# מיני פרויקט

# נושאים במערכות הגנה לרשת

## מגישים :


- ג'ואד סעיד
- אבראהים אבראהים

מנחה : מר דורון אופיק

# סקירת האפליקציה

## תיאור האפליקציה

**"SecureMail"** הינה אפליקציה בסביבת אנדרואיד שמטרתה להגן על המשתמש מפני אימיילים שמכילים ווירוסים או כל קובץ יפגע במשתמש.

האפליקציה משתמשת בבסיס נתונים שמכיל חתימות כך ש דרכם היא מזהה אם הקובץ שקיבלנו הוא קובץ זדוני או לא.

## מטרת האפליקציה

האפליקציה באה לתת מענה לנושאים הבאים:

- הגנת הלקוח והוספת עוד חומת אש
- אי-התלבטות בהורדת קבצים שמקבלים באימייל
- סריקה מהירה במקום אנטי-וירוס

## אופן שימוש באפליקציה

- המשתמש נכנס לאפליקציה באמצעות חשבון ג'ימייל שלו.
- המשתמש מקבל הודעה שימתין כמה דקות.
- בזמן שהמשתמש ממתין , האפליקציה טוענת כל החתימות.
- האפליקציה מתחברת לחשבון ג'ימייל , במידה והנתונים לא נכונים , המשתמש יקבל הודעה בכך וינסה עוד פעם.
- אחרי ההתחברות , האפליקציה עוברת על כל ההודעות שעוד לא קראנו , ובודקת אם מצורף קובץ.

- תוך כדי בדיקה , אנו מסמנים שכבר קראנו את ההודעה , ומורדים את הקובץ במידה וקיים.
- לאחר מכן , האפליקציה סורקת כל הקבצים שהורידה.
- במידה ומצאה שהוא קובץ חשוד , היא שולחת בקשה למחוק אותו ומודיעה למשתמש , אחרת היא משאירה אותו.
- ובסוף , המשתמש מקבל הודעה כמה הודעות נמחקו.

## אתגרים שנתקלנו בפרויקט

- **בחירת פרוטוקול :**

בהתחלה עבדנו עם pop3 משום שהוא הכי קל והכי מחיר , עד שנתקלנו בכמה בעיות :

1. אי-סנכרון , בהנחה ואנו משתמשים באותו אימייל בו זמנית , יתכן ונשלוף מידע שכבר בדקנו אותו מקודם
2. באפליקציה , כל אימייל שבדקנו , מסמנים אותו כ Read, אבל POP3לא תמך ב Flags .

ואז עברנו ל IMAP , והוא כמובן מסונכרן , וגם אפשר לנהל בעזרתו קבצים כמו Inbox , Trash , Sent mails .

- **בניית שפה לחתימות :**

זה היה האתגר הכי מעניין , לבנות סינטקס שבעזרתו נוכל לזהה קבצים חשודים ,וגם שהשפה תהיה נוחה לפתח בעזרתה חתימות.

**השפה בנויה משני מרכיבים :**

1. lss: משמש כמילת מפתיח לקובץ החשוד, כלומר במידה ו lss לא נתפס , אז לא ממשיכים למרכיב השני.
   lss הוא בנוי באופן הבא : האות הראשונה אומרת אם המידע שאנו רוצים לבדוק אמור להופיע בתחילת הקובץ או לא משנה איפה בקובץ , והאות האחרונה אומרת אם הנידע נמצא בסוף הקובץ או לא משנה. דוגמה :

Lss:@jawadwashere!;

נשתמש ב @ ! כדי שנוכל לדעת אם קיים או לא , אז הגדרנו ש @ אומרת כן ו ! אומרת לא , כלומר , אנו בודקים רק jawadwashere בתחילת הקובץ (offset 0).

יש גם אפשרות לכתוב בבסיס 16 , כלומר , החתימה הבאה שקולה לדוגמה :

**lss":"@\\x6a\\x61wad was here!**

2. Signature: פה נמצא כל הבשר , בשלב הזה אנו כבר יודעים ש lss נתפס , אחרת לא היינו מגיעים לשלב הזה.
ב signature יש שתי מילים שמורים
   I. Data : בעזרתה אנו יכולים לבדוק כל מילה אנו רוצים ב אסקיי או בבסיס 16.
   II. Length : בעזרתה אנו יכולים לבדוק אם גודל הקובץ שווה לגודל כל שהוא.

