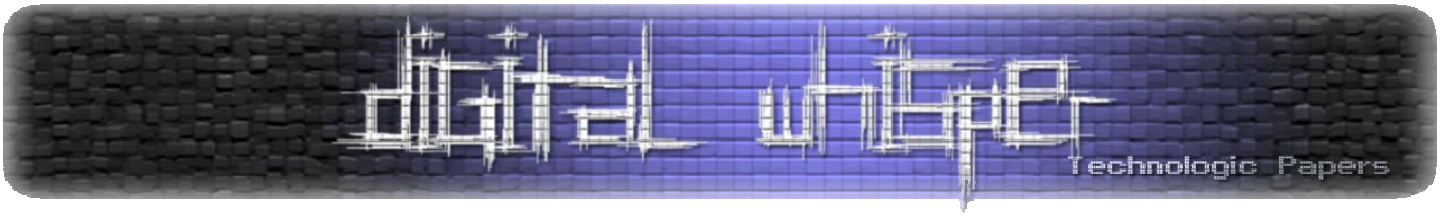


טכנולוגיות NAC

מאת רועי חורב

הקדמה

רוב ההשקעה באבטחת מידע בארגונים גדולים התמקדה מאז ומתמיד באיומים המגיעים מבחוץ לארגון. חומות אש הוקמו בכדי לתפקד כ"סלקטור" בכניסה לארגון, שרתי דואר הוקמו במיוחד על מנת למנוע כניסה של וירוסים ודברי ספאם, שרתי פרוקסי הוקמו על מנת למנוע גלישה לאתרים מזיקים ולא חסרות דוגמאות נוספות. בשנתיים האחרונות החלה לגבור המודעות של אותם ארגונים לאיומים הבאים מבית. הגברת המודעות מתבטאת בסגמנטציה של רשתות גדולות, מערכות endpoint-security מפוצצות וחומות-אש פנימיות.

אחת מהתובנות הגדולות שעלו עליהן במסגרת האיומים הפנימיים היא שכשאר אדם כלשהו נמצא פיזית בארגון, הוא יכול לחבר את המחשב הנייד שלו לכל נקודת רשת באשר היא. הדבר מהווה שני סיכונים גדולים מאוד- הראשון הוא שכל איש יכול להתחבר ל-LAN, להאזין לתעבורה, לגשת לשרתים, לספוג מידע ולזרוע הרס. הבעיה השנייה, לא פחות חמורה אך מקבלת פחות תשומת לב, היא שאנשים שמתחברים לרשת (אפילו אם במטרה טובה) יכולים להדביק את כל הרשת במזיקים שחיים להם על המחשב הנייד. סיפור מפורסם למדי מתאר בחור שעבד כיועץ אבטחת מידע ובשוגג חיבר את המחשב הנייד שלו לרשת פנימית של חיל האוויר, הנזק שנגרם הוא הידבקות של 20,000 מחשבי רשת חיל האוויר בוירוס.

על מנת להתמודד עם בעיות אלו הגיעו פתרונות ה NAC למיניהם אותם נסקור בכתבה הבאה. באופן כללי, מטרתם של פתרונות אלו היא למנוע גישה של גורמים זרים לרשת הארגונית. מכאן מגיע השם Network Access Control – בקרת גישה לרשת. המטרה היא למנוע כניסה של אנשים בעלי כוונות זדוניות אך גם מניעת גרימת נזק בשוגג על ידי אנשים שאינם מודעים למעשיהם.

ארגונים רבים מצפים מפתרון ה-NAC שלהם להרבה יותר מבדיקה מסורתית לגבי הרשאת ההתקן ורוצים בדיקות compliance מקיפות. בדיקות אלו, לדוגמא, יכולות לכלול וידוא כי המחשב הנייד שמתחבר יהיה בעל טלאים מסוימים של מערכת ההפעלה, שיהיה בעל חתימות בעלות תוקף מסוים באנטי-וירוס שלו, שיהיה שייך ל-Domain מסוים או דרישות אחרות התואמות את החלטות הארגון.

אז מה ניתן לעשות?

ישנן כמה דרכים להתמודד עם הבלגאן, במאמר זה אנסה לפרוס את מגוון הפתרונות הקיימים ולפרט קצת לגבי היתרונות והחסרונות של כל אחת מהארכיטקטורות.

