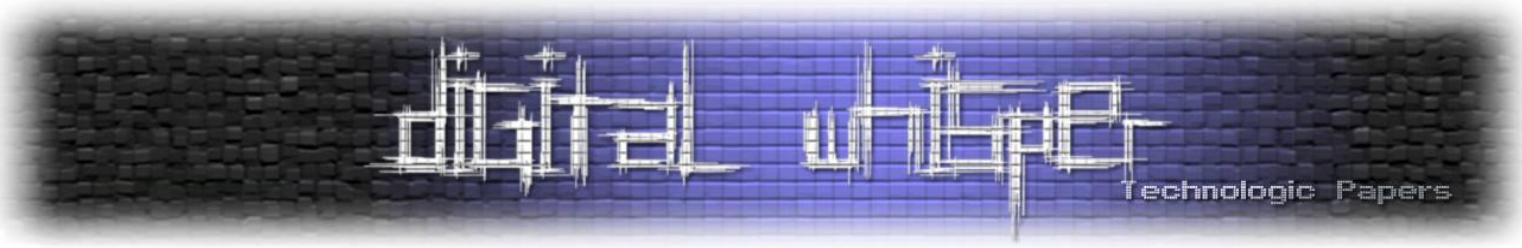


זיהוי ומניעת חיבור שרותי פרוקסי אנונימיים

מאת: אדיר אברהם (Adir@computer.org)

הקדמה

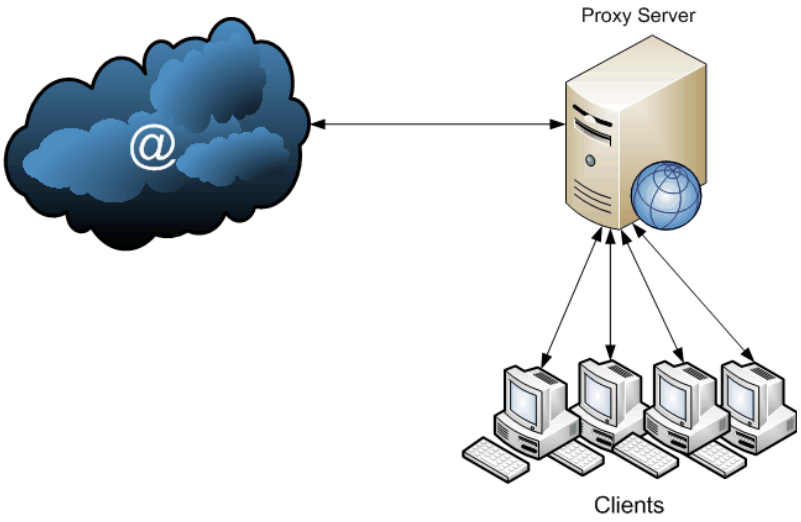
במהלך השימוש של משתמשים שונים במערכת מתרחשים אירועים שונים אשר מחייבים את מעורבות מנהל המערכת או מומחה אבטחת המידע המקומי. במהלך הפעלת השרת וניטורו מתגלים נסיונות חדירה למערכת אשר מגיעים ממקומות שונים ברשת, כאשר אחד המקומות הבעייתיים ביותר הוא משרתי פרוקסי. נתחיל בשאלה למה מיועדים שרתי פרוקסי?

- לאכוף איסורי-גישה פנים-ארגוניים שונים.
- להגדיל את המהירות למשאבי הרשת השונים.
- לבצע caching בין אתרים או דפים פופולריים לבין מחשבים ספציפיים שניגשים אליהם תכופות.
- להתיר גישה לתוכן מסויים או לאסור גישה אליו.
- לבצע גישה פנים-אירגונית בעזרת login מסודר דרך שרת ספציפי.
- לסרוק מעבר מידע לפני הגעתו לתוך האירגון או מחוצה לו מוירוסים ומזיקים אחרים.
- למנוע דליפת מידע רגיש החוצה (DLP).
- לשמור על המחשבים שנמצאים בתוך הארגון כאנונימיים.

