Digital Whisper
Technologic Papers

Zip Bombs

מאת Crossbow ו-cp77fk4r

הקדמה ורקע כללי

במאמר זה נסביר את מושג הנקרא "פצצות Zip" (Zip Bombs או Zip of Death). כמו כן נסביר איך ליצור אותה ואילו שימושים מעניינים ניתן לעשות איתה.

פצצות Zip אלו קבצי Zip אשר נוצרו בעבר על מנת לגרום לקריסת מערכת ההפעלה. כיום, מערכות ההפעלה יודעות להתמודד איתן ולכן הן פחות מסוכנות למערכת ההפעלה- אך לאפליקציות אנטי-וירוס אשר לא נכתבו כך שיכולו "לטפל בהן" - הן מסוכנות מאוד.

התיעוד הראשון למתקפה שכזאת הוא מאמצע שנת 2001 ע"י בחור בשם Michel Arboi, ועוד אז היא סוייגה כ-"Failure to Handle Exceptional Conditions" (שכמובן הובילה ל- Local denial of service attack). הרעיון המרכזי בהתקפה הוא לנצל עקרון בסיסי מאוד במנגנוני הדחיסה של אלגוריתמי הכיווץ וליצור קובץ Zip בעל יחס דחיסה (compression ratio) גבוה מאוד. כך-בזמן פתיחת הקובץ בעזרת מנגנוני פתיחת הכיווץ של תוכנות האנטי-וירוס בעת סריקתו-לגרום לו לקריסה. חשוב לשים לב כי מדובר ביחס דחיסה אדיר. לדוגמא, ה-Zip Bomb המפורסמת ביותר היא "פצצה" בשם "42.zip", מדובר בקובץ Zip בגודל 42kilobytes (ומכאן שמה) שלאחר פתיחת דחיסתו הוא נהפך לקובץ המכיל 4.5peta-bytes (כל peta-bytes אחד מכיל 1024 tera-bytes, וכל terabytes אחד מכיל 1024giga-bytes) משמע יחס דחיסה של יותר **ממאה ביליון** bytes! הזוי משהו.

קצת תיאוריה

בכדי להבין איך זה אפשרי ליצור פצצה שכזאת עלינו להכיר מקרוב את עולם דחיסת המידע, איך בכלל אפשר לכווץ קובץ? איך אנחנו יכולים לאחסן את אותו המידע כך שיתפוס פחות נפח בכונן שלנו? השאלות האלה קצת מפוצצות, אך התשובות להן בעצם לא כל כך קשות להבנה. כל הרעיון בדחיסת מידע הוא שינוי האופן בו המידע נשמר. לדוגמא, קיימים היום הרבה מאוד סוגים של פורמטים לקבצי

גרפיקה, מהמוכרים בינהם אפשר למנות את ה-BMP, JPG, PNG ועוד. שימו לב שאם תציירו תמונה בצייר ותשמרו אותה ב-BMP היא תתפוס נפח מסויים (יחסית גדול), ואם תשמרו את אותה התמונה בפורמט שונה, למשל-JPG, תראו שהיא תתפוס נפח שונה לחלוטין (וקטן בהרבה) מאותה התמונה שנשמרה בפורמט BMP.

איך זה קורה?

הסיבה הראשונה-התמונה היא לא אותה התמונה. כאשר שומרים תמונה ב-JPG האיכות שלה יורדת. מה זאת אומרת "האיכות יורדת"? זאת אומרת שאם תשימו לב טוב טוב מספר הגוונים בתמונה ירד, התמונה הרבה פחות חדה ואפשר לזהות שהיא קצת "מרוחה".

הסיבה השניה (והחשובה לנו)-אופן שמירת מידע התמונה בזכרון המחשב שונָה.

מה זאת אומרת?