802.1X

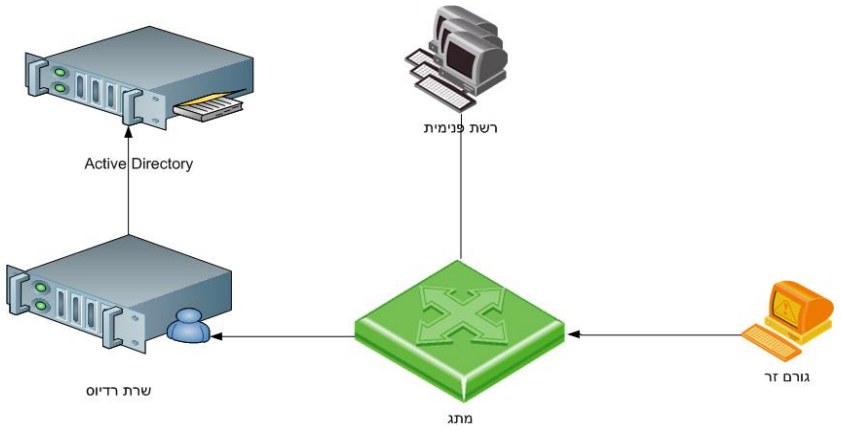
802.1X הוא פרוטוקול השייך למשפחת פרוטוקולי הרשת 802 של ה-IEEE (כלומר, הוא לא קשור ביצרן כזה או אחר וניתן ליישם אותו בכל סביבה שנרצה). ההיגיון העומד מאחורי הפרוטוקול הוא:

1. אדם מגיע בתור גורם זר לחברת מייקרוסופט ובזמן שהמארח שלו בחברה הלך לחוג יוגה השבועי שלו, אותו אדם מעוניין לקרוא מיילים. לצורך העניין, המבקר מחבר את המחשב הנייד שלו לנקודת הרשת הפנויה הראשונה שימצא.

2. המתג, בהיותו תומך בפרוטוקול, רואה שמישהו מנסה להתחבר ומעביר את הבקשה לשרת הרדיוס שחברת מייקרוסופט הקימה מבעוד מועד. רדיוס הינו פרוטוקול בפני עצמו שיודע לקבל צורת הזדהות מסוימת (למשל שם / סיסמא או תעודה חכמה) ולהחזיר תשובה – רשאי או אינו רשאי.

3.שרת הרדיוס בודק אם לאדם מאושרת הכניסה (למשל אל מול שרת Active Directory) ומחזיר את התשובה למתג.

4. המתג יודע שאם אדם כלשהו מאושר כניסה, עליו להעביר את הפורט עליו הוא מחובר ל-VLAN של הרשת הפנימית. במידה ואיננו רשאי (וזה כנראה המצב עבור מישהו שלא עובד במייקרוסופט), כנראה שהמתג יפנה אותו ל VLAN מבודד או שאפילו יכבה את הפורט עליו הוא יושב. תלוי באנשי ה-IT של חברת Microsoft.

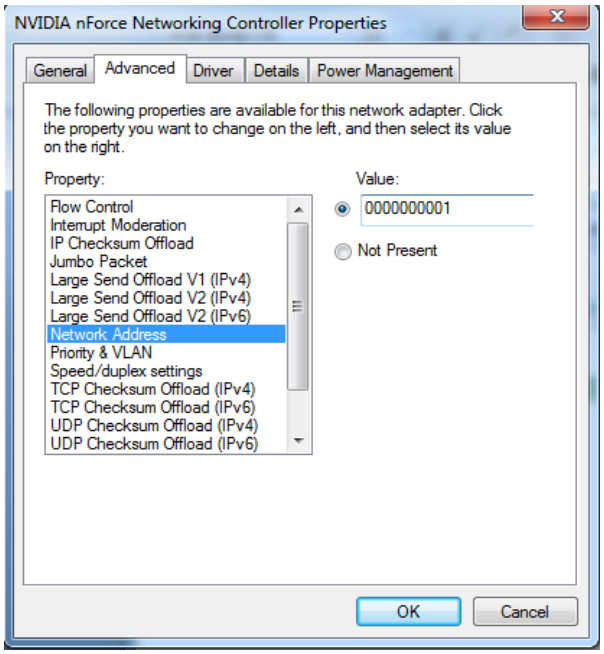


הפרוטוקול נותן מענה חזק מאוד מבחינת רמת האבטחה שהוא מספק. הבעיה העיקרית היא הקושי להטמיע מערכת שכזאת. נתחיל כראיה עם ארצנו הקטנטונת, שבדרך כלל היא חלוצה בתחומי

טכנולוגיות ה-IT. מספר הארגונים בארץ אשר ביצעו הטמעה מלאה של 802.1X מזערי עד אפסי. הקושי הגדול טמון בכך שצריך לוודא שכל ציודי התקשורת יתמכו ויכירו את הפרוטוקול, דבר המצריך ציוד חדיש יחסית ומערכות הפעלה חדשות... לאחר ששדרגנו את כל ציוד ההתקשרות, יש לוודא שכל התקני הקצה תומכים. נתחיל עם המחשבים, עליהם חייב להיות רכיב המסוגל לדבר את הפרוטוקול (supplicant בעגה המקצועית) ונמשיך הלאה למיליון התקנים אחרים שמחוברים לרשת וצריכים להיות מסוגלים לתקשר – טלפונים, מדפסות, שעוני נוכחות, מתגים וקוראי תגים. לבסוף, בלית ברירה, בנסיון להטמיע את המערכת, מדרדרים לאישור התקנים ע"פ כתובת ה-MAC שלהם, דבר הפותח פרצת אבטחה די גדולה – כי לשנות את כתובת ה-MAC של המחשב זה אינו סיפור גדול. רק בכדי להדגים כמה זה פשוט, תחת לינוקס אפשר להריץ את הפקודה הבאה:

```
ifconfig eth0 down hw ether 00:00:00:00:00:01
```

שמשנה את כתובת ה-MAC ל: 00:00:00:00:00:01 (גם תחת windows זה לא סיפור גדול) ברוב כרטיסי הרשת ניתן לשנות את הכתובות בהגדרות של כרטיס הרשת:



במקרים בהם כרטיס הרשת אינו תומך, אפשר לשלוט בכתובת ע"י ערך ב-registry:

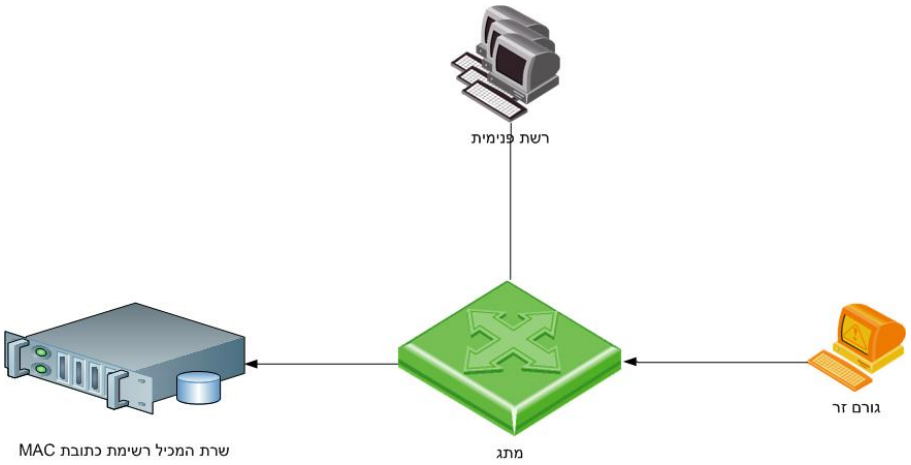
```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4D36E972-E325-11CE-BFC1-08002BE10318}
```

שם ישנו יצוג לכל כרטיס רשת לפי מספר סידורי והערך *NetworkAddress* אחראי על כתובת ה-mac. הפעלה מחדש של המחשב, וזהו – יש לנו כתובת MAC חדשה.

ניהול MAC

הפסקה האחרונה מובילה אותנו ישירות למערכות ניהול של MAC-ים למיניהם. אם כי לא בהקשר חיובי כל כך. ההיגיון מאחורי המערכות האלה אומר הוא כזה: נסתכל ברגע זה או בתקופה הקרובה על כל מי שמתחבר לרשת. לאחר תקופת הלמידה הזאת (בד"כ כשבועיים), נחליט שאת כל כתובת ה-MAC שצברנו נגדיר בתור ה"רשת" הארגונית שלנו, וכל כתובת חדשה שתצוץ בעתיד תחשב כפולש או מזיק ותחסם.

