**מסמך מחקר – ממ"ן 15:**

תומר ברנע – 316254713

ראשית, אפרט על חולשות פוטנציאליות בפרוטוקול. לאחר מכן אפרט על שימוש בשילוב החולשות וכיצד ניתן לנצלן למטרות זדוניות (התקפות) וכן הצעות למניעת כל התקפה.

**חולשות פוטנציאליות בפרוטוקול:**

1. אף פעולה מול השרת אינה דורשת אימות (לא מצד השרת ולא מצד המשתמש)
2. ה-Headerים של כל הבקשות עוברים ב-plain.
3. כל משתמש יכול לבקש את רשימת כל המשתמשים וכך לקבל את ה-ClientID של כל שאר המשתמשים, אשר נדרש לפעולות זדוניות אפשריות נוספות.
4. קיימות בקשות כמו Request Messages שלהן אין דרישות נוספות מעבר לידיעת ה-ClientID של המשתמש הנתקף
5. אין מגבלה על מספר לקוחות או על מספר בקשות (במידה וקיימת אופציה לשליחה ומשיכה של קבצים – אין מגבלה על הגודל שלו)
6. השרת שומר את ההודעה עד הקריאה הבאה בלבד (אמנם לא בעייתי בפני עצמו, אך בעייתי בשילוב עם חולשות נוספות)
7. ניתן לשלוח מפתח סימטרי גם ללקוחות שלא שלחו בקשת קבלת מפתח סימטרי
8. גודל ה-MessageID 4 בתים ואינו בהכרח ציקלי.

**מתקפות אפשריות ע"י משתמש זדוני**

חולשות אלה מאפשרות מגוון רחב של ניצולים.

ראשית נקדים ונאמר שברור מאליו שבשל העובדה שאין מנגנון אימות לכל אורך תהליך ההרשמה ועד שליחת וקבלת הודעות (חולשות 1+2), משתמש זדוני יכול בקלות להתחזות למשתמש אחר ולבצע פעולות בשמו. ע"מ לבצע פעולות כמשתמש אחר, על המשתמש הזדוני לדעת את ה-ClientID של הנתקף, אך לפי חולשה 3 אין זו מגבלה אמיתית.

**מתקפת DOS\DDOS -**

מתקפת DOS (Denial of service) הינה מתקפה שבה משתמש אחד או משתמשים רבים מבצעים בקשות רבות אל מול השרת במטרה למנוע ממנו לתת שירות למשתמשים אחרים ואף להקריסו.

העובדה שאין מגבלות פונקציונליות על המשתמש מאפשרת למשתמש זדוני לבצע מספר רב מאוד של בקשות אל מול השרת, ובכך לנסות להקריסו.

ניתן לעשות זאת באמצעות ריבוי בקשות מכל סוג שהפרוטוקול מאפשר - *הרשמה, רשימת לקוחות, שליחת הודעה, קריאת הודעות, שליחת קובץ* או *קבלת קובץ*. (חולשה 5)

נציין שני סוגי בקשות ספציפיים שהם מעט יותר בעייתיים (ושילוב ביניהם עלול להיות אף יותר בעייתי) –

*בקשת הרשמה -* בקשה שמאוד קל ליישם ובאופן לוגי השרת לא ירצה להגביל את כמות ההרשמות ולמנוע שירות ממשתמשים.

*בקשת רשימת לקוחות –* בקשה כבדה יחסית, על אחת כמה וכמה כאשר השרת מקבל בקשות רבות של הרשמת משתמשים חדשים, למשל אם השרת אינו מבצע Caching של התשובות ושולף בכל פעם מה-DB את רשימת הלקוחות.

בנוסף – משתמש זדוני יכול לבצע בקשות שליחת הודעות או קבצים באורך בלתי מוגבל, מה שיכול להעמיס מאוד על השרת. לדוגמה כך המשתמש הזדוני יכול לגרום לשרת להריץ הרבה Threadים לטיפול בהרבה בקשות שמתרחשות במקביל ובכך לנסות ולהקריס את השרת.

ניתן גם לשלוח הודעות עד שהשרת עובר את מגבלת גודל ה-MessageID – 4 בתים (חולשה 8)

**הצעות למניעת התקפת DDOS –**

* ניתן להגביל בצורות שונות את כמות הבקשות שהשרת מתמודד איתן, בין אם בתוך מימוש השרת או ברכיב רשתי חיצוני (כמו Firewall למשל) שמונע מבקשות רבות להתקבל.
* הוספת מנגנון התאמות אל מול השרת, ובכך לא לאפשר התחזות
* הוספת מגבלות על גודל מקסימלי של הודעה או קבצים הנשלחים בין משתמשים.

**מתקפת Man in the middle –**

כפי שציינתי בחולשה 1 – אין מנגנון אימות מול השרת, וכן אין מנגנון אימות מול המשתמש. כלומר – בהינתן התנאים המתאימים, תוקף זדוני יכול לקגרום למשתמש לגיטימי לתקשר איתו ולהעביר דרכו הודעות ומידע כאילו הוא השרת הלגיטימי.