במדעי המחשב, ישנו ענף שלם העוסק בתחום זה ונקרא "דחיסת נתונים". לא נסביר את כולו כי זה לא נושא המאמר, ולכן רק נצטט מויקיפדיה תחת הערך "דחיסת נתונים":

> "דחיסת נתונים אפשרית משום שבנתונים בצורתם הגולמית קיימת פעמים רבות יתירות גבוהה, כלומר מידע החוזר על עצמו או מידע שניתן לייצג בצורה חסכונית יותר. העיקרון הבסיסי בדחיסה הוא כי מידע שאנו עושים בו שימוש יש בו "סדר" מסוים. מציאת הסדר הזה מאפשרת לייצג את המידע בדרך יעילה יותר. מידע שהוא מקרי (רעש) לא יכול להידחס. מידת הסדר מכונה 'אנטרופיה.'"

(צוטט מויקיפדיה תחת הערך: דחיסת נתונים)

ישנם מספר רב מאוד של אלגוריתמי דחיסת נתונים שונים, המוכרים שבהם הם קוד הופמן, קידוד אורך חזרה, ועוד מספר רב של אלגוריתמים שאנו משתמשים בהם בדרך קבע במהלך היום-יום בעת הפעלה של קבצי תמונה, קבצי שמע או וידאו. אנחנו מדברים כמובן על קבצים כגון mp3, mpeg, jpg, png, avi ורבים אחרים.

העקרון אותו מנצלים בעת יצירת Zip Bombs הוא אפקט יחס הכיווץ הגדול הנוצר בעת דחיסת מידע בעל מידת אנטרופיה גבוהה ביותר. וכמו שראינו, ככל שהאנטרופיה בנתוני הקובץ גבוהה, כך יחס הכיווץ יגדל. איפה נקבל את מידת האנטרופיה הגבוהה ביותר? ברב המקרים-יהיה זה כאשר כל נתוני הקובץ זהים לחלוטין.

