

נושאים מערכות הגנה לרשות - תוכנית עבודה

מגישים: ברק שליט 316222280 ונועה דוד 207465634

מטרת הפרויקט

בכוונתנו ליצור אפליקציה service שמטרתו הגנת המשתמש מפני התקפות מסוג DNS Spoofing. האפליקציה מתרכזת בהגנה מפני תרחיש בו המשתמש ניגש לכתובת אתר מוכר, אך מנוטב לאתר זדוני אחר בעל כתובת זו שונה שמטרתו פגעה במשתמש (תרחישים מתוארים מטה). זדוני אחר בעל כתובת זו שונה שמטרתו פגעה במשתמש (תרחישים מתוארים מטה).

במקרה של נסיוון גישה לאתר זדוני שמתמחה באתר נורטיפי, המערכת תחשוף את גישת המשתמש לאתר הניל ובמוקם זאת, תנתב אותו לדף בטוח, בו הוא יזהה מגישה לאתר זהה, וכן יקבל הנחיות והכוונה לגבי איך להתמודד במצב של תקיפה מסוג זה (כלומר **יקבל גם העשרה לימודית לגבי הנושא**).

בנוסף, האפליקציה תהיה בעלית GUI שבו יהיה ניתן לראות את כל הפעולות הרשות העדכניות של המשתמש עם הסטטוס שלו, וכן אפשרות ליצא דו"ח מקיף יותר בו פרטים על תקריות עבר שהמשתמש חווה.

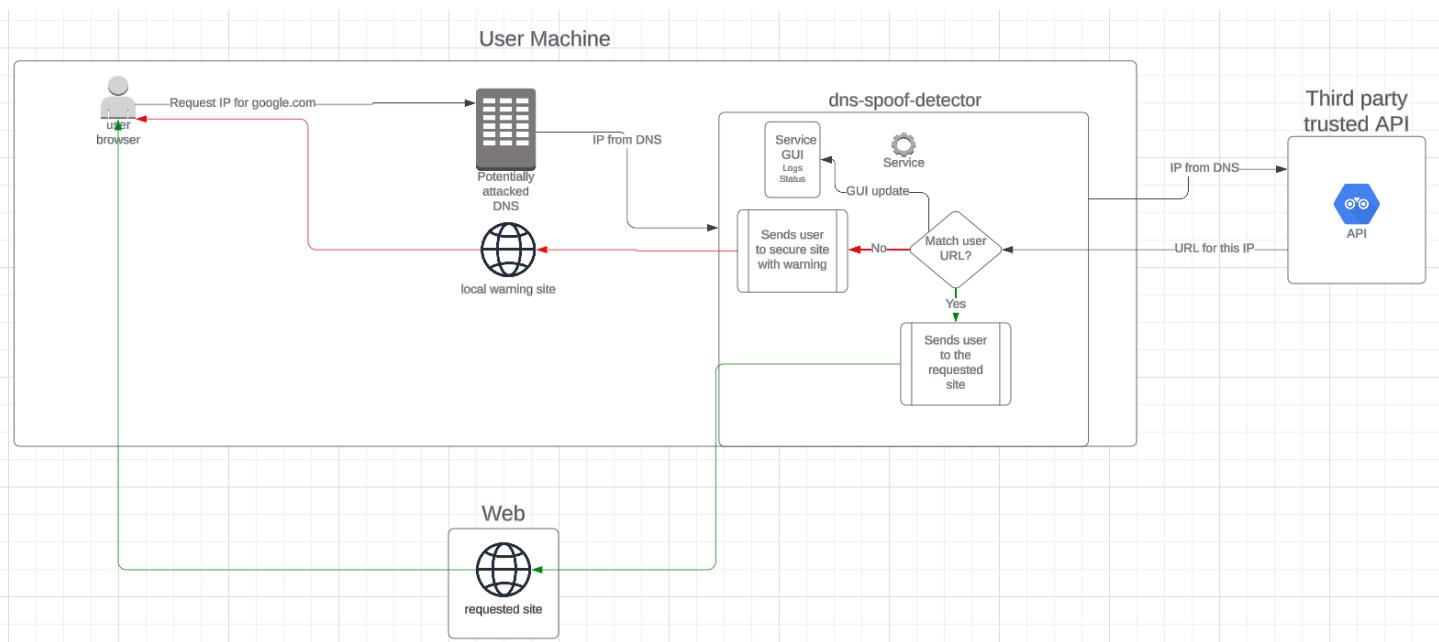
תיכון המרכיב:

פרטים טכניים על הפרויקט:

1. הפרויקט נכתב בשפת פיתון.
2. הפרויקט מיועד לרוץ על מערכת הפעלה Ubuntu עם GUI (עליה מבוסס dockerfile של הפרויקט).
3. הAPI החיצוניים בהם השתמשו בפרויקט הם: ipify, IPinfo, DNS Google API, Whois *
*תוכנו שינויים בהם השתמש לאחר תחילת העבודה על הפרויקט חלק מאילוצים שנאנחו עלולים להיות קלים בהם.
4. לצורך הוכחת יכולות וביצוע בדיקות, נעזרנו בהקמת שרת DNS מקומי: dnsmasq.