מנגנון העברת התוכן בין לקוחות מוגן באמצעות מנגנון הצפנה משולב של הצפנה סימטרית וא-סימטרית (לצורך תיאום המפתחות הסימטריים).

תוקף זדוני בעל יכולת להאזין ולהזריק תעבורה אל משתמש לגיטימי כלשהו יכול להתערב בתעבורה.

יכולת כזו יכולה להיות ממומשת ע"י הרעלת DNS או הרעלת ARP וכו'.

בצורה כזו תוקף זדוני יכול בעצם לשמש כשרת עבור המשתמש הלגיטימי ולהעביר עבורו את התעבורה אל השרת האמיתי, ובכך להאזין לכל התעבורה שלו, ואף לשנותה.

מתקפה כזו רלוונטית בפני עצמה ומאפשרת לתוקף זדוני להתערב בתעבורה ברמת האם להעביר את ההודעות הלאה או לא, אך לא מאפשרת קריאה וכתיבה של תוכן ההודעות ע"י התוקף הזדוני, משום שאין לו בשלב זה את המפתח הסימטרי של אף אחד מהצדדים.

כדי להתערב בתעבורה התוקף צריך יכולת האזנה והזרקת מידע אל שני המשתמשים הנתקפים, ולהתערב בקבלת המפתח הפומבי הא-סימטרי של שניהם. כאשר נתקף מבקש מפתח פומבי של משתמש לגיטימי, התוקף הזדוני יעביר לו מפתח פומבי אחר, כך שברשותו המפתח הא-סימטרי הפרטי. לאחר מכן, בזמן העברת המפתח הסימטרי בין המשתמשים - התוקף הזדוני יכול לפענחו באמצעות אותו מפתח א-סימטרי פרטי (וכמובן להעביר אותו לנתקפים בצורה לגיטימית כך שיוכלו לתקשר)

לאחר מכן תוקף זדוני יכול גם לקרוא תוכן מוצפן ע"י המשתמשים ואף להתערב בתעבורה ברמת התוכן ולשנות הודעות טקסט או קבצים.

**הצעות למניעת התקפת Man in the middle –**

* ניתן לממש מנגנון חתימות ו-Certificates שמאמת עבור ה-Client שהוא אכן יוצר קשר עם שרת לגיטימי ולא עם גורם זדוני.
* אפשרי, אבל פחות פרקטי – ניתן לתאם מראש בצורה אחרת מפתחות סימטריים בין לקוחות שמעוניינים לתקשר אחר עם השני, כך שלעולם לא יעברו המפתחות עצמם בהודעות.

**מתקפת שיבוש תקשורת בין לקוחות קיימים –**

כפי שצוין, תוקף זדוני יכול להתחזות לכל משתמש ולבצע פעולות בשמו (חולשות 1+2+3+4)

כך למשל, הוא יכול למשוך מהשרת הודעות של משתמש אחר. השרת בתורו ימחק את ההודעה, ובכך התוקף הזדוני לא מאפשר למשתמש הלגיטימי לקרוא הודעות שמיועדות אליו. (חולשה 6)

ניתן אף להרחיב מתקפה זו ולקרוא באופן פריודי את כל ההודעות של כל המשתמשים ובכך לא לאפשר לאף משתמש לקבל הודעות שמיועדות אליו (סוג של DOS)

חשוב לציין שמתקפה זו הינה רק שיבוש של קבלת ההודעות, ולא מאפשרת להתערב בתוכן שלהן כמו תקיפת Man in the middle למשל.

ראה גם סעיף ב' תחת 'מתקפת מפתח סימטרי'.

**הצעות למניעת מתקפת שיבוש תקשורת בין לקוחות קיימים –**

* ניתן לדרוש אימות עבור בקשות שונות של הלקוח, בין היתר של בקשת קבלת כל ההודעות הממתינות.

**מתקפת מפתח סימטרי**

כפי שכתוב בחולשה 7 – ניתן לשלוח למשתמש לגיטימי את המפתח הסימטרי של כל משתמש ללא צורך באימות וללא צורך בבקשה מצד המשתמש הלגיטימי.

1. זה אמנם תלוי במימוש הClient, אך ניתן לגרום לו לפענח כך מראש הודעות או קבצים שנשלחים אליו, ללא רצונו של ה-Client.
2. ניתן באמצעות בקשה זו גם לשבש תקשורת בין לקוחות ע"י דריסת המפתח הסימטרי ע"י מפתח לא רלוונטי. לאחר מכן המשתמש הנתקף לא יוכל יותר לפענח הודעות ממשתמש אחר עד שלא ישלח בקשה חדשה למפתח סימטרי שתיענה ע"י המשתמש המרוחק. כמובן שכל הודעה שהתקבלה בינתיים ולא התפענחה כראוי תימחק מהשרת ולכן הנתקף לא יוכל לקרוא אותה עד שלא תישלח מחדש.

**הצעות למניעת מתקפת מפתח סימטרי –**

* ניתן להגביל (בשרת) את האפשרות לשלוח מפתח סימטרי ללקוח שלא ביקש את המפתח הסימטרי.
* ניתן לדרוש אימות עבור בקשות שונות של הלקוח, בין היתר של שליחת מפתח סימטרי.