בס"ד

אוניברסיטת
אריאל

מעבדת סייבר – התקפה

מטלת סיכום מעבדת התקפה תשפ"ג

<u>שמות מגישים</u>: רז אלבז - 207276775, אמיר ג'ילט -324942077

בס"ד

מטרות:

גניבת מידע:

- לאחר קבלת קובץ APK של אפליקציה תקינה יש להוסיף לאפליקציה זו תוכן זדוני.
- התוכן הזדוני יופעל כאשר המשתמש ילחץ על כפתור הrandom.
- מטרתו של התוכן הזדוני היא להשיג כמה שיותר מידע על המשתמש/המכשיר שיוריד את האפליקציה.
- לאחר הכנסת התוכן הזדוני על ידי repackaging, יש לוודא שאכן האפליקציה עומדת ביעד שלה ולא קורסת בשום שלב.
- המידע שנגנב ייוצא לקובץ בשם information.txt שישמר בתיקיה בה רצה האפליקציה.

תיאור מהלך העבודה:

ראשית, קיבלנו קובץ apk של אפליקציה בסיסית אשר מייצרת תאריך "קסם" על פי מספר פרמטרים שאותם מזין המשתמש או בצורה רנדומלית באמצעות כפתור ה-random שהיה מבצע את מה שהיה צריך במקור, ובנוסף גונב מידע על מכשירו של המשתמש / על המשתמש.

לאחר מכן, ניסינו בשלב הראשוני לחקור על איך בנוי הקוד עם reverse engineering על קובץ ה-apk בעזרת דיקומפליזציה של apktool. ראינו כי בהתבסס על "תרגום חופשי" שלנו עבור המתודות בקובץ ה- smali של MagicDate , הפעולות של הכפתורים נמצאים בפונקציה שיש בה switch case לid של הכפתורים וראינו שהמתודה של כפתור ה-random שנקראת getRandom() אשר עושה הוספות שונות של מספרים למערך ולאחר מכן מבצעת חישוב על תא רנדומלי במערך.

על כן, חשבנו להוסיף את הקוד הזדוני באותו case של ה-random ב-switch אחרי שהמתודה getRandom() תיקרא בכדי למנוע דיליי. הואיל וחשבנו שהוספת כל הקוד בswitch שגויה מבחינה הגיונית, השתמשנו בטכניקה שלמדנו במעבדת הseed שסופקה לנו בגוף המטלה. קראנו את המטלה והבנו שבתכנות באנדרואיד ניתן לשלוח broadcast לclass ייעודי שנקרא Receiver אשר מאזין לevent שאנחנו קובעים. בפונקציית ההאזנה השתלנו את הקוד שלוקח את המידע על מכשירו של המשתמש ומייצא לקובץ בתוך תיקיית הקבצים של האפליקציה. באופן זה, הייתה לנו את הגמישות להוסיף דברים במינימום זמן וסיבוך.

כפי שכתוב בגוף המטלה, הרבה יותר קל לקחת אפליקציה שכתובה בjava, להוסיף לה שורות, לעשות לה דיקומפליזציה ואז לקחת את הקוד (את המתודות) המתאים שכתוב בsmali ולהוסיפו לתיקיית קבצי ה-smali של האפליקציה שאנחנו רוצים לשנות עם שינוי ה-package path המתאים.
לכן, כך עבדנו: הוספנו את הקוד לאפליקציה שבנינו בעבר באחת מהרצאות הקורס, והוספנו לכפתור שעושה חישוב את השליחה של ה-broadcast ל-class שבנינו שמקבל את ה-event ובעצם עושה את כל התהליך של גניבת המידע והייצוא לקובץ שייכתב בתיקיית files של האפליקציה. בנוסף, הוספנו מחלקות נוספות שמטרתן לקחת על נושא מסוים כגון אינטרנט כמה שיותר פרטים. לקחנו את כל הclass-ים שכתבנו לאחר הדיקומפליזציה והעתקנו אותן לתיקיית ה-smali של אפליקציית magicDate. שינוי שכן היינו צריכים לבצע בצורה ידנית באפליקציית magicDate היה אתחול הרישום של ה-event של ה-broadcast וה-class שמקבל אותו, וכן שליחת ה-broadcast בסוף ה-case של ה-random. לאחר כל השינויים שביצענו עשינו build לתיקיית האפליקציה החדשה עם apktool וחתמנו את הקובץ apk לאחר השינויים.

לאחר מכן השתמשנו בadb:
השתמשנו בפקודה : adb devices בכדי להבין אילו מכשירים מחוברים ובעיקר בכדי לדעת מה השם המלא כולל הPort של הemulator המחובר לצורך התחברות, הדפסות, מציאת הקובץ וכו'.
השתמשנו בפקודה : adb -s emulator-5554 uninstall com.MagicDate כדי להסיר את אפליקציית ה-magicDate, לפני השינויים שביצענו, מהמכשיר המחובר ולאחר מכן, להתקינה באמצעות גרירת הapk למכשיר האנדרואיד ב-Android Studio.

וידאנו כמובן שאין קריסה של האפליקציה בשום שלב, ובדקנו את ההדפסות של הלוגים שביצענו לצורך דיבאגינג באמצעות הפקודה : adb -s emulator-5554 logcat אשר מראה את הlogcat של האימולטור בזמן ריצה.

עתה, נרצה לבדוק שהקובץ information.txt אכן נכתב כראוי ולייצאו. לכן, התחברנו לshell של אותו מכשיר באמצעות הפקודות :
adb connect localhost:5554 – פקודה אשר מתחברת לlocalhost עם הפורט של האימולטור.

adb -s emulator-5554 shell – פקודה מתחברת לshell של אותו אימולטור שצוין בפקודה.

su – פקודה שמחליפה לroot. היינו צריכים להשתמש בה על מנת לגשת עם cd לתיקיות כמו data או לחילופין לתיקיית האפליקציה.

מצאנו כי תיקיית האפליקציה נמצאת בנתיב : /data/data/ ותיקיית הקבצים של האפליקציה נמצאת בנתיב : data/data/com.MagicDate/files/ , ושם אכן ראינו את הקובץ information.txt שייצאנו אליו את המידע. כדי לייצא אותו למחשב חיצוני למכשיר השתמשנו ב-adb. היה צורך בלתת הרשאות root ל-adb באמצעות הפקודה : adb root , ולאחר מכן היה ניתן לייצא את הקובץ לDesktop של המחשב באמצעות הפקודה : adb pull /data/data/com.MagicDate/files/information.txt ~/Desktop , ואכן ראינו שהקובץ יוצא בהצלחה לשולחן העבודה של המחשב.

מבחינת ההרשאות שבהן השתמשנו כולן (כל שלוש ההרשאות) שייכות לאינטרנט בעוד שהמידע האחר שהשגנו, הושג ללא הרשאות כלל וכלל. מאחר ששמנו לב שיש ספריות שמוציאות מידע כזה או אחר על המכשיר ללא הרשאות, ניסינו לחשוב על מספר גבוה ככל האפשר של ספריות אשר מספקות sdk של מידע רב על אותו נושא שאותו ייעדנו לייצא כגון: מידע על מערכת ההפעלה ועד מידע על סנסורים קיימים. בהתאם לכך, ביצענו שינויים מתאימים בקובץ ה manifest של האפליקציה שאותה רצינו לשנות לאחר הדיקומפליזציה.