כך למשל, את המידע הבא:

```
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
                                AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

נוכל לציין על פי RLE באופן כל כך פשוט בדרך הבאה: A187

קידוד Huffman

אם נקח את WinZip כדוגמא, נראה שבכדי לדחוס את הנתונים שבה היא משתמשת במעין שילוב של אלגוריתם הדחיסה LZ77 וקידוד Huffman.

שיטת קידוד Huffman נחשבת לאופטימלית במובנים רבים, האופטימליות מתבטאת בעיקר בכך שהיא דוחסת נתונים באופן מירבי, כלומר, משתמשת בהכי מעט ביטים שאפשר כדי לייצג כל תו. הרעיון הוא שלתווים שיחזרו על עצמם הרבה יוקצו מעט ביטים, בעוד שתווים 'נדירים' ישתמשו בהרבה ביטים.

כמה מונחים לא פורמליים בתורת הגרפים:

- גרף הוא צומת של צמתים וקשתות. ניתן לדמיין את הצמתים כ-"איים", בעוד הקשתות הם חיבורים בין הצמתים.
- גרף יכול להיות מכוון או לא מכוון. גרף מכוון הוא גרף שבו יש כיוון לקשתות.
- עץ הוא גרף ללא מעגלים, בו קיים צומת (לו קוראים 'שורש העץ') שממנו קיים מסלול אל כל צומת אחר.
- בעץ, לכל צומת יש צומת אב (מלבד לשורש). צומת שלו אין בנים נקרא עלה.

- עץ בינארי הוא עץ בו לכל צומת שאינו עלה יש לכל היותר שני בנים.

פעולת האלגוריתם:

האלגוריתם בונה עץ בינארי הפוך, כך שבעלים (צמתי הקצה) יש תווים. ניתן לתאר אותו כך:

- בנה טבלת תדירויות עבור כל תו, כלומר, בנה עלים כך שבכל עלה כתוב התו וכמה פעמים הוא חוזר על עצמו.
- מצא את שני הצמתים היתומים (צמתים ללא אב) המינימליים במספרם.
- צור צומת חדש שיהווה את אב שני הצמתים הללו, וערכו יהיה חיבור ערכם של בניו.
- אם הצומת החדש הוא הצומת היתום היחיד, סיים.

