תקיפת XZ

השאלה :
איזה פעולות נקט התוקף - Jia Tan על מנת להקשות על גילוי התקיפה?

תשובה:
התוקף ביצע כמה מהלכים באספקטים שונים,

1. בניית אמון -
התוקף תרם באופן קבוע לפרויקט במשך יותר משנתיים והפך דה פקטו למתחזק העיקרי של הקוד, תחזוק הקוד על ידו היה עניין בשגרה שלא עורר חשד או צורך בבדיקה מיוחדת.

2. החבאת הקוד הזדוני מחוץ לקוד המקור של הפרויקט -
החלק הקריטי של הדלת האחורית הופעל רק לאחר ה – build-כלומר, **לא הופיע בקוד המקור בצורה ישירה**. התוקף השתמש בתקיית הטסטים שאליה הוא העלה קבצים בינרים אשר יצרו את הפרצה. הקוד הזדוני הופעל רק בשלב מאוחר בבנייה ובכך נמנע חשד בבדיקות קוד רגילות.
"
```
This injects an obfuscated script to be executed at the end of configure. This
script is fairly obfuscated and data from "test" .xz files in the repository.
```
"

3. הסתרת ההתקפה בקוד דחוס וקשה לניתוח-
חלקי הקוד הזדוני הועלו כקבצים בינרים בדחיסת XZ (LOL),
כך שהרצת בדיקות פשוטות על הקובץ ללא "פתיחת הדחיסה" אינן מאפשרות לזהות קטעים או מחרוזות חשודות.
```
../../tests/files/bad-3-corrupt_lzma2.xz
```

4. פיצול למספר קבצים ושילוב קוד תקין-
התוקף פיצל את התקיפה שלו לקבצים תמימים למראה ושתל את החלקים הזדוניים בין קוד לגיטימי מה שהקשה על מציאת החלקים הזדוניים.

5. שמירה על תקינות הקוד ויצירת מעטה לגיטימי לקבצים זדוניים -
התוקף דאג שהשימוש התקין לא יפגע והכל ירגיש כרגיל לשאר המתחזקים והמשתמשים. (מה שלא הצליח לגמרי כי ככה עלו על ההתקפה).
בנוסף, הוא טען בהערות של הפרויקט שהוא "שיפר" את אלגוריתם בשם CRC
כך שגם אם מישהו ימצא בין כל הקוד התקין את הזרקת הקוד הזדוני, הוא עשוי לחשוב שמדובר בקובץ O תקין בשל ה"שיפור" הצפוי באלגוריתם זה.
```
test -f liblzma_la-crc64-fast.o;
```

לסיכום, התוקף שילב הנדסה חברתית, הסתרת קוד בצורה מתוחכמת ושימוש במנגנונים הקיימים בצורה מבריקה.

מקורות:

- מצגת הקורס
- הפוסט המקורי של אנדרס (https://openwall.com/lists/oss-security/2024/03/29/4)