ארQUITקטורה:

https://lucid.app/lucidchart/cf307c26-78e0-4e67-90e8-8050836f3f80/edit?viewport_lo
[c=358%2C-392%2C2994%2C1416%2C0_0&invitationId=inv_3a48d9bf-1b0c-4aae-a73e-f29babe8af9c](#)



תיאור תרחישים אפשריים:

1. גניבת פרטי כרטיס אשראי:

- a. משתמש מנסה לגash לאתר קניות ברשת (לדוג' AliExpress\Amazon וכו').
- b. מכיוון שהוא עבר מתקפת DNS, הוא מנוטב במקום לאתר בעל כתובת IP אחרת שמתמחה לאתר הקניות האמיתית אליו הוא מנסה להגיא.
- c. באתר זה, המשתמש מנסה לבצע רכישה של מוצר.
- d. במהלך תהליך הרכישה, הוא מתבקש להזין פרטי כרטיס אשראי, אשר נשלחים ישירות לתוקף.
- e. המשתמש מקבל מחווה לגיטימית מהאתר שמצויה כביכול שהרכישה הבצעה בהצלחה - דבר שגורם למשתמש לא לחשוד בדבר.

2. גניבת פרטי התחרבות:

- a. משתמש מנסה לגash לרשות חברתית כלשהי.
- b. מכיוון שהוא עבר מתקפת DNS, הוא מנוטב במקום לאתר בעל כתובת IP אחרת שמתמחה לרשות החברתית האמיתית אליו הוא מנסה להגיא.
- c. במהלך ניסיון ההתחברות לחשבון האישי שלו, הפרטים האישיים שלו (שם משתמש וסיסמה) נשלחים לתוקף.
- d. התוקף יכול להציג הודעה "סיסמה לא נכונה" באתר הדוני או מחווה דומה שיכולה לגרום למשתמש למלא פרטים נוספים של חשבון המיל שלו (לאיפוס) וכן הלאה.
- e. התוקף, יכול לעשות שימושים רבים על פרטי ההתחברות של המשתמש, לרבות כופר לנויד (שליחת בקשה כופר למשתמש תוך כדי איום פרסום פרטיים אישיים שלו המשמש בחשבון הרשות החברתית שלו).

3. הורדות תוכנות זדוניות:

- a. משתמש מנסה לgash לאתר הורדת תוכן כלשהו (משחקים\מודיקה וכו').
- b. מכיוון שהוא עבר מתקפת DNS, הוא מנוטב במקום לאתר בעל כתובת IP אחרת שמתמחה לאתר ההורדות האמיתית אליו הוא מנסה להגיא.
- c. באתר זה, המשתמש מבצע הורדה של תוכן מסוים למחשב האישי שלו, אשר מתחזק להיות התוכן שהוא בקיש להורד (משחק לדוגמה).
- d. בהרצת הקובץ שהוא הורד, מוצגת תוכנה זדונית על גבי המחשב של המשתמש וחושפת אותו לשול סכנות.

4. איסוף מידע:

- a. משתמש מנסה לgash לאתר חיפוש כלשהו (גוגל למשל).
- b. מכיוון שהוא עבר מתקפת DNS, הוא מנוטב במקום לאתר בעל כתובת IP אחרת שמתמחה לאתר החיפוש האמיתית אליו הוא מנסה להגיא.
- c. האתר הדוני מציג לו תוצאות אמיתיות על החיפושים שלו (מנتب אליו לنتائج מהאתר האמיתי), ולאחר מכן חודשים אוסף את השאלות שהמשתמש מփש באתר הדוני.
- d. הדבר חושף את המשתמש לשול סכנות, בהם אפשרות של בקשות כופר, או שיווק מכון ללא ידיעת המשתמש (אם חיפש לגבי הלוחות למשל, התוקף יוכל לשוק לו הלוחות או מוצרים בסוגנון זהה ולנצל את המידע החסוי שיש ברשותו לגבי התעניניות המשתמש במוצרים מסוימים).

קיים עוד תרחישים רבים שכולם סובבים סביב התוצאות באתר לגיטימי במטרה לגרום למשתמש למסור פרטיים שהוא לא התכוון למסור לגורם שלישי.

אתגרים וקשיים:

1. מהנחת העובודה שה-DNS של המשתמש הותקף, נצטרך לוודא שanon התשובה שנתקבל מה-API החיצוני (שמטרתו לוודא כתובות URL אך מול IP) אמינה עצמה, שכן תוקף יכול גם לשבש את התשובה שנתקבל ממנו או להתחזות אליה.
 2. כדי להתגבר על קושי זה קיימות מספר דרכים -
 - a. גישה לAPI הרצויים ישירות דרך כתובות IP ולא URL.
 - b. עובודה אל מול מספר API חיצוניים (דבר שיוריד את הסבירות שהთוקף זיהה את כולם ופגע בתקשותיהם).
 - c. תקשורת HTTPS ושימוש ב-CERTIFICATES אל מול ה-API הנ"ל כדי להבטיח תשובה אמינה.

חזקות:

- האפליקציה רצה באופן תמידי ורכזף על המכונה של המשתמש ולכון המשתמש תמיד מוגן מפני מתקפות DNS, מוביל שום צורך שלו לוודא כל בקשה ובקשה שהוא שולח לדפדן.
 - לאפליקציה יש GUI ברור וידידותי שמקל על המשתמש להבין מה הסתטוס הנוכחי שלו והאם הוא נחשף לתקיפה.
 - האפליקציה מנtabת את המשתמש לחומרה קריאה אודוט המתקפה שהיא חשוף אליה ולכון לא רק מגנה עליו פנינה, אלא גם מספקת לו הכוונה איך להיות מוגן בהמשך.

לוחות זמנים:

מINI פרויקט תוכנית עבודה