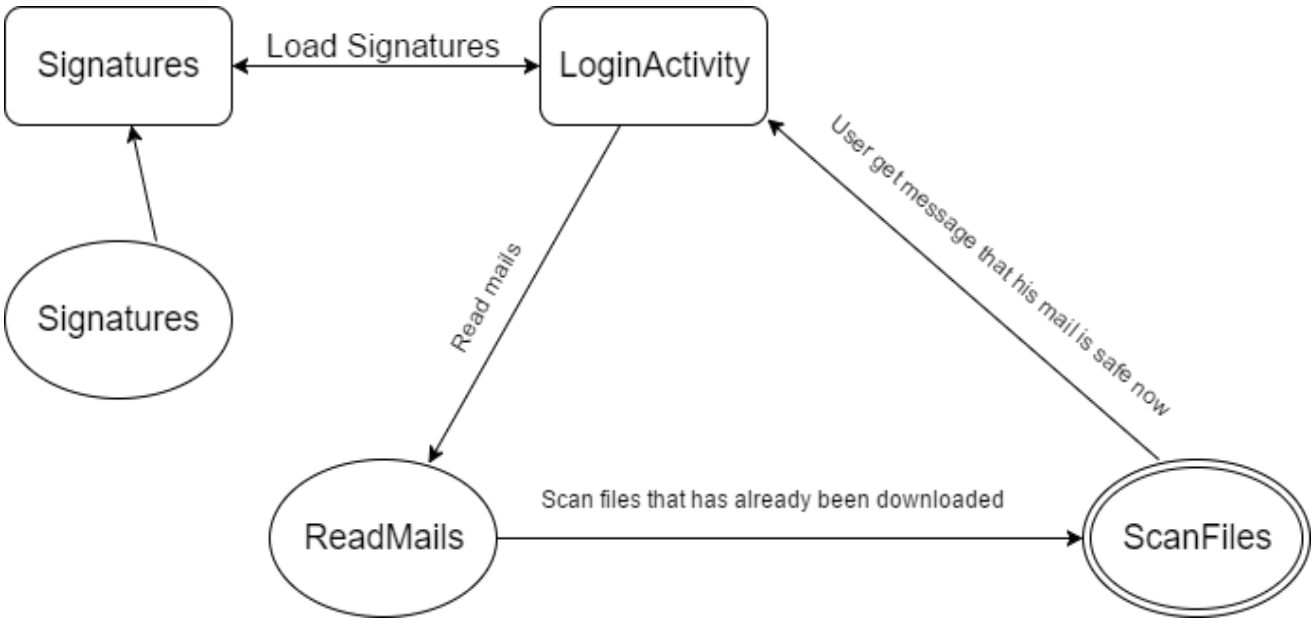
דוגמה :

**signature":"data:'\\x74fl\\x64';length:180;data:'takecare';**

- **Javamail**
  Api חדש בשבילינו , מעניין , לקח לנו זמן עד להבין איך בנוי , ואיך מפרסרים ההודעות.

- **טעינת קובץ החתימות :**
  החתימות כתובות בקובץ json , צריך לפרסר אותם ולאחסן אותם במבני נתונים כך שיהיה קל לעבור עליהם כדי להשוות.

# דיאגרמה


Signatures
Load Signatures
LoginActivity
Signatures
Read mails
User get message that his mail is safe now
ReadMails
Scan files that has already been downloaded
ScanFiles

## ביבליוגרפיה

- JavaMail : היא ספריית התקשורת שמספקת פלטפורמה עצמאית ללא תלות בפרוטוקול לבנות יישומים אלקטרוני והודעות.
https://javamail.java.net

- Gson : היא ספריה שמספקת כלי לפרסור קבצי json
https://github.com/google/gson

- jsonschema2pojo : אתר שבעזרתו בנינו קובץ json עם קוד java

## סיכום

- האפליקציה תעבוד על גרסאות מעל 4.0.3 .
- לצורך נוחות , יצרנו מייל לפרויקט שמוגדר היטב , במידה וברצונך להשתמש באימייל אחר , תידרש לתת הרשאות לאפיליקציה.

Email: bgu.test.cs@gmail.com
Password: jawad123

- בפרויקט יש קובץ בשם signatures.json תחת תיקיה בשם assets אפשר שם להוסיף חתימות.