אם נסכם את מטרת שרתי הפרוקסי - שרת פרוקסי משמש כמתווך בין בקשות של לקוחות לבין שירותים או שרתים אחרים אשר מהם הם מיועדים לקבל את המידע. שרת הפרוקסי מתוקף תפקידו יכול "להחליט" איזה מידע להעביר הלאה מצד הלקוח ואיזה להשאיר. באופן כזה, שרת הפרוקסי עשוי לשנות את המידע המגיע המזהה את הלקוח ולסננו כך שהלקוח לא יזוהה.

שימוש נוסף עליו לא נדבר כאן, הוא ששרת הפרוקסי יכול להשאיר אצלו את המידע מהשרת ממנו אנו מייעדים לקבל שירות רלוונטי, כך שנקבל משרת הפרוקסי את המידע ישירות ולא נצטרך להתחבר אל השרת המרוחק. האופציה האחרונה מאפשרת מתן שירות יעיל יותר ע"י שיפור המהירות (תהליך הנקרה "Caching").

בגדול, זה נראה כך:



(התמונה פורסמה במקור במאמר "Java Java Proxy Proxy", שנכתב על-ידי רועי חורב (AGNil) ופורסם בגליון ה-11 של המגזין. - שווה לקרוא בכדי להבין יותר על תפקידיהם של שרתים אלו)

זן "מיוחד" של שרתי פרוקסי הוא שרת פרוקסי אנונימי, המאפשר למעשה סינון מוחלט של זיהוי הלקוח המתחבר (למשל, את כתובת ה-IP שלו, פרטי הדפדפן שלו וכו'), ואם לקוח זה יתחבר דרך שרת הפרוקסי אל השרת שלך, הוא למעשה יוכל לדלג על איסורים שונים שמנהל הרשת קבע במערכת שלו, למשל חיבור מתחום כתובות IP מסויים (כגון מדינה).

כיום השימוש בשרת פרוקסי אנונימי נעשה נפוץ יותר לנסיונות החדירה הרבים לשרתים שונים. במאמר זה נדבר על זיהוי ומניעת שימוש בשרתי פרוקסי שונים.

זיהוי פרוקסי על-ידי כותרים

לא תמיד, אך במספר לא מבוטל של מקרים, שרתי HTTP Proxy מוגדרים להוסיף מספר כותרים (Headers) לבקשת ה-HTTP שהמשתמש שולח. דוגמא לכותר כזה הוא כותר ה-"X-Forwarded-For" (או מוכר גם כ-"XFF"), הערך שיגיע לאחר הכותר הנ"ל הוא כתובת ה-IP המקורית של לקוח הפרוקסי, בכדי למנוע את האפשרות להתחבר לשרת שלנו בעזרת שרותי פרוקסי המוסיפים את הכותר הנ"ל פשוט מאוד נוכל לבדוק האם בבקשת ה-HTTP שקיבלנו- מופיע הכותר הנ"ל. במידה הוא אכן מופיע אז כמעט ואין ספק כי מדובר בלקוח הגולש דרך שרת פרוקסי.

חשוב לציין כי הכותר הנ"ל אינו מופיע ב-RFC המקורי של הפרוטוקול והוא נוסף על-ידי המפתחים של שרת הפרוקסי Squid.

כותרים עם תפקידים דומים שהמצאותם בבקשת ה-HTTP שהגיע לשרת שלנו תוכל להדליק אצלנו נורה אדומה הם: (ותודה רבה לרועי / Hyp3rInj3ct10n שאסף את הכותרים הללו והציג אותם במאמר "Playing With HTTP" שפורסם בגליון השמיני של Digital Whisper!)

- Client-IP
- Proxy-User
- Forwarded
- Useragent-Via
- Proxy-Connection
- Xproxy-Connection
- Pc-Remote-Addr
- Via

דוגמא לחסימת בקשות HTTP הכוללות את אחד מהכותרים הנ"ל בעזרת שימוש בקבצי htaccess. ניתן לראות בקוד הבא:

```
RewriteEngine on
RewriteCond %{HTTP:VIA}                  !^$ [OR]
RewriteCond %{HTTP:FORWARDED}            !^$ [OR]
RewriteCond %{HTTP:USERAGENT_VIA}        !^$ [OR]
RewriteCond %{HTTP:X_FORWARDED_FOR}      !^$ [OR]
RewriteCond %{HTTP:PROXY_CONNECTION}     !^$ [OR]
RewriteCond %{HTTP:XPROXY_CONNECTION}    !^$ [OR]
RewriteCond %{HTTP:HTTP_PC_REMOTE_ADDR} !^$ [OR]
RewriteCond %{HTTP:HTTP_CLIENT_IP}       !^$
RewriteRule ^(.*)$ - [F]
```

