זיהוי מכשיר סמארטפון בקרבת הרספברי. איך זה יעבוד?

# אופן הפעולה

* הזיהוי יתבצע באמצעות תשתית WIFI
* ה Raspberry והאפליקציה יחליפו בניהן הודעות דרך הענן, שבסיומן ידע הרספברי מי הם המכשירים הנמצאים בסביבתו, וכל מכשיר טלפון ידע מי הרספברי הקרוב אליו.
* נאמר על מכשיר שהוא *קרוב* לרספברי אם הוא נמצא באותה רשת LAN (ללא תלות במרחק הפיזי)

לפרוטוקול של זיהוי המכשירים בסביבה נקרא בשם **פרוטוקול MEP** (MAC Exchange Protocol). והוא יעבוד באופן הבא:

1. התקנת הרספברי תהיה כרוכה ברישומו בענן – כלומר בסיס הנתונים יכיל טבלה של כל מכשירי הרספברי שבשרתנו יחד עם כתובת הIP שלהם וכתובת ה-MAC שלהם.
   * *הערה*: יש לנו מכשיר רספברי אחד ולכן תהיה רק רשומה אחת בטבלה, אבל אנחנו נבנה את המערכת כך שתתאים גם לכמות גדולה של מכשירים.
2. כאשר משתמש מפעיל את האפליקציה, היא מיד תבקש מבסיס הנתונים את כתובות הIP של כל מכשירי הרספברי שתואמים את הsubnet שבו נמצא המכשיר, ואז והיא תשלח **הודעת MEP** לכל הכתובות ברשימה (הסבר נוסף בהמשך)
3. ע"ס הודעות הMEP הרספברי יוכל לרשום לעצמו בזיכרון את כתובות הMAC של המכשירים בסביבה
4. לאחר מכן הרספרי ישלח תשובה על כל הודעה שקיבל (במידת הצורך).
5. לאחר החלפת ההודעות, הן הטלפון והן הרספברי ידעו את כתובות הIP וה-MAC אחד של השני, וכמו כן הרספברי ידע בדיוק מי נמצא בסביבתו.

# מהי הודעת MEP?

* **MEP** - **M**AC **E**xchange **P**rotocol - זהו פרוטוקול החלפת כתובות MAC שאנחנו נפתח.
* הפרוטוקול יהיה מבוסס על JSON
* מבנה ההודעה:
* הסבר:
  + כל הודעת MEP מתחילה במחרוזת **$MEP**
  + **Dev Type** – סוג המכשיר. יכול לקבל רק שני ערכים: **RBPI** או **APP** (האפליקציה תשלח הודעות מסוג APP והרספברי ישלח הודעות מסוג RBPI)
  + **MAC Addr** – כתובת ה-MAC של המכשיר בצורת מספר הקסדצימלי של 12 ספרות וללא שום תווים נוספים!
  + **IP Addr** - כתובת ה-IP של המכשיר בצורת מספר הקסדצימלי של 8 ספרות וללא שום תווים נוספים!
  + **Callback** – פעולה שאנחנו מבקשים מהמכשיר המקבל את ההודעה לבצע מיד לאחר קריאת ההודעה. הפעולות האפשריות הן:
    - **RES** (קיצור של Response) – מבקש לשלוח בתגובה הודעת MEP לשולח
    - **ACK** (קיצור של Acknowledge) – מבקש לשלוח חיווי האם ההודעה הגיעה ליעד, ללא ציון כתובת MAC ו-IP (אפשר להעביר כתובת 0 בשניהם)
    - **NOP** (קיצור של No Operation) – אל תשלח שום הודעה בתגובה.
  + **Date** – תאריך ושעת שליחת ההודעה (עדיין לא לגמרי סגור על הפורמט)
* דוגמה:

{“$MEP”:

{

“DevType”:”RBPI”,

“MACAddr”:”123456789ABC”,

”IPAddr”:”C0A80101”,

”Callback”:”RES”,

“Date”: “28/07/2017 19:02”

}

}

# התחברות של מספר משתמשים במקביל

אם יש כמה משתמשים המחוברים לאפליקציה באותו זמן ובאותה רשת LAN אז הרספברי צריך לדעת מי מהם נשקל.

* המשתמש יצטרך ללחוץ על כפתור "התחל שקילה" רגע לפני עליה על המשקל.
* אם מישהו נשקל, והמערכת אינה מצליחה לזהות מי (למשל אם מישהו "שכח" ללחוץ על הכפתור) אז המערכת תסתכל על השקילות ב-X זמן האחרון ותנסה לנחש. זו הזדמנות טובה להתמיע אלגוריתם clustering כלשהו.
* במקרה הנ"ל האפליקציה תשאל את המשתמש "Are you Ramy Lerner?". אם המשתמש עונה בשלילה, אז התוצאה לא תישלח לענן ולא תוצג למשתמש.

# אבטחת הפרוטוקול

יש כל מיני בעיות אבטחה בשיטה הזאת לאיתור מכשירים.

אם ישאר זמן נטפל בהם כאשר הפרוייקט יהיה גמור. כרגע פחות קריטי.