רשתות מחשבים בגישה מחקרית – מבחן מועד ב' תשפ"א מרץ 2021

ברק גונן

FTIP

ראשת שירות הבטחון תופפה בעצבנות באצבעותיה על השולחן. "סוכן שלנו, שנמצא במדינה זרה, חייב גישה לשרת שלנו. יש שם כמה קבצים שהוא חייב להוריד. לצערנו, המדינה הזרה הצליחה להגיע אל המחשב שלו ולהתקין שם תוכנת ריגול, כך שהם יכולים להאזין לכל התעבורה שלו. מה עושים?"

בחדר השתררה שתיקה עצבנית. "ממידע מודיעיני שיש לנו, תוכנת הריגול יכולה להאזין לכל מה שעובר מעל סוקטים כלשהם, בין אם TCP או UDP. נראה שהלך עליו" אמר מפקד המבצע.

לפתע נזכרתם במשהו שלמדתם פעם, מזמן. הרמתם יד ושאלתם בהיסוס "יש לסוכן אפשרות להתקין על המחשב סקאפי ופייתון?"

"כן" אמר מפקד המבצע "איך זה עוזר?"

מעודדים, הסברתם את התכנית: "אפשר לשלוח Raw data מעל IP, בלי לעשות שימוש ב-TCP או UDP. נמציא פרוטוקול משלנו שיודע לפענח מה עובר ב-Raw data, הסוכן יריץ את הלקוח ואנחנו נריץ את השרת. **מי שמפלטר את תעבורת ה-TCP / UDP של הסוכן לא ימצא כלום.** מה הסוכן צריך?"

"הסוכן צריך לקבל יכולת לבקש קובץ כלשהו שיישלח אליו. לדוגמה c:\cyber\secret.txt. כדי לא לעורר חשד, **כמות המידע המקסימלי שאפשר לשלוח מהשרת חזרה ללקוח היא 100 בתים בכל פעם, כך שאם הקובץ ארוך יהיה צורך לפרק אותו**"

"ואנחנו מניחים שפקטות עלולות ליפול בדרך?"

"כמובן"

"אין בעיה, **איישם מנגנון של שליחה אמינה, שיוודא שכל הפקטות הגיעו ליעד**"

ראשת שירות הבטחון סיכמה את הדיון "יש לך בדיוק **שעתיים וחצי** לסיים את השרת והלקוח. קדימה לעבודה, צאו לי מהחדר"

קובץ לקוח

יש לכתוב קובץ לקוח שמקבל קלט מהמשתמש, הנתיב ושם הקובץ המבוקש, שולח אותו אל השרת.

הקלט יתקבל באמצעות פרמטר לסקריפט, לא באמצעות input.

הלקוח לא צריך לבדוק את תקינות המידע שהמשתמש מזין.

הקלט חד פעמי, אין צורך לרוץ בלולאה.

הקובץ שהתקבל מהשרת יישמר במקום קבוע.

קובץ שרת

יש לכתוב קוד שרת שמקבל את הבקשה של הלקוח ושולח אל הלקוח את הקובץ המבוקש. קובץ הפרוטוקול יכלול את כל הפונקציות הנדרשות לחלוקה של הקובץ שנשלח לחלקים בגודל 100 בתים של מידע לכל היותר, ושליחה שלהם.

קובץ פרוטוקול

קובץ שהן השרת והן הלקוח ישתמשו בפונקציות שלו. הקובץ יכלול את כל הקבועים הנדרשים בשרת ובלקוח.

יש לכתוב פילטר מתאים, שאוסף רק את הפקטות של הפרוטוקול שלנו. שימו לב- אפשר להוסיף למידע שדות כרצונכם, או לשנות שדות קיימים בפרוטוקול IP. חשוב מאד לתעד את הפרוטוקול ולהסביר את פונקציית הפילטר. אין להשתמש בכתובות ה-IP של השרת והלקוח כחלק מהפילטר,קל וחומר לא בכתובות ה-MAC של השרת והלקוח.

הוספת מנגנון שליחה אמינה לקובץ הפרוטוקול

שימו לב – כדי לאלץ אובדן של פקטות, בשליחה אין לעשות שימוש ב-send, sendp או sr1. ניתן להשתמש רק בפונקציה special_send, שבאופן אקראי "זורקת" חלק מהפקטות. להלן הקוד של special_send, עליכם להוסיף אותו לקובץ הפרוטוקול:

```
import random

def special_send(packet):
    fail = random.randint(1, 10)
    if not (fail == 1):
        send(packet)
    else:
        print("Oops\n")
```

כעת, כיוון שהן השרת והן הלקוח לא יכולים לסמוך על כך שמידע שהם שולחים אכן יגיע ליעד, אתם נדרשים להוסיף פונקציונליות שתבטיח שהמידע אכן הגיע ליעד. אין צורך ליישם מנגנון שיבדוק אם חלו שגיאות במידע, רק לוודא שכל המידע הגיע תקין ובסדר הנכון. השתמשו בעקרונות של פרוטוקול אמין, כפי שלמדנו.

דגשים לפיתוח:

- אין להתבסס כלל על מודול Socket, שימוש בו יקבל ציון 0
- אין להתבסס על כתובת ה-IP של השרת או של הלקוח לטובת פילטור הפקטות. כתובת ה-IP של השרת תהיה 127.0.0.1 לטובת הפיתוח, אבל הפילטר לא יכול להשתמש בכתובת ה-IP של השרת כיוון שהשרת מקבל פקטות ממקורות רבים, והפילטר לא יכול להשתמש בכתובת ה-IP של הלקוח מכיוון שהיא איננה ידועה מראש לשרת.
- קל וחומר אין להתבסס בשום מקום על כתובות ה-MAC של השרת והלקוח, הקוד חייב לרוץ היטב גם על מחשבים אחרים.
- על השרת להחזיר תשובה לכתובת ה-IP שפנתה אליו, לא לכתובת קבועה

חלוקת הנקודות:

1. כתיבת לקוח – 20 נקודות.
2. כתיבת שרת – 20 נקודות.
3. קובץ פרוטוקול – 35 נקודות.
4. השרת לא יקרוס על שום קלט שהלקוח ישלח לו. 15 נקודות
5. הקוד יעשה שימוש בקבועים, ללא הערות PEP8, חלוקה הגיונית לפונקציות, הפרוטוקול והפילטר מתועדים היטב. 10 נקודות