הקוד מופיע במקור בבלוג "perishablepress.com", בקישור:

http://perishablepress.com/press/2008/04/20/how-to-block-proxy-servers-via-htaccess/

במידה ואתם מכירים שרת המשמש כשרת פרוקסי ציבורי, ניתן לשמור את כתובתו בתוך רשימה שחורה ולהפיץ אותה באינטרנט. באופן זה בעלי שרתים אחרים יוכלו להשתמש ברשימה זו גם כן ובמקום לעדכן בכל פעם מחדש את שרתי הפרוקסי המתגלים, תוכלו להשתמש ברשימה שמכילה מקרים קודמים כדי לסנן חיבורים בלתי רצויים.

אמנם, רשימה כזו מתעדכנת וגדלה כל הזמן, אך השימוש בה לטווח הרחוק שווה את ההשקעה, מכיוון שכך אנו גורמים לשרתי פרוקסי אנונימיים זדוניים לא להיות בשימוש. בנוסף, לעתים אין דרך מסודרת לגלות שרתי פרוקסי אנונימיים בעייתיים, ואז בידיעה על אחד מהם נוכל להוסיף אותם לרשימה בעצמנו ולמנוע את השימוש בהם אצלנו.

לדוגמא, אחד האתרים שמכילים רשימה כזו הוא האתר http://proxy4free.com. נוכל לדלות את הרשימה ממנו, למשל ע"פ הדומיין באמצעות הפקודה הבאה:

```
curl http://proxy4free.com/list/webproxy_domain1.html > proxy_list1.html
curl http://proxy4free.com/list/webproxy_domain2.html > proxy_list2.html
curl http://proxy4free.com/list/webproxy_domain3.html > proxy_list3.html

…
```

לאחר שהורדנו את דפי ה- HTML אלינו, נהפוך את רשימת הדומיינים לקובץ טקסט אחד גדול ונשתמש בו:

```
grep whois\.cgi\?domain\=proxy_list1.html | cut -d \= -f 3 | cut -d \" -
f 1 | sort | uniq > proxy_list.txt
```

```
grep whois\.cgi\?domain\=proxy_list2.html | cut -d \= -f 3 | cut -d \" -
f 1 | sort | uniq >> proxy_list.txt
```

```
grep whois\.cgi\?domain\=proxy_list3.html | cut -d \= -f 3 | cut -d \" -
f 1 | sort | uniq >> proxy_list.txt
```

נוכל לעדכן את הרשימה באופן יום-יומי ואוטומטי (בעזרת cron/schtasks) מתוך הרשימה הראשית הנ"ל, כך שלא נצטרך לעדכן אותה בעצמנו בפועל.

זיהוי שרתי TOR

TOR הוא פרויקט המאפשר גישה אנונימית ברשת ע"י שימוש ברשת מבוזרת של שרתים ברחבי העולם. (ניתן לקרוא עוד על הפרוייקט במאמר שפורסם בגליון השביעי של Digital Whisper על-ידי ליאור ברש) שרתי TOR עובדים על פורטים 9001-9004, 9030-9033 ו-9100 בנוסף לפורט 80 ו- 443 הסטנדרטיים.

נוכל להשתמש ב-Snort כדי לזהות שימוש בשרת TOR:

```
alert tcp $HOME_NET any -> $EXTERNAL_NET 80,443,9001,9030 (msg: "TOR
client access
detected"; pcre:"/.*(Tor).+(client <identity>).*/i"; classtype:policy-
violation; sid:50009;
```

שורה זו תגלה שימוש בפורטים הנ"ל ותתריע על גישת TOR, כמובן שבמידה ותוכנה אחרת תבצע שימוש בפורט הנ"ל בכדי להעביר את המידע לשרתים שלנו- נקלע ל-False Positive.

בנוסף, בעזרת אינטגרציה קלה, ניתן לבצע שאילתות מול אתר כגון torstatus.blutmagie.de כדי לדעת האם כתובת ה-IP שמנסה להתחבר משמשת כתחנת TOR.

איתור תבניות פרוקסי ידועות

PHPProxy

לשרתי פרוקסי ישנן "חתימות" ידועות הבנויות ממנועים עיקריים. אחד מהם הוא PHPProxy שנמצא באתר:

http://sourceforge.net/projects/phpproxy

למשל, אם ננסה להתחבר לאתר ynet דרך שרת הפרוקסי שנמצא בכתובת proxy.arcticgames.net, אנחנו נקבל את הכתובת הבאה:

http://proxy.arcticgames.net/proxy/index.php?q=aHR0cDovL3d3dy55bmV0LmNvLmlsL2hvbWUvMCw3MzQwLEwtOCwwMC5odG1s

כלומר, התבנית היא:

```
{hostname}/index.php?q={encoded_URL}
```

במצב כזה, שרת הפרוקסי יפתח מסגרת בדף כך שבתוך המסגרת יש את האתר אליו אנו רוצים לגשת. אתר האינטרנט אליו ניגשנו יראה את כתובת ה-IP וכן את החתימה של proxy.arcticgames.net ולא שלנו.

הכתובת הנ"ל מקודדת בעזרת Base64, אותו ניתן לפרש בעזרת כלים / אתרים ייעודיים כגון:

www.opinionatedgeek.com/dotnet/tools/base64decode

מספיק שנכניס את המחרוזת לאחר ה-"?q=" ונראה את הכתובת המקורית. באופן זה נוכל לדעת לאיזה דף ניסו לגשת בפועל. בנוסף לכך נוכל לראות לעתים מקרים דומים, כגון "Rotate13" אשר מקדם את האותיות 13 פעמים קדימה, כך ש-"www" יופיע בתור "jjj" וכן הלאה.

לסיכום, כדי לגלות שימוש בשרתי פרוקסי מסוג PHPProxy, נוכל לזהות ולסנן בקשות מהסוג הנ"ל ע"י שימוש בשורה הבאה:

```
grep -Ei '(index\.php\?q=).+(&hl).*' proxy_list.txt
```

וכך נכניס עוד שרתי פרוקסי לרשימה.

CGIProxy

CGIProxy הוא מנוע פרוקסי נוסף אשר בנוי בשפת Perl. בתור ברירת מחדל אין ערבול של ה- URL אליו אנו רוצים להתחבר, אך יש שימוש אופציונלי ב-Rotate13 וב-Base64.

למשל, עבור אתר הפרוקסי הבא:

http://rosinstrument.com/cgi-proxy.htm

נוכל להתחבר לאתר ynet ונקבל את הכתובת הבאה:

http://antt.tk/u.php/Oi8vd3d3/LnluZXQu/Y28uaWwv/aG9tZS8w/LDczNDAs/TC04LDAw/Lmh0bWw_/3D/b5/

לאחר שנוריד את כל ה-"/" לאחר "u.php" עד הסימן "_", נקבל את המחרוזת הבאה:

```
Oi8vd3d3LnluZXQuY28uaWwvaG9tZS8wLDczNDAsTC04LDAwLmh0bWw
```

כש"נפענח" את הכתובת הנ"ל, נגלה שהיא נותנת את ynet.

לכן, כדי לזהות פרוקסי מסוג CGIProxy, אנו נרצה להכניס פילטר מהסוג הבא (דרך grep או Snort):

```
/(/u\.php/).+/.+/.+(/b).+/
```

Glype

באופן דומה ל-PHPProxy, Glype מכיל מחרוזת שמשתמשת ב-browse.php. למשל, עבור האתר http://glype-proxy.info ואתר האינטרנט ynet, אנחנו נקבל את הכתובת הבאה:

```
http://www.glype-
proxy.info/browse.php?u=Oi8vd3d3LnluZXQuY28uaWwvaG9tZS8wLDczNDAsTC04LDAw
Lmh0bWw%3D&b=5
```

ונוכל לסנן אותה בעזרת המחרוזת הבאה:

```
(browse\.php\?u=).+(&b).*
```

ישנם עוד מספר רב של שרתי פרוקסי מבוססי Web בסיגנון זה, אך אני מאמין כי הבנתם את העקרון.

SSL Proxy

ישנם שרתים המאפשרים להצפין את העברת המידע בפועל, וכך הגילוי שלהם קשה עוד יותר. אלו שרתים שהונפקו עבורם רשיון SSL. חסרונם הוא ששרתים אלו בדר"כ משלמים על הנפקת הסרטיפיקטים ולכן יהיה קל לחסום אותם בעזרת רשימה שחורה במידה ונזהה חיבור דרכם.

מתרגמים

שימוש לא שגרתי כפרוקסי הוא המתרגמים (translators) למיניהם שנמצאים ברשת. המפורסם שבהם הוא http://translate.google.com המתרגם מילים, משפטים ואתרים שלמים משפה אחת לשפה אחרת ע"פ בחירה. מתרגמים נוספים נמצאים גם אצל Yahoo וכן אצל Microsoft.

לעתים, נרצה להשתמש במתרגמים לא רק בשביל לתרגם, אלא גם בשביל להסתיר את כתובת ה- IP שלנו, שכן נקבל את המידע המתורגם ישירות מהמתרגם, והאתר יופיע בתוך מסגרת בדומה לתבנית של PHPProxy.

נניח שנרצה לתרגם את ynet לאנגלית, אז ניגש לאתר ונכניס לתוך המסגרת את שם האתר, ונקבל את הכתובת הבאה:

http://translate.google.com/translate?sl=auto&tl=en&js=n&prev=_t&hl=en&ie=UTF-8&layout=2&eotf=1&u=ynet.co.il

נשים לב כי ynet.co.il מופיע בתור מחרוזת פשוטה. כמו-כן, "&sl" היא שפת המקור, "&tl" היא שפת היעד ו-"&ie" הוא ה-character set שלנו.

נוכל לחסום מחרוזת מן הצורה הזו אם נרצה לחסום את גישת המתרגם אלינו. במקרה כזו אנו עלולים לפגוע בלא מעט משתמשים תמימים שירצו להשתמש באתר שלנו, אך במידה ויש גישה נרחבת דרך המתרגם ויש יותר מדי נסיונות לגשת לדפים שלא קיימים, כדאי לשקול את אופציית החסימה.

זיהוי גישה דרך פרוקסי אל השרת שלנו

כשדפדפן מתחבר אל האתר הוא מזדהה באמצעות מחרוזת מסוג User Agent, המכילה את המידע על הדפדפן, גרסת הדפדפן, מערכת ההפעלה ועוד.

כדי לאפשר חיבור דרך פרוקסי, הפרוקסי צריך לתת מחרוזת כזו ייחודית השונה ממחרוזת ה- User Agent המקורית. דרך לנסות לגלות מי התחבר אל השרת שלך היא ע"י סריקה אחורה. למשל, אם התחברו אליך דרך translate.google.com (כדי לבצע תרגום של דף שנמצא באתר שלך), תוכל לתשאל את google.com איזו כתובת IP עשתה זאת.

סיכום

גילוי גישה משרתי פרוקסי היא משימה חשובה שתעזור לך לשמור על גישה "נקיה" יותר לשרת שלך ע"י סינון גישה אנונימית (שלא באמצעות ספק האינטרנט הישיר) שעלולה לפגוע בשרת. מניעת הגישה אפשרית בעזרת הכנסת מסננים מתאימים בעזרת grep או snort.

למרות שאין דרך אחת קלה לזהות שימוש שוטף בשרתי פרוקסי אנונימיים - ניתן לזהות דפוס פעילות שאותו ניתן לנטר ולסנן.

בתור התחלה ניתן להשתמש ברשימות שחורות המכילות מידע מוקדם על שרתי פרוקסי שכבר "הוכיחו" את עצמם כבעייתיים, וכן ניתן ואף רצוי להוסיף לרשימה זו את שרתי הפרוקסי שהיו אצלך. רשימה זו מתעדכנת כל הזמן.

בעזרת שימוש ב-grep נוכל לזהות מקרים כגון שימוש ב-Base64 ו/או Rotate13. זיהוי כזה יבטיח לנו גילוי השימוש בשרת פרוקסי אנונימי.

דוגמאות רבות המופיעות במאמר זה נלקחו מהמאמר שנכתב על ידי John Brozycki ופורסם במקור בשנת 2008 בחדר הקריאה של Sans.org - שווה לקרוא!

http://www.sans.org/reading_room/whitepapers/detection/detecting-preventing-anonymous-proxy-usage_32943