כעת, נקבע שכל בן ימני הוא ביט 0, וכל בן שמאלי הוא ביט 1. הקידוד של תו מסויים הוא שרשור הביטים על הקשתות המובילות מהשורש אל התו עצמו (שהוא עלה). לכן, ניתן לקודד כעת את הטקסט המקורי, ובמקום כל תו לשים את הקידוד שלו. הדבר היחיד שצריך להוסיף הוא טבלת התדירויות המקורית, או לחילופין את הקידוד של כל תו.

ניתן למצוא מידע נוסף ומורחב על האלגוריתם במאמר של גיל כהן בפרוייקט UnderWarrior בקישור הבא: http://www.underwar.co.il/document-details.asp?id=134

פעולות ושימושים

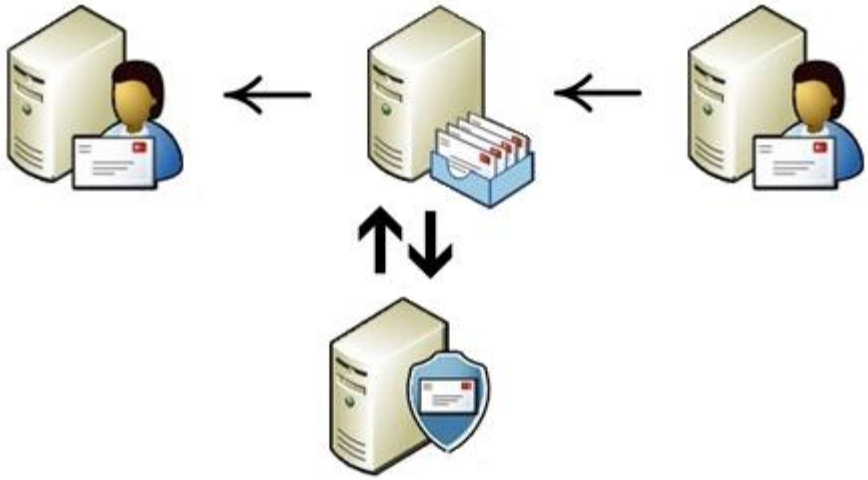
לאחר שהבנו איך העניין עובד, נשאלת השאלה-למה אנחנו צריכים את זה? חוץ מלהקריס לנו את תוכנת WinZip-על ימין ועל שמאל (מה שכיום גם כבר לא יקרה), איזה שימוש יעיל אפשר למצוא בחיה שכזאת? אחד השימושים היעילים שאפשר למצוא לפצצת Zip הוא ניטרול תוכנת ה-Anti Virus על מחשב מרוחק, לדוגמא-שרת ה-Exchange של אירגון מסויים. באירגונים גדולים המאפשרים לעובדיהם שליחת מיילים אל כתובות אינטרנט הנמצאות מחוץ לרשת הפנימית של האירגון, קרי: לתקשר עם מיילים כגון ג'ימייל, יאהו, ומיילים של עובדים מאירגונים אחרים- ישנו "Mail Delivery Server" שאחראי על שליחת המיילים. השרת מתפקד מעין "Default Gateway" לתכנות Outlook ודומיהן.

- דוגמא נפוצה לשרתים כאלה ברשתות חלונאיות היא שרתי ה-Exchange של מיקרוסופט.
- דוגמא נפוצה לשרתים כאלה ברשתות לינוקסאיות/יוניקסאיות היא שרתי Cyrus IMAP עם Sendmail.

מכאן תצורת המערכת יכולה להיות (בדרך כלל) אחת מהשתיים הבאות:

- על השרתים הללו מותקנת תוכנת (או מספר) אנטי-וירוס אשר מבצעת סריקת לכלל הקבצים המצורפים לתכתובות המייל (Attachments) - ומאשר האם להעביר הלאה או האם מדובר בקובץ המכיל וירוס:

- שרת המייל מעביר את הקובץ המצורף לשרת אנטי-וירוס ייעודי אשר מותקנת עליו סביבת אנטי-וירוס ייעודית:



ישנה תצורה נוספת שבה שרת האנטי-וירוס מתפקד גם כ-"Mail Relay" והוא זה השולח את המייל לאחר בדיקתו.

חבילת אנטי-וירוס מוכרת היא למשל GFI MailSecurity הכוללת בתוכה את חמשת מנועי האנטי-וירוס הבאים היודעים לבצע עבודה במקביל:

- McAfee
- Norman
- AVG
- BitDefender
- Kaspersky

לאחר הסריקה-השרתים מחזירים הודעה לשרת המייל האם יש אישור להעביר את הדוא"ל לנמען או האם יש למנוע מהוירוס להגיע ליעדו. חבילה נוספת המוכרת גם היא, היא-McAfee Email Gateway של McAfee. כל מתקפה מוצלחת על אחד מהתצורות שהצגנו תוביל להתקדמות קריטית מבחינת התוקף.

- במידה ומדובר בתצורה הראשונה-קיים סיכוי שתוכנת האנטי-וירוס תקרוס ומעתה-כל אימייל אשר ישלח דרך השרת ישר יעבור לנמען מבלי להיסרק בדרך.
- במידה ומדובר בתצורה השניה/שלישית-קיים סיכוי ששרת האנטי-וירוס יתקע או יקרוס והדבר יוביל ל-Denial of Service על מנגנון שליחת המיילים של האירגון-מה שימנע מכלל המשתמשים לקבל/לשלוח אימיילים על גבי רשת האירגון.

איך אפשר לדעת איזו תוכנת אנטי-וירוס יושבת על השרת של האירגון? פשוט מאוד-תוכנת האנטי-וירוס לרב מוסיפה חתימה לסוף המייל אשר מעידה על כך שהדואר והקבצים המצורפים אליו-נבדקו על-ידייה. לדוגמא, זאת חתימה שנוספה לקובץ שנסרק ע"י תוכנת האנטי-וירוס MailMarshal של m86security:

This e-mail message has been scanned for Viruses and Content and cleared by **MailMarshal**

זאת חתימה שנוספה לקובץ שנסרק ע"י תוכנת האנטי-וירוס MailScanner:

This message has been scanned for viruses and dangerous content by **MailScanner**, and is believed to be clean.

(תודה רבה ל-**spdr** ול-**execute** על החתימות)

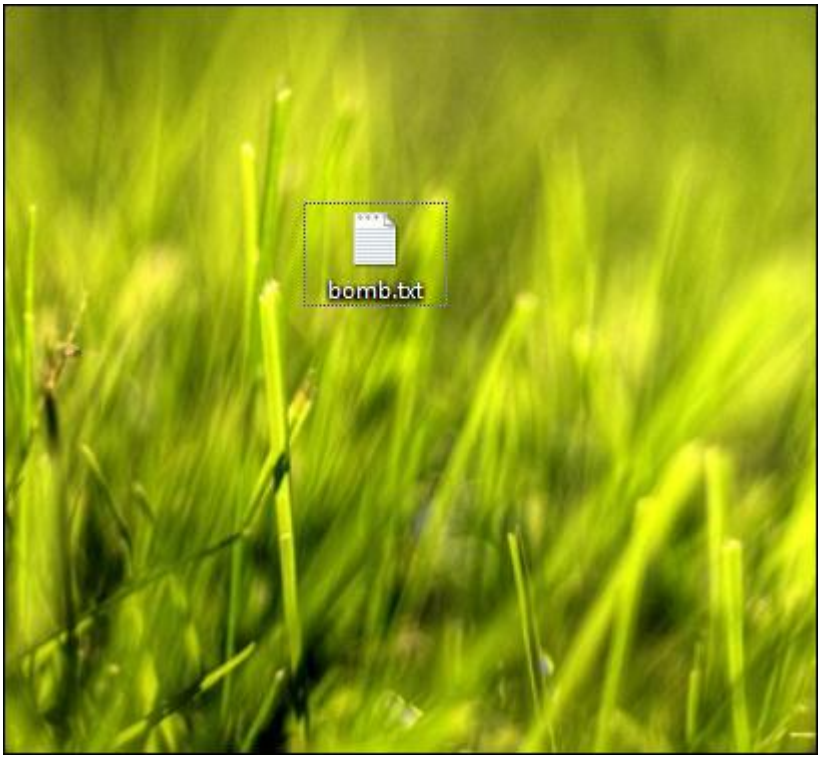
גם במקרים בהם שם תוכנת האנטי-וירוס לא מופיעה בחתימת האנטי-וירוס, מספיקה הרצה קצרה של החתימה בגוגל בכדי להבין לאיזו תוכנת אנטי-וירוס שייכת החתימה.

אופן ביצוע המתקפה

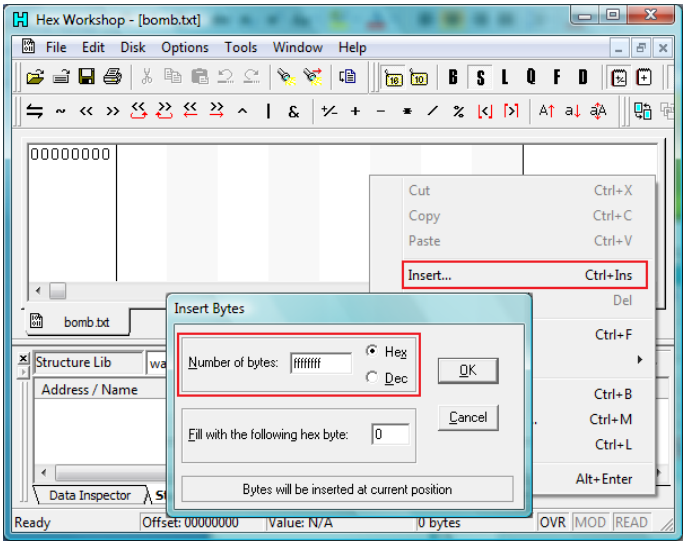
שלבי המתקפה ברורים ביותר- הרעיון הוא לשלוח קודם כל את ה-Zip Bombs אל שרתי האירגון ולאחר מכן לשלוח את המייל עם הוירוס/סוס טרויאני. במידה והצלחנו לבצע את המתקפה בהצלחה, או שהוירוס יעבור באופן חלק ללא כל בדיקת אנטי-וירוס, או ששום מייל לא יוכל להכנס לאותו אירגון - או שכלום לא יקרה ☺

אחרי שהבנו את התיאוריה והרעיון הכללי- בואו נראה איך אפשר להכין Zip Bomb משלנו.

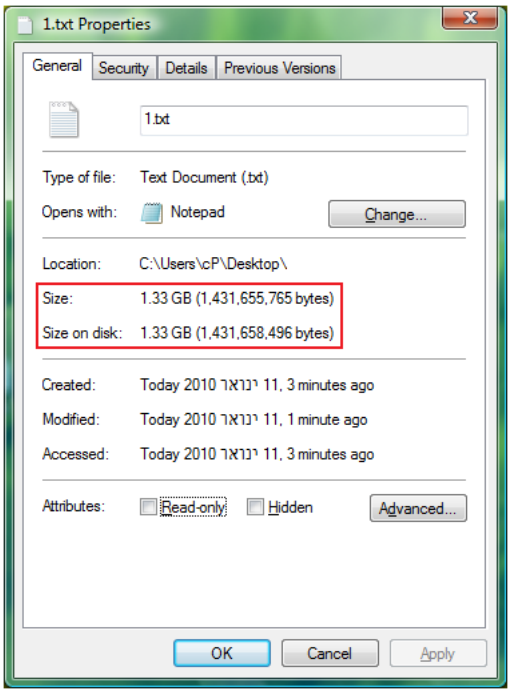
צרו קובץ טקסט בשם: Bomb.txt



פיתחו אותו בעזרת תוכנת Hex Editor והכניסו לו מספר רב של בייטים ריקים:



פעולת ה-Filling תמשך קצת-ויש סיכוי שתקבלו שגיאת Access Denied עקב נסיון יצירת קובץ גדול ממה שהמערכת מאפשרת ולכן יש סיכוי שתאלצו להוריד את מספר הבייטים שתרצו להוסיף בתוכנת ה- Hex Editor, לאחר שתוסיפו את הבייטים-שמרו את הקובץ. בסוף הפעולה תקבלו קובץ גדול יחסית מלא בבייטים ריקים:



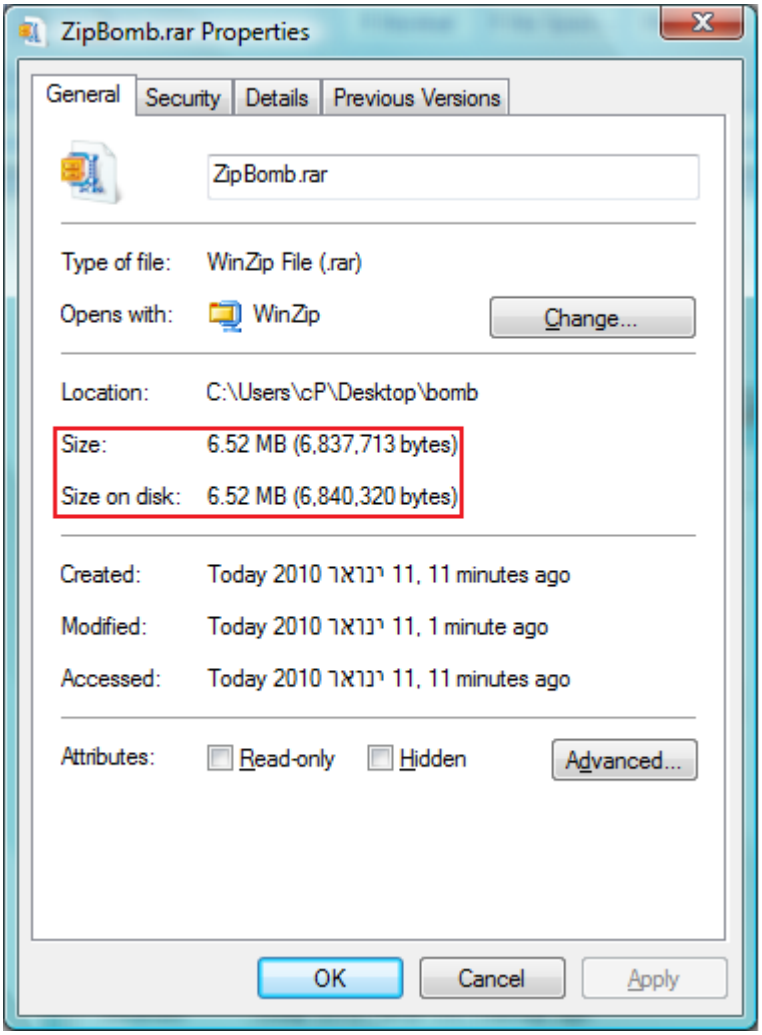
הקובץ אומנם ענק, אך מפני שכלל ערכי הבייטים שבו זהים האנטרופיה שלו גבוהה מאוד. צרו העתקים רבים של הקובץ והכניסו אותם לתיקיה חדשה. ב-CMD נווטו לתיקיה והקלידו את הפקודה:

Copy *.txt Zipbomb.txt

למעבד יקח פרק זמן מכובד לבצע את הפעולה ולאחר מכן יהיה לכם קובץ אשר יכיל את כלל הקבצים. כעת מיחקו את יתר הקבצים. אנו יצרנו שבעה העתקים חדשים לקובץ הקיים, ואת כל שמונת הקבצים המרנו לקובץ בודד השוקל כמעט 11.5 gigabytes. כאן אנו רק ממחישים לכם את הדבר, אך לצורך בניית הפצצה אנו צריכים נפח גדול הרבה יותר.

```
Administrator: C:\Windows\system32\cmd.exe

C:\Users\cP\Desktop\bomb>dir
 Volume in drive C has no label.
 Volume Serial Number is 5A75-7CCD

 Directory of C:\Users\cP\Desktop\bomb

01/11/2010  06:32 PM    <DIR>          .
01/11/2010  06:32 PM    <DIR>          ..
01/11/2010  06:17 PM     1,431,655,765 bomb.txt
01/11/2010  06:17 PM     1,431,655,765 bomb1.txt
01/11/2010  06:17 PM     1,431,655,765 bomb2.txt
01/11/2010  06:17 PM     1,431,655,765 bomb3.txt
01/11/2010  06:17 PM     1,431,655,765 bomb4.txt
01/11/2010  06:17 PM     1,431,655,765 bomb5.txt
01/11/2010  06:17 PM     1,431,655,765 bomb6.txt
01/11/2010  06:17 PM     1,431,655,765 bomb7.txt
               8 File(s) 11,453,246,120 bytes
               2 Dir(s)  12,846,669,824 bytes free

C:\Users\cP\Desktop\bomb>copy *.txt ZipBomb.txt
bomb.txt
bomb1.txt
bomb2.txt
bomb3.txt
bomb4.txt
bomb5.txt
bomb6.txt
bomb7.txt
        1 file(s) copied.

C:\Users\cP\Desktop\bomb>del bomb*.txt

C:\Users\cP\Desktop\bomb>dir
 Volume in drive C has no label.
 Volume Serial Number is 5A75-7CCD

 Directory of C:\Users\cP\Desktop\bomb

01/11/2010  06:53 PM    <DIR>          .
01/11/2010  06:53 PM    <DIR>          ..
01/11/2010  06:53 PM    11,453,246,121 ZipBomb.txt
               1 File(s) 11,453,246,121 bytes
               2 Dir(s)  18,279,866,368 bytes free

C:\Users\cP\Desktop\bomb>
```

כעת, דחסו את הקובץ הנותר בעזרת תוכנת דחיסה כגון WinZip או WinRAR. גם הפעולה הזאת תמשך פרק זמן לא קצר, אך בסיומה תווצר לכם הפצצה. אצלנו הקובץ יצא בגודל של כ-6.5megabytes. אנשים לפנינו הצליחו להגיע לייחס הרבה יותר רציני ויעיל-אך בשביל כתיבת המאמר היחס הזה מספק.



בכדי להגיע ליחס איכותי יותר משלנו אפשר לבצע למשל:

- הכפלה של נתונים הדחיסה בעזרת עורך HexEditor.
- ייעוד סידור תוכן הקובץ באופן ספציפי כלפי אלגוריתם דחיסת הנתונים הספציפי.

הפצצה מוכנה, כעת נותר לנו רק לשגר אותה.

דרכי התגוננות

כיום כבר רב האנטי-וירוסים מצויידים במנגנון כנגד Zip Bombs. ישנם מספר דרכים ליישם מנגנון שכזה:

- ניתוח הקובץ הדחוס בטרם פריסתו והשוואה בינו לבין גודלו העתידי.
- פריסת הקובץ בארגז חול יעודי בלתי-תלוי במשאבי השרת ואתחולו בכל פריסה חדשה.
- קביעת גודל מקסימלי והשוואתו לגודל הקובץ **בזמן פריסת הקובץ** ולא בסופה.
- קביעת פרק זמן מקסימלי לפעולת פריסת הקובץ.

בנוסף, ההמלצה של כל החברות המספקות שרתי אנטי-וירוס היא שהשרת יהיה חזק מאוד, יציב ומתוקצב במשאבים רבים.

סיכום

מתקפות בעזרת פצצות ZIP הם לא דבר חדש, אך עדיין, גם כיום אפשר למצוא תוכנות אנטי-וירוס אשר עדיין רגישות למתקפות מסוג זה, דוגמא לתוכנות אנטי-וירוס שגם כיום פגיעות למתקפות אלה הן:

- Dr. Web cureit - כל הגרסאות (נכון לכתיבת שורות אלה הגרסא החדשה ביותר היא 5.00.9)
- Clam AntiVirus - כל הגרסאות (נכון לכתיבת שורות אלה הגרסא החדשה ביותר היא 0.95.3)

מומלץ בחום לבדוק את גרסאת האנטי וירוס שאתם משתמשים בה במחשבכם.