במצב כזה בעצם נקבל, בסופו של דבר, שרת אחד המכיל רשימה עצומה של כתובות MAC (רשימה שעלולה להכיל לעשרות אלפי כתובות בארגונים גדולים). בנוסף, כל התקן חדש שמגיע לארגון מחייב הכנסה של כתובת ה MAC שלו לרשימה – עבודה שלא נגמרת אף פעם. במידה ומגיע גורם לא מזוהה, הסינון מתבצע על ידי חסימת הפורט במתג. השרת שמוגדר לנהל את המתגים ב-SNMP מזהה כתובת MAC לא מאושרת (על ידי משיכת טבלת ה-MAC של המתג ב-SNMP), ושולח הוראה למתג לכבות את הפורט. מאותו רגע והלאה, אין לגורם הזר יכולת לבצע שום פעילות רשתית.



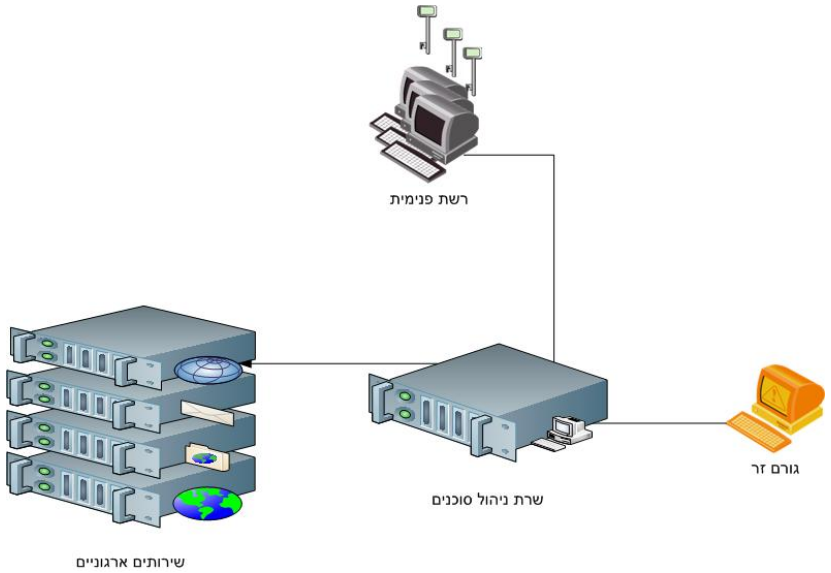
מצד אחד, המערכת מאוד קלה להקמה, וכמעט לא דורשת הגדרות מיוחדות ומצד שני התוצר שמתקבל היא רשימה שמאוד קשה לעקוב אחריה או לנהל אותה, וכמעט אף פעם לא נמחקות כתובות MAC שאינן בשימוש עוד.

אז מה צריך בשביל לעקוף מערכת שכזאת? לזייף MAC היא אחת האפשרויות

Agent Based NAC

לאור הטרנד החדש של חברות אבטחת המידע הגדולות בעולם לאחד את כל מוצרי הגנת תחנות הקצה שלהם לסוכן יחיד, גם יכולות אכיפת מדיניות נכנסות לסוכן זה. הפרטים הקטנים משתנים מיצרן ליצרן, אך התמונה הכוללת נשארת זהה. ברגע שיש סוכן על התחנה אפשר לבצע מספר רב מאוד של בדיקות. ואם המוצר משלב בתוכו חומת-אש אז בכלל מדובר ביתרון מכיוון שאפשר לשחק עם התקשורת לאחר הבדיקות. בדרך כלל בפתרונות האלו ישנו שרת, שאחראי על אימות התחנה ובדיקת הקריטריונים, ולא מתבצעת תקשורת ובדיקות אל מול המתגים עצמם.

לדוגמא – אני מעוניין שכל התחנות שלי בארגון ידברו עם שאר הרשת אך ורק במידה וחתימות האנטי-וירוס שלהם הן משלושת הימים האחרונים, זאת בכדי למנוע התפשטות של וירוס באמצעות תחנות לא מעודכנות. במידה והשרת שלי מזהה תחנה שהחתימות שלה בנות שבועיים, הוא מקבל מדיניות המורה לחומת-האש על התחנה עצמה ולמנוע תקשורת אל מול שאר הרשת. באותו אופן ניתן גם להגן על התחנה ו"לנעול" תקשורת מבחוץ לגבי כל גורם שלא עבר את הבדיקות שנקבעו מראש.



החלק החזק ביותר בשיטה זו הוא הגמישות שמתאפשרת ברמת הבדיקות על התחנות הנבדקות. הדבר מאפשר למנוע מצב של חוסר יכולת לבחון תמונת מצב ארגונית, כפי שקורה לעיתים די קרובות בארגונים. בנוסף, אם נתקלים בתקלה כלשהיא על התחנה, בדרך כלל בזכות הסוכן, אפשר לתקן את הבעיה.

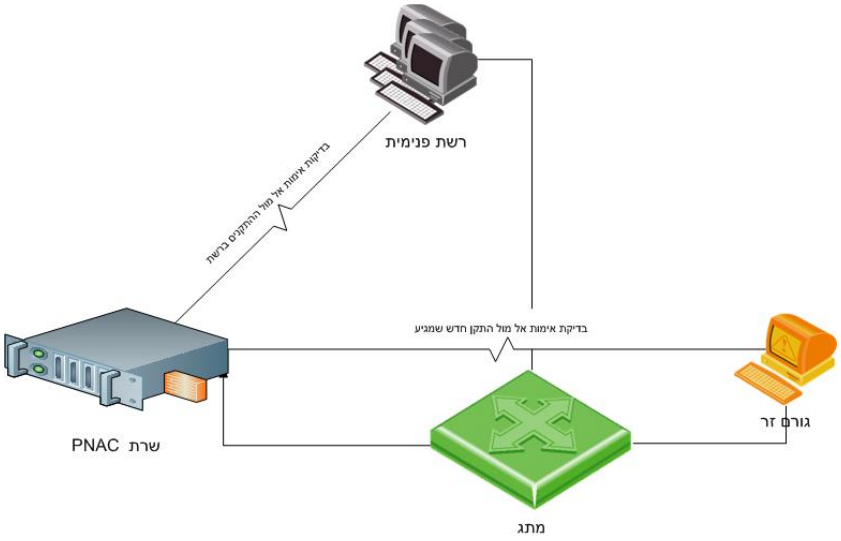
קיימים פטנטים רבים בכדי לדאוג שהתחנות אכן יעמדו בקריטריונים, כגון – מניעת גישה החוצה לאינטרנט, מניעת קבלת כתובת IP משרת ה-DHCP או השמה של מערכת הבדיקה בין התחנות לבין השרתים אליהם הם מנסים לגשת. החיסרון העצום בפתרון שכזה הוא שבעצם ההגנה היא על משאבי הרשת, ולא על הרשת עצמה. הפתרון הוא לא היקפי אלא תמיד נקודתי. אם השרת המאמת יושב לפני ה-DHCP, נוכל להכניס כתובת סטטית אך אם השרת יושב בדרך לאינטרנט, כל ה-LAN חשוף בפנינו.

בנוסף, כמו בחלק מהמערכות האחרות שנבחנו עד עכשיו, אנו נתקלים בבעיה עם טלפונים, מדפסות ושאר התקנים שלא ניתן להתקין עליהם סוכן. חשוב לציין כי ישנן מערכות כאלה שיודעות לשמש ב-supplicant ל-802.1x ובכך מתקבל שילוב מעניין בין שתי הטכנולוגיות.

Port NAC

הטכנולוגיה האחרונה שנשארה לנו מסתכלת על תמונת הרשת הכוללת ומנהלת את מדיניות הגישה ברמת הפורטים על המתגים. כך זה עובד:

מתקינים שרת שאחראי על ניהול הסימפוזיון. השרת יכול להאזין לתעבורת הרשת על ידי port-mirroring או באמצעות התממשקות ב-SNMP למתגים. לאחר שמתקבלת תמונת מצב של כל ההתקנים המחוברים ברשת, עלינו ללמד את השרת לתקשר עם ההתקנים. אם בפתרון שראינו של ניהול רשימות mac למדנו את הרשת ב-layer 2, כאן נתקדם ונעבור ל-layer 3 והלאה. המהלך נותן לנו את היכולת להיות חכמים יותר ולחסוך עבודה בעתיד כאר העבודה הנדרשת כאן היא בעצם להכיר לשרת את הרשת. אם תחנות ה-windows שייכות לדומיין מסוים, השרת יכול לרוץ ולבדוק את כל התחנות לשייכות לדומיין (על ידי WMI למשל). ברגע שהשרת זיהה שהתחנה שייכת לדומיין – מבחינתו היא מאושרת. גלומר, כל תחנה שתגיע בעתיד לרשת ואף היא שייכת לדומיין, תהיה מאושרת ללא צורך בהלבנה.



עוד דוגמא היא יכולת דיבור ב-SNMP אל מול טלפונים ומדפסות. ברגע שהתקן מסוים עונה לשרת ב-SNMP הארגוני, כנראה שהוא שייך לארגון ולכן ניתן לאשר גם אותו. חשוב לציין כי ניתן למצוא פתרונות אימות ל-90% מסוגי ההתקנים הקיימים ברוב הארגונים וכי אדם בעל ניסיון בתחום יכול למפות רשת ארגונית שכזאת בתקופה של כשבועיים.

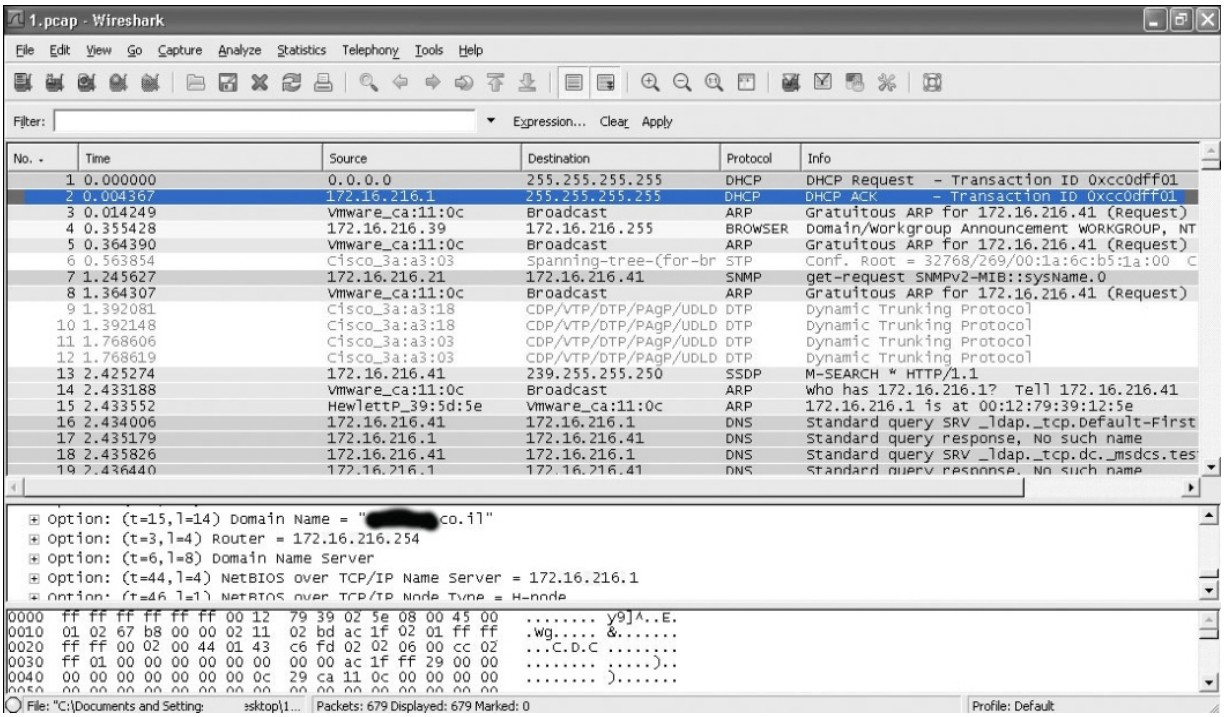
החסימה עצמה מתבצעת אף היא בתקשורת SNMP אל מול המתגים. החולשה של המערכת נובעת מכך שגם כאן, כאשר נתקלים בהתקנים מסוימים קיימת נטייה לפנות לרישום ה-MAC שלהם, ובעצם, כך גורמים ליצירת פריצת אבטחה.

מרבית טכנולוגיות ה-NAC שסקרנו כאן מענישות את הפורץ באחד משתי דרכים:

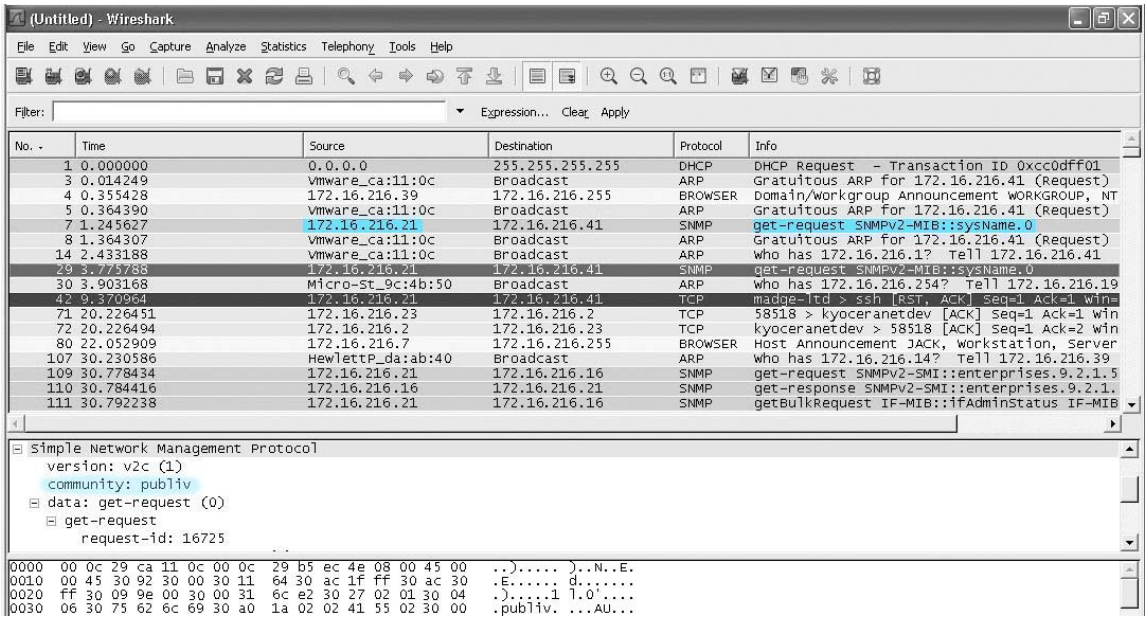
1. חסימת הפורט במתג – מצב בו כל התקשורת נפסקת ב-layer 1.
2. העברה ל-VLAN – העברת הפורט ל-VLAN אורחים, או VLAN בידוד בו הגישה לרשת מזערית או לא קיימת בכלל.

סיכום

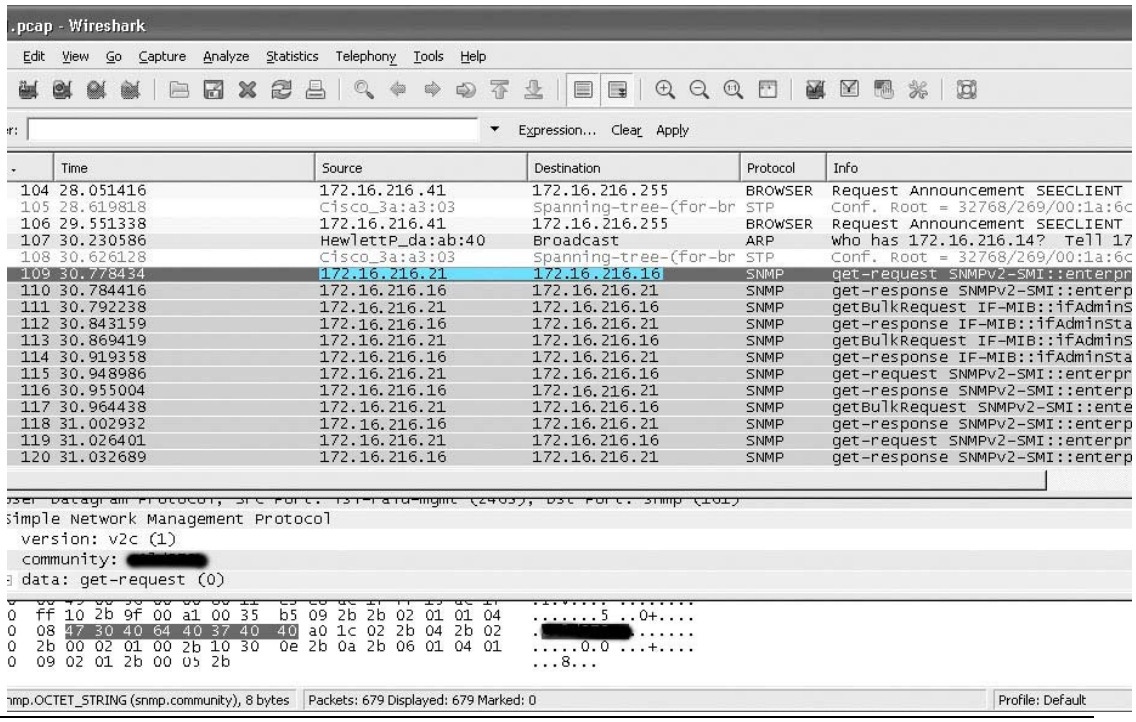
הרעיון וההיגיון מאחורי ה-NAC הוא יעיל והכרחי. הבעיה אצל מרבית הארגונים היא יישום הרעיון באופן מוצלח. לפורץ חיצוני עם קצת תושייה לא תהיה בעיה גדולה להתחבר ל-LAN, גם כשמדובר בארגונים שמשקיעים ומיישמים פתרונות NAC מורכבים ומריצים אלפי בדיקות לתחנות הארגוניות . כפי שהניסיון מראה – במרבית הארגונים מספיק לחבר את הנייד עם sniffer רץ, לזמן פעוט כדקה, בכדי לדעת איך הנייד צריך ל"היראות" פעם הבאה שהוא יתחבר לרשת. אם לדוגמא נחבר את הנייד בארגון בו קיימת מערכת אימות, הבה נראה איזה מידע אנחנו מספיקים לקבל בשניות הראשונות:

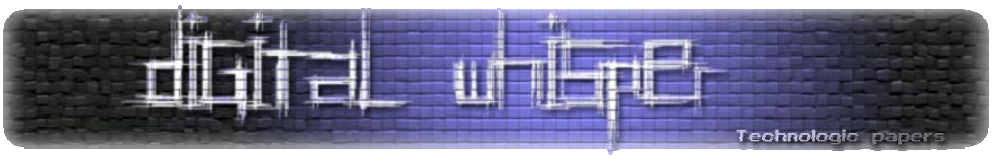


ה-packet השני שקיבלנו מה-DHCP עצמו חושף את שם הדומיין הארגוני (בחלק המחוק), כך שכבר אנחנו יודעים לאן עלינו להשתייך. הבה נראה איזה מידע מעניין אנחנו נקבל בהמשך:



כאן אנחנו רואים שכתובת IP מסוימת מנסה לדגום את התחנה שלנו ב-SNMP, מקבלים את הכתובת של השרת שמבצע את הבדיקות (172.16.216.21) וגם את סיסמת את ה-SNMP (community: publiv). אם נמשיך להאזין עוד ממש טיפה, נגלה מידע ממש מעניין:





כאן אנחנו רואים את שרת הבדיקות מדבר עם המתג עצמו וממש חושף את הסיסמא לניהול המתג. מנקודה זו והלאה נוכל לדבר ישירות עם המתג, לפתוח פורטים סגורים, למשוך טבלאות ניהול ולטייל בין ה-VLAN-ים השונים.

לא שבכל ארגון המצב יהיה זהה, אך תמיד נקבל מספיק פרטים בכדי להתקדם הלאה כבר ההאזנה ראשונית.

נסיים בציטוט של פו הדוב (שלא ניסה לפרוץ לשום מקום):

"אל תזלזל בערך של עשיית כלום, רק להמשיך, להאזין לדברים שאתה לא יכול לשמוע"