בתוך הרשת ואנו חשופים למידע מאוד פגיע.

כיום נתבים בייתים אינם יכולים לנתח, לנטר ולהבדיל בין משתמשים מורשים שיש להם סיסמה לרשת לבין גורם ג' שיש לו את הסיסמה לרשת.

*ניתוח הבעיה* : הנתב הביתי מאפשר את הכניסה לרשת בעזרת סיסמה נכונה בלבד.

*פתרון:*

שילוב שרת RADIUS - Remote Authentication Dial In User Service פרטי עבור כל רשת ביתית לצורך פיקוח האימות ע"י גורם מורשה שמכיר את הרשת. וזה בגלל שכיום הנתבים לא יכולים להבדיל בין פורץ שיש לו את הסיסמה לרשת לבין משתמש רגיל שיש לו את הסיסמה לרשת.

ולכן בפרויקט זה אנסה להתמקד בכך שאפילו אם לפורץ יהיה סיסמה לרשת, כל משתמשי הרשת יידעו על כך ובנוסף תהיה אפשרות לא לאפשר לו כניסה לרשת למרות שיש לו את הסיסמה לרשת. (מפורט מסמך אפיון בפרויקט).

הגדרת הפרויקט: (החלקים מתבצעים במקביל)

1. לימוד והבנה של פרוטוקולי הSecurity והאימות ברשת אלחוטית ביתית וארגונית **(סטטוס בוצע)**
2. הבנת מנגנונים לשיבוץ קוד בנתב **(מוסבר ומוצג בסרטון, סטטוס התחלה)**
3. כתיבת image עבור שרת radius **(סטטוס התחלה, מאוד מאתגר)**
4. צריבה של הקוד בנתב והפצה ברשת לצורך קבלת ביקורת. (**סטטוס לא ניתן להתחיל** כי יש תלות ב1,2,3)

*הערה :*

מומלץ לצפות בסרטון **לפני** קריאת המסמך לצורך קבלת רקע כללי על הפרויקט

**תוכן עניינים**

תוכן עניינים

[מבוא 5](#_Toc471744862)

[- אימות 6](#_Toc471744863)

[- סינון 7](#_Toc471744864)

[- הצפנה 7](#_Toc471744865)

[Wep – Wired Eqivalent Privacy - 8](#_Toc471744866)

[- מהו WPA – Wi-Fi Protected Access 9](#_Toc471744867)

[- מהו WPA2 – Wi-Fi Protected Access 10](#_Toc471744868)

[- תהליך ה 4 way handshake 11](#_Toc471744869)

[- מילון מונחים, סימנים וקיצורים 13](#_Toc471744870)

[תיאור הבעיה: 14](#_Toc471744871)

[- Personal 15](#_Toc471744872)

[- Enterprise 15](#_Toc471744873)

[דרישות ואפיון הבעיה: 15](#_Toc471744874)

[הבעיה מבחינת הנדסת תוכנה: 16](#_Toc471744875)

[תיאור הפתרון: 16](#_Toc471744876)

[תרשים זרימה כללי של האלגוריתם (עוד עתיד להשתנות): 18](#_Toc471744878)

[תיאור הפתרון המוצע: 18](#_Toc471744880)

[שלבים העבודה במהלך הפרויקט: 19](#_Toc471744881)

[תהליכים ונתונים המערכת: 19](#_Toc471744882)

[תיאור הכלים המשמשים לפתרון: 19](#_Toc471744883)

[תיאור ואפיון הפתרון עבור הבדלה בין המשתמשים שמחוברים לרשת: 19](#_Toc471744884)

[תכנית בדיקות 20](#_Toc471744885)

[טבלת סיכונים 21](#_Toc471744886)

[רשימת ספרות: 22](#_Toc471744887)

[תכנון הפרויקט 23](#_Toc471744888)

[טבלת דרישות 24](#_Toc471744889)

[נספחים: 25](#_Toc471744890)

## מבוא

היום, השימוש ברשתות אלחוטיות (WIFI) הופך להיות בסיסי לכל משתמש, 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 ביתית או משרדית שאינה מאובטחת כיאות לעולם הגדול, חושפת את הנתב ומשתמשי הרשת הביתית עצמה, האקרים 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.

ברצוני 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 שיישם במהלך הפרויקט. עם זאת ובהנחה שעל המחשבים לא שמורים סודות חשובים ביותר, יישום האבטחה שאציג בפרויקט, יגביר את בטחון הרשת האלחוטית בבית ובמשרדים קטנים.

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

**IEEE 802.11** זהו אוסף סטנדרטים (פרוטוקולים) הנושאים אינפורמציה ותקנים אשר ממכרזים בצורה פורמלית כיצד על המערכות ברשת המקומית האלחוטית בתדרים 2.4Ghz, 5Ghz לפעול.

תקן זה מכיל פרוטוקול מסודר שעל פיו יצרני רכיבי התקשורת עובדים. דוגמה לאחד מן ההוראות בסטנדרט הינו דרך האפנון לצורך העברת המידע ברשת האלחוטית ושימוש בפרוטוקול בסיסי שמשמש לרשת מקומית (LAN).

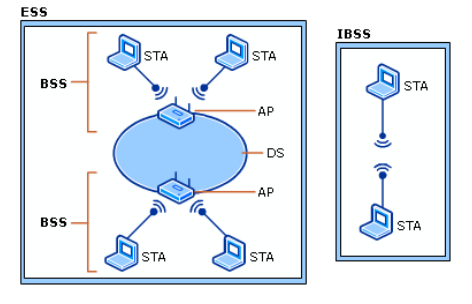
המבנה הבסיסי של הסטנדרט 802.11 מכיל מספר מרכיבים עיקרים: תחנה (STA), נקודת גישה אלחוטית (AP), מערך שירות בסיסי עצמאי (IBSS), מערך שירות בסיסי (BSS), מערכת הפצה (DS), ומערכת מורחבת(ESS).

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 בלמעלה מ- STA אחד. רשת זו נקראת רשת אלחוטית בסיסית.

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

דוגמה לסטנדרט אפשר לראות באיור הבא:



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

עקב חוסר פרטיות ברשת האלחוטית, נצטרך לספק אמצעי אבטחה שמחולקים לשלושה חלקים:

- אימות – כל STA מחויב להזדהות. על התחנה ליצור שייכות עם הרכיב AP עוד לפני שהוא משדר מידע. השייכות מתבצעת באמצעות שני קריטריונים של אימות:

- אימות במערכת פתוחה – אימות במערכת זאת (AP) אינה דורשת סיסמה וכל התחנות יכולים להתחבר למערכת.

- אימות באמצעות מפתח משותף – זאת מערכת AP שמחייבת מפתח 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.

- 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.

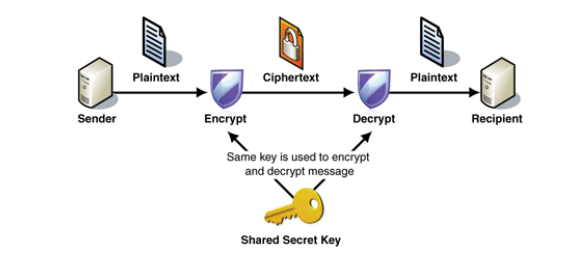
- הצפנה – זהו תהליך שהAP הופך מסך מקורי למספר מוצפן.

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

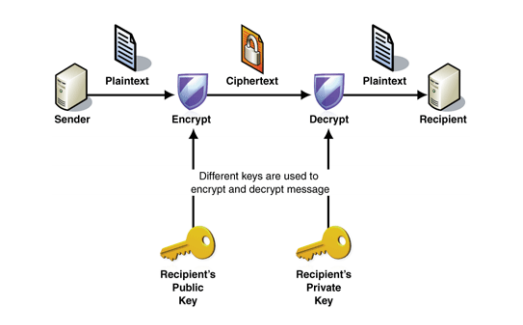
ההצפנה מתבצעת ע"י העברת המידע דרך נוסחת הצפנה מסוימת (מפתח).

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)



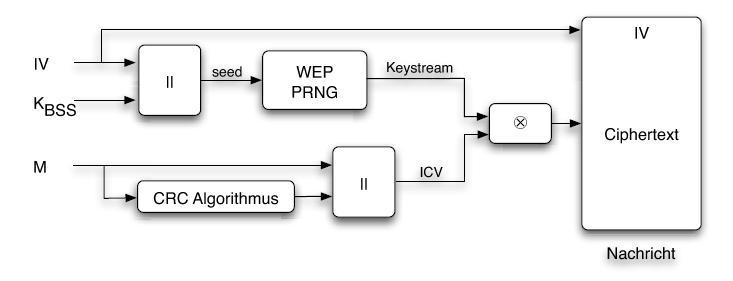
Wep – Wired Eqivalent Privacy - – הוא אלגוריתם לאבטחת רשת נתונים אלחוטית העומדת בתקן IEEE 802.11. מנגנון אשר מאפשר הצפנה סימטרית. WEP משתמש במפתחות הצפנה באורכים של 40 bit או 104 bit. דרך פעולתו של האלגוריתם היא: הצפנת זרם סימטרי אשר לוקח את גוף הframe אשר מכיל את המידע הרצוי לשליחה. מעביר אותו דרך הצפנה 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.

## מהו WPA2 – Wi-Fi Protected Access

עד לשנת 2006 הפרוטוקול הרשמי לאבטחת נתונים ברשת האלחוטית היה WPA. לאחר שנת 2006 הוחלף באופן רשמי הפרוטוקול על ידי הפרוטוקול WPA2. אחד השינויים המרכזיים בין הפרוטוקולים הוא הבאה לשימוש של האלגוריתם AES, תוך שימוש גם ב-CCMP במקום TKIP. נכון לעכשיו, לא ידוע עד כמה פגיעה מערכת האבטחה של הפרוטוקול. על מנת שתהייה חשופה לפגיעה היא דורשת מהתוקף בדיעבד להיות כבר מאושר בתוך הרשת ושם לחולל את זדונו ולגנוב את המפתחות האחראים לחלק הטכני של ההצפנה בפרוטוקול. משם הדרך לתוקף קלה יותר לבצע זדון לשאר רכיבי התקשורת במערכת. ההשלכות של השיפורים במעבר הפרוטוקולים גורמים לכך שבwpa2 – enterprise חדירות האבטחה מצטמצמת עדי כדי אי יכולת חדירה לרשת האלחוטית. ברשת ביתית הפרוטוקול מכסה את מרבית החדירות למערכת התקשורת האלחוטית. למרבה הצער, גם כאשר שופר הפרוטוקול, אותם גורמי בעיות חדירה אשר מהווים נקודה רעה לWPA לא פוסחים על הפרוטוקול הנ"ל וכן גם WPA2 נושא את נקודות חולשה דומות לWPA. מתקפת הווקטור דרך הWPS מהווה איום ממשי על הפרוטוקול החדש.

EAPOL - Extensible Authentication Protocol (EAP) over LAN

הסטנדרט IEEE 802.1X מגדיר דרך להעברת הודעות עם הצפנת EAP דרך הרשת האלחוטית. הפרוטוקול המוגדר על הוא EAPOL. המטרה היא לעטוף את הפרוטוקול EAP, זאת עבודתו של EAPOL. לא כל ההודעות המועברות על ידי EAPOL נושאות את ההצפנה EAP. חלקן משמשים לביצוע פעולות אדמניסטרטיביות. חמשת סוגי ההודעות של EAPOL הינן:

* EAPOL-Start
* EAPOL-Key
* EAPOL-Packet
* EAPOL-Logoff
* EAPOL-Encapsulated-ASF-Alert

אפרט על ארבעת סוגי ההודעות הראשונות ועל החמישית אפסח זאת בשל חוסר רלבנטיות לנושא אבטחת מידע.

EAPOL-Start

כאשר משתמש קצה רוצה להתחבר אל הLAN הוא אינו יודע את כתובת הMAC של השרת האימות. למעשה איננו יודע מי הוא השרת או קיים שרת שכזה. על מנת לפתור את הבעיה הנ"ל, הוגדר בסטנדרט IEEE 802.1X דרך פתרון לבעיית הקישור בין המשתמש לשרת והפתרון הינו הודעת START. הודעה זו נשלחת בצורת שליחה לכתובות רבות בצורה מקבילית (מולטי קאסט) לכל שרתי האימות הנמצאים באזור המוכרים בסטנדרט על מנת למצוא שרת ולהודיע לו שהמשתמש (תחנה) מוכן לשידור.

EAPOL-Key

על ידי שימוש בהודעה זו, השרת שולח מפתחות מוצפנים למשתמש על מנת שיוכלו לדבר ביניהם בצורה מוצפנת ולהכיר בזה שהמשתמש קיים ברשת.

EAPOL-Packet

הודעות אלו נועדו לעטוף את המידע ואת הפרוטוקול EAP ושלח את ההודעה. בצורה פשוטה זו מעין קופסה שבתוכה נמצא פרוטוקול הEAP והודעה זו נועדה לשלח את הקופסה בין הצדדים.

EAPOL-Logoff

הודעה זו מציינת כי המשתמש רוצה לסיים את החיבור לרשת ולהתנתק ממנה.

- 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

כעת אסביר את התהליך כולל העברת המפתחות בין הרכיבים המשתתפים ברשת:

רכיב הAP מוודא (מבצע אימות) שהSTA הוא אכן גורם אמין שמנסה להתחבר לרשת, תוך החזקת מפתח שנקרא Per Master Key(PMK) שאותו הוא יוצר מחדש ומחלק בצורה חד חד ערכית את המפתח עבור כל STA.

בשלב הראשון רכיב הAP שולח את המפתח הPMK לSTA (נכלל בהודעה הראשונה). כאשר הSTA מקבל את ההודעה מהAP הוא יוצר הודעה חדשה שבה בין היתר הוא גוזר מפתח הנקרא Pairwise Transient Key (PTK) והיא מכילה בנוסף 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 הכלולים בהודעה, הPTK מחזיק 5 מפתחות שונים שנשתמש בהם בהמשך (64bytes).

5 המפתחות הינם:

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

את 5 המפתחות הנ"ל, הPTK שולח עם כל המידע הזה לAP (הודעה 2 ).

במידה וכל המידע של הודעה 2 אכן נכון (אומת ע"י הAP), נמשיך בתהליך הלחיצה להודעה 3. אחרת רכיב הAP יפסיק את התקשורת בין הSTA לAP.

נדבר כרגע על המצב בו ה-AP מקבל את ההודעה השנייה מהSTA. הוא מבצע אימות של הדברים הבאים:

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

בעת שכל המידע הנבדק אכן נמצא אמין, הAP מתחיל לייצר 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.

## https://mrncciew.files.wordpress.com/2014/08/cwsp-4-way-07.pngמילון מונחים, סימנים וקיצורים

**MSK – Master Session** **Key** - מפתח זה נועד בשביל לקיים משא ומתן במשותף בין ה Supplicant לבין הAuthenticator Server.

מפתח זה מועבר באמצעות ערוץ מאובטח מה Authenticator Server לבין הSupplicant.

**PMK- Pairwise Master Key** - הPMK יכול להיגזר מפרוטוקול EAP או ישר להיגזר ע"י הPSK (Preshared Key).

ה PMK הוא ה256 ביטים הראשונים של הMSK.

**GMK – Group Master Key**  - הינו רנדומלי ונוצר ע"י הAuthenticator.

**PTK – Pairwise Transient Key**  - הינו ערך שנגזר מהPMK ,Authenticator nonce- Supplicant\_nonce, Authenticator Address, Supplicant Address.

זה למעשה משמש אותנו בשביל להצפין את כל השידורים ליעד בודד בין הclient לAP .

PTK – מכיל 5 מפתחות שונים. (הסבר על המפתחות הוסבר במעלה הדף)

* **KCK – Key Confirmation Key**
* **KEK – Key Encryption Key**
* **Temporal Key**
* **Temporal MIC**-1
* **Temporal MIC**-2

**GTK – Group Temporal Key** - משמש אותנו להצפין את כל המידע (Broadcast/Multicast) בין הAP לclients.

GTK נגזר מהAuthenticator ונשלח לSupplicant, כאשר הSupplicant מקבל את הGTK הוא שולח תגובה לAP ואומר לו שהוא קיבל את הGTK ויישם אותם ומוכן להקמת תקשורת מוצפנת.

## תיאור הבעיה:

בשני הפרוטוקולים (WEP,WPA) נשארה דרך אחת פריצה להיכנס לרשת שלא נפתרה ע"י IEEE 802.11.

הבעיה המרכזית שעלי לפתור בפרויקט זה היא מתקפת brute force ע"י האזנה לחבילות מידע של האימות ופיענוח של תהליך האימות.

אתחיל להבהיר מה היא המתקפה:

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.

אסביר את אותה נקודת חולשה של הפרוטוקולים הקיימים היום:

WPA (וגם WEP) משאירים אותי עם הבעיה המרכזית והיא שניתן להאזין למידע ולתעבורה ברשת התקשורת ע"י כלים חינמיים ברשת ולדגום את כל המידע שעובר ברשת. למעשה, לא אוכל לדעת יותר מדי על התעבורה, עד שלא אפענח את המפתח של המידע שעובר באותה רשת שברצוני להאזין לה. ולכן כל מה שנשאר לעשות הוא לדגום את פקטות האימות ולפענח את המפתח, כך שאוכל להבין ולהאזין לשאר המידע שעובר ברשת ולבצע ברשת פעולות נוספות.

לא אוכל למנוע מהאקר (או גורם אחר) להאזין לרשת או לאסור עליו לדגום את פקטות האימות, אך כן אנסה להוסיף/לשנות את דרך האימות ובכך למנוע מהאקר להיכנס מלכתחילה לרשת.

בפרויקט זה ארצה לאמוד את הבעיה ולנסות לפתור אותה עבור רשת ביתית או משרדית קטנה ולנסות למנוע מתקפה מסוג brute force עבורן.

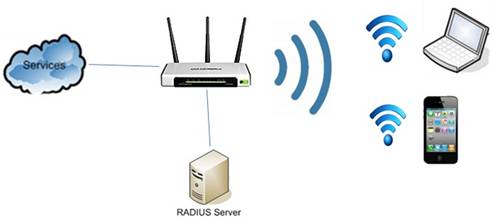
ברצוני להבדיל בין שני סוגים של רשתות נתונים:

* 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 זו, ניתן לבנות דרך אימות חדשה/שונה. אציין שרשת זאת מסובכת מאוד להקמה ומצריכה המון ידע על רשתות.



## דרישות ואפיון הבעיה:

נתבונן כעט מנקודת המבט של המשתמש אשר ישתמש במערכת האבטחה.

הדרישה שלי מהפיתוח והמחקר היא *מניעת האפשרות לבצע מתקפות מסוג brute force* עבור רשתות קטנות. זה ידרוש ממני להתמודד עם האתגרים הבאים:

* להתבסס על התקן IEEE 802.11. זאת מפני שרשתות ביתיות או משרדיות קטנות מתבססות ועובדות על תקן זה.
* התוכנה שאכתוב, תצטרך להיות יעילה וקלה לשימוש עבור כל משתמש.
* התוכנה צריכה להיות בעלת יכולת אוטומטית, יכולת הפעלה ובעלת יכולת לזהות ניסיון חדירה.
* אני רוצה לאפשר יכולת בקרה מרחוק על מצב הרשת ובמידת הצורך גם הכנסת גורם אנושי למערכת אשר יבצע את האימות שעליו אפרט בהמשך.

## הבעיה מבחינת הנדסת תוכנה:

בפרויקט זה אעבוד מול תקן IEEE 802.1X, אשר אוסף החוקים בסטנדרט שלו צרובים בכל נתב בייתי/משרדי ולכן אצטרך לסנכרן את עבודתי עם סביבת העבודה בעלת התקן הזה. לכן אתכנת בסביבת עבודה Linux ואעבוד עם שפת C.

ישנן מספר בעיות תכנותיות שברצוני לפתור בפרויקט:

1. שינוי של דרך האימות בו עובד היום רכיב ה - AP בשביל למנוע את דגימת פקטות האימות של הEAPOL.
2. שאיבת ושליחת נתונים בזמן אמת מה-AP לצורך זיהוי וניתור ניסיונות חדירה לרשת.
3. מניעת פגיעה במשתמשים אשר מורשים גישה בעוד מועד ובהסכמה ל-AP, אך אינם יכולים להתחבר כתוצאה משילוב הפיתוח שלי. אצטרך לפתח מערכת לומדת אשר מזהה ומכירה כל משתמש שמתחבר אליה. קיימים מספר פרוטוקולים שניתן להיעזר בהם (DHCP,NAT,VLAN) ולהשתמש בהם בפרויקט.
4. המערכת תהיה צריכה להיות קלה כמה שיותר ויעילה שלא תעכב את תעבורת הרשת כתוצאה מקוד לא חכם. המערכת בנוסף צריכה להיות קלה להתקנה ושקופה למשתמש.

## תיאור הפתרון:

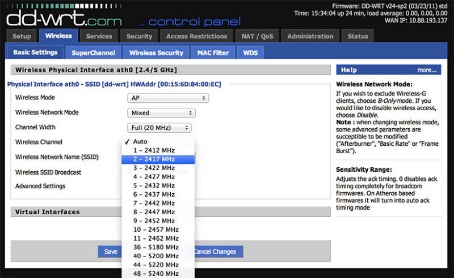
התרשים הבא מתאר את ארכיטקטורת המערכת:



## Router – Dlink-615 – הנתב שנצרב בתוכו dd-wrt ומאפשר גישה לשרת RADIUS לצורך אימות ובקרה על משתמשים. יש לו יציאה לרשת WAN לצורך בקרה מרחוק על הרשת והתחברות לאפליקציה (עוד לא הוחלט דרך איזה פורט נקבל ונמשוך מידע בשת"פ עם ליאור ספיר).

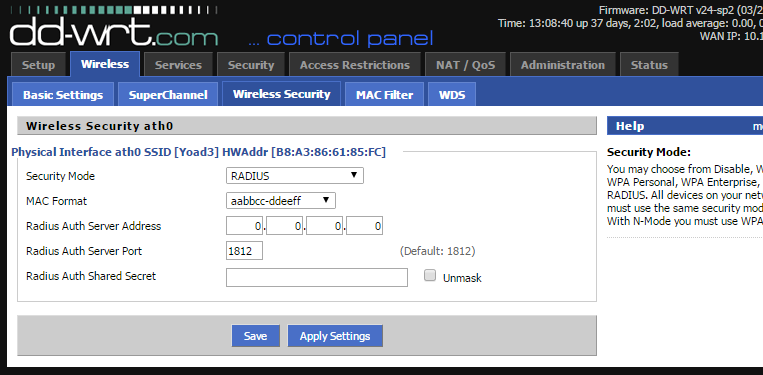
בנוסף יש לו משתמשים שמחוברים אליו (משתמשים מורשים ומשתמשים זמניים).

תמונה של הנתב מימין ותמונה של הממשק של הנתב עם הdd-wrt משמאל:



* RADIUS – השרת שאנו מפתחים לצורך בקרה ואימות משתמשים.

השרת מקבל ומושך מידע דרך פרוטקול EAPOL ע"י פורט 1812 מהנתב. מתואר בתמונה הבאה ההגדרות של ההתקשרות עם הRADIUS דרך הנתב.



* Database – מסד הנתונים שיחזיק את פרטי המשתמשים המורשים.

בשלב הראשוני זיהוי המשתמשים דרך הDatabase יזוהה ע"י כתובת MAC. (בשלב מתקדם יותר דרך קוד PIN/אמצעי ביומטרי).

* Machine Learning – המערכת שתדע לזהות וללמוד משתמשים מורשים לרשת ומשתמשים שאינם מורשים. דוגמה : משתמש מורשה שאישר משתמש אחר שאינו מורשה ורוצה שהמערכת תכיר בו בתור משתמש קבוע. המערכת תדע לזהות את הכתובת MAC שלו ולאפשר לו גישה קבועה לרשת
* Log – כל התראה על ניסון התחברות/התנתקות יתועד וידווח למשתמשים המורשים דרך האפליקציה.
* Application – החלק השני של הפרויקט בשת"פ עם ליאור ספיר.
* Internet Gateway(ISP) – התקשרות לנתב הביתי כאשר אנו לא מחוברים אליו באופן ישיר כלומר התקשורת לנתב הביתי דרך הרשת WAN מכל מקום בעולם.
* Authorized User – משתמש שנמצא בDB של השרת Radius ומורשה גישה לרשת.
* Temporary users – משתמשים זמניים שמחוברים לרשת.

## תרשים זרימה כללי של האלגוריתם (עוד עתיד להשתנות):

## 

## תיאור הפתרון המוצע: (בשת"פ עם ליאור ספיר)

כאשר התחלנו לעבוד על הפרויקט, חקרנו לעומק את פרוטוקול EAP לצורך הבנת האימות ברשת LAN עבור נתבים בייתים המבוססים על רשת WIFI. לאחר המחקר וניסיונות רבים הוחלט שנפריד את הפרויקט לשני חלקים: החלק הראשון הוא בניית שרת RADIUS לצורך בקרה על הרשת והחלק השני הוא שליטה על הרשת מרחוק באמצעות אפליקציה.

הפרויקט התחיל כפרויקט אחד גדול, פוצל ונעשה הגדרת תפקידים עבור 2 פרויקטים שונים. אפרט על הצד שלי בפרויקט.

תחילה, ניסיתי להבין כיצד ניתן להגדיר את הנתב הביתי (Dlink615) כנתב שמתקשר לAS חיצוני (RADIUS) ולא משתמש בAS שלו (תהליך ה4 way handshake). לפי המחקר שעשיתי היום יש לכל נתב ביתי שעובד בתקן IEEE802 גישה לקנפג (configure) אותו לעבוד עם רשת Enterprise.

רשת Enterprise הינה רשת המבוססת WIFI(מבוססת תקן IEEE 802) שנועדה להתמודד עם כמות משתמשים גדולה (מפורט למעלה). רשת זאת משתמשת בשרת חיצוני (RADIUS) לצורך אימות המשתמשים שמנסים להתחבר לרשת.

מטרת הפרויקט היא שאבנה תהליך לאימות מחדש של ואטמיע אותו בנתב ביתי (ברמת הLow Level). הסיכון שעולה בכך שאין לנו מספיק ידע בצריבת חומרה לנתבים בייתים ויש חשש שאהרוס את החומרה של הנתב ולכן בשלב הראשוני אני אבנה שרת RADIUS תוכנתי שנמצא על מחשב ברשת שמחובר קווי עם IP (שחולק ע"י הDHCP בנתב) ופורט במצב listening (הוגדר כפורט 1812).

בשלב הבא, אתחיל לכתוב קוד בשפת C מכיוון שהחומרה שצרובה על הנתב (dd-wrt) מבוססת על שפת C ובעתיד יהיה יותר קל להטמיע את הקוד. השלב הראשוני בקוד יהיה בניית echo udp server מכיוון ששרת RADIUS מקבל מידע על בסיס פרוטוקול UDP.

בהמשך הקוד אנתח את המידע שמתקבל מהAP (הframe כאשר client מנסה להתחבר). במידה ומידע תקין אבנה challenge חדש עבור המשתמש לפי תיאור האפיון (מפורט בהמשך הפרויקט) ובכך אנסה לפתור את מתקפת הbrute force עבור נתבים ביתיים כלומר אפילו אם לגורם זר יש את הסיסמה לרשת הוא עדיין לא אוכל להתחבר כי שום גורם לא אישר אותו ברשת.

## שלבים העבודה במהלך הפרויקט:

1. בחירת נתב עם יכולת צריבת חומרה חדשה. **בוצע**
2. מחקר על דרך התקשרות עם שרתים חיצונים. **בוצע**
3. חיבור וקינפוג הנתב עבור התקשרות עם RADIUS שמבוסס תוכנה (בשלב הראשוני). **בוצע**
4. בניית echo udp server לצורך אישור שלב ג' - כלומר קבלת מידע ממשתמש שמחובר ברשת. **בוצע**
5. ניתוח ובקרה של מידע שמתקבל בradius server. (המידע שמגיע מוצפן).
6. בניית frame חדש לצורך האימות.
7. בניית בקרה חדשה עבור משתמש שמורשה ברשת – בשלב הראשוני תישלח הודעה בתוך הRADIUS server לצורך אימות המשתמש.
8. שליחת challenge עבור client שמנסה להתחבר לרשת.
9. בניית DB שמחזיק את הנתונים של המשתמשים המורשים ברשת.

## תהליכים ונתונים המערכת:

* המערכת תכלול מספר מצבי עבודה
  1. מצב listening – שרת הRADIUS תמיד יחכה לקריאות מclients לצורך חיבור לרשת WIFI.
  2. שידור – הAP תמיד ישלח אותות (beacons) לצורך זיהוי הAP בסביבת רשת שמבוססת WIFI.
  3. קבלה של מידע (recvfrom) – שרת הRADIUS יקבל את המידע ויתחיל לנתח אותו.
  4. שליחה של מידע (sendto) – שרת הRADIUS ישלח את המידע לclient.

## תיאור הכלים המשמשים לפתרון:

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

## תיאור ואפיון הפתרון עבור הבדלה בין המשתמשים שמחוברים לרשת:

* השאלה המרכזית היא איך נדע אם המשתמש שמנסה להתחבר הוא משתמש מוכר למערכת/משתמש לא מוכר למערכת/גורם זר
* משתמש מוכר למערכת – משתמש שכבר התחבר בעבר והAP מכיר בו.
* משתמש לא מוכר למערכת – משתמש שאינו התחבר בעבר וAP לא מכיר בו, אבל צריך לאפשר לו להיכנס לרשת. במקרה זה הפתרון המוצע הינו שמשתמש שמוכר למערכת יזהה את ההתחברות של המשתמש שלא מוכר למערכת ויאפשר לו גישה עם שני אופציות : התחברות תמידית למערכת והתחברות זמנית למערכת. כלומר ברגע שהמשתמש מתנתק מהמערכת הוא אינו יכול להתחבר לרשת אך ורק אם משתמש שמוכר למערכת יאשר אותו מחדש או שהמשתמש המוכר אישר אותו בתור מוכר למערכת ואז הוא לא יצטרך אישור מהמשתמשים שמחוברים לרשת.
* גורם זר – משתמש שלא מוכר ע"י המערכת ומנסה להתחבר, משתמשי הרשת יצליחו לראות אותו אבל לא יאפשרו לו להתחבר למערכת.
* רשת הRADIUS יזהה את הגדרת המשתמש ולפי כך הוא:
  1. אם המשתמש מוכר למערכת – יאפשר לו גישה.
  2. אם המשתמש אינו מוכר למערכת – השרת ישלח הודעה למשתמשים המורשים ברשת עם כל המידע הנחוץ.(בשלב הראשוני דרך האימות תהיה דרך השרת עצמו, בשלב יותר מתקדם תישלח הודעה ייעודית לאפליקציה) אחד ממשתמשי הרשת המורשים יצטרך לאשר את המשתמש שלא מוכר למערכת.

## תכנית בדיקות

לאורך פיתוח הפרויקט אבצע כמה וכמה בדיקות:

1. לאחר חיבור הAP לשרת הRADIUS אבדוק דרך רכיב שלא מחובר למערכת האם הוא יכול להתחבר, והאם יש פלט בשרת RADIUS של ההתחברות לתפקוד תקין של התקשורת ביניהם.
2. לאחר שמידע התקבל לשרת הRADIUS אתחיל לנתח אותו ולבדוק אם המידע שהתקבל תקין ע"י שימוש ב- rfc2865/rfc2866 (מידע זה מצורף ברשימת הספרות) למשל אם המידע שהתקבל גדול X וקטן מY.
3. ברגע שהמידע נותח ואומת אתחיל לבנות את הDB והבקרה עבור משתמש מורשה שנמצא ברשת – בשלב הראשוני תקפוץ הודעה להכניס קלט בשרת הRADIUS לצורך אימות ובשלב העתידי תקפוץ הודעה באפליקציה.
4. ברגע שהמידע יישלח לclient אדפיס את המידע ואבדוק אם הוא תקין לפני הגעת המידע לclient.

## טבלת סיכונים

|  |  |  |  |
| --- | --- | --- | --- |
|  | הסיכון | חומרה | מענה אפשרי |
| 1 | אי עמידה בזמנים | 5/5 | הוספת שעות נוספות |
| 2 | תלות בשותף | 1/5 | 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 ושליחת הנתונים לאפליקציה | 2/5 | מחקר על אפליקציות ששואבות נתונים משרתים |
| 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 |
| 11 | חוסר ידע בהצפנות | 3/5 | קריאה ומחקר של הצפנות בפרוטוקול EAP |
| 12 | חוסר בהכרת ספריות לשימוש בשרתים המבוססים EAPOL | 2/5 | קריאה ומחקר של פרויקטים דומים |
| 13 | חוסר ידע בניתוח מידע שמתקבל ברשת | 2/5 | קריאה על מודל OSI וקריאה של מחקרים בנושא |

## 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/>

הסבר על פרוטוקול WPA2

<http://www.howtogeek.com/167783/htg-explains-the-difference-between-wep-wpa-and-wpa2-wireless-encryption-and-why-it-matters>

הסבר על פרוטוקול EAPOL

<http://etutorials.org/Networking/802.11+security.+wi-fi+protected+access+and+802.11i/Part+II+The+Design+of+Wi-Fi+Security/Chapter+8.+Access+Control+IEEE+802.1X+EAP+and+RADIUS/EAPOL/>

מסביר על תהליך 4 way handshake.

<https://tools.ietf.org/html/rfc2865>

מסביר על rfc2865

<https://tools.ietf.org/html/rfc2866>

מסביר על rfc2866

<https://technet.microsoft.com/en-us/library/cc726017(v=ws.10).aspx>

RADIUS Protocol and Components

## 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 |
| 5.12.16 | קונפיגורציה של הAP |
| 6.12.16 | הקמת שרת RADIUS – התחלה מecho udp server עם פורט 1812 |
| 18.12.16 | חיבור הRADIUS לAP |
| 26.12.16 | שיחה ביניים – עדכון מצב הפרויקט |
| 5.1.17 | בניית בקרה לניתוח המידע בשרת |
| 23.1.17 | שיחה על האב טיפוס |
| 12.2.17 | הגשת אב טיפוס |
|  | בניית מצגת להצגה החולשות והפתרונות לרעיון |
|  | בניית בקרה לקבלת מידע בשרת |
|  |  |
|  | בניית בקרה לפיענוח והצפנה של המידע בשרת |
|  | בניית בקרה לזיהוי משתמש מורשה/לא מורשה בשרת |
|  | בניית בקרה לשליחת המידע בשרת |
|  | סנכרון עם האפליקציה |
|  | בדיקות תוכנה |
|  | מסירה |

## טבלת דרישות

|  |  |
| --- | --- |
|  | 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 כיצד הם עובדים ולנסות לשלב את זה ברשתות קטנות. |
| 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:

