Security Issues of Contemporary Multimedia Implementations: The Case of Sonos and SonosNet

Stylianos P. Kavalaris[1] and Dr. Emmanouil Serrelis[2]
[1]Dixons South-East Europe SA, AMC Metropolitan College, [2]AMC Metropolitan College
74, Sorou st. 151 25, Amaroussio, Greece
skavalaris@amcstudent.edu.gr
eserrelis@metropolitan.edu.gr

ABSTRACT

Exploiting vulnerabilities on multimedia devices and implementations that are made for Small Office Home Office (SOHO) environments induce new risk trends. This happens due to usability and user friendliness taking precedence over the security on these implementations, but also due to market expansion and propagation that introduces a vast number of such devices to almost any SOHO network. This paper aims to identify, explore and analyse vulnerabilities and related attacks on such environments, focusing on specific devices manufactured by Sonos, as a proof-of-concept case study. The applied research has been done in such a way that can also be used for other SOHO devices. Real tests and attacks have been applied, in order to demonstrate and verify the proposed methodologies, as well as to evaluate all related results and findings. Finally, countermeasures and solutions for the identified vulnerabilities are proposed and evaluated.

KEYWORDS

SOHO Attack Tree, SOHO Multimedia devices, UPnP, Wi-Fi, WPA2-PSK, WPS Attack, Sonos, SonosNet.

1 INTRODUCTION

Traditional home networks contain computers, printers, networking devices as switches and routers as well as Network Attached Storages or other storage devices.

Over the last years this landscape has dramatically changed with the introduction and widespread use of new types of smart devices such as network media players, televisions, IP cameras and other network attached home equipment. These devices develop the new trend for modern home networks.

A lot of relevant studies and cases already exist (e.g. [1], [2], [3]), which prove that such devices have vulnerabilities that can be can be exploited using well-known attack methods and tools. These studies prove that the most serious security gap on such implementations is the wireless networking due to its nature [1]. It has been also proven that serious risks and flaws exist and can be exploited through the user interface [2], the UPnP architecture that is used on these devices [3], as well as through hybrid implementations and protocols that are based on web technologies [4].

SonosNet, is a proprietary network, which presents one of the dominant implementations for wireless music players using the wireless Wi-Fi of a SOHO network. Actually SonosNet is a proprietary extension of the Layer 2 in a Wi-Fi network [5], which allows the creation of a mesh, AES encrypted wireless network, in a way that all associated devices can communicate with the best possible connectivity to provide an uninterrupted communication service.

Since the aforementioned implementations are intended for SOHO users, the resources and tools that will be considered within this paper, will be narrowed down to resources and tools that can be found on a SOHO network, thus simulating the capabilities of a non-professional hacker that has access only to common hacking tools.

ISBN: 978-1-941968-03-1

2 ATTACK VECTORS

The various areas of Computer Science define several threat classification models. Criteria vary within different studies and focus on different threat characteristics and attributes, such as type, implementation method and resulting effect of the technology or medium used (e.g. [6], [7], [8]).

For the purposes of this paper, the methodology of an attack tree [9], which combines the effectiveness, the level of difficulty and the ease of carrying out an attack, is considered a suitable method to evaluate attacks on a given SOHO environment. A physical security resemblance would be that in order to crack a safe open, one could potentially tear it apart in two pieces. But this would not be the easiest or more efficient way [9]. To do so, someone must examine in every aspect, the effort that must be made, as well as the impact that this effort should result.

According to this methodology, the attack vectors in this paper will be distinguished in three dimensions. This renders the categorisation of the possible threats, as shown in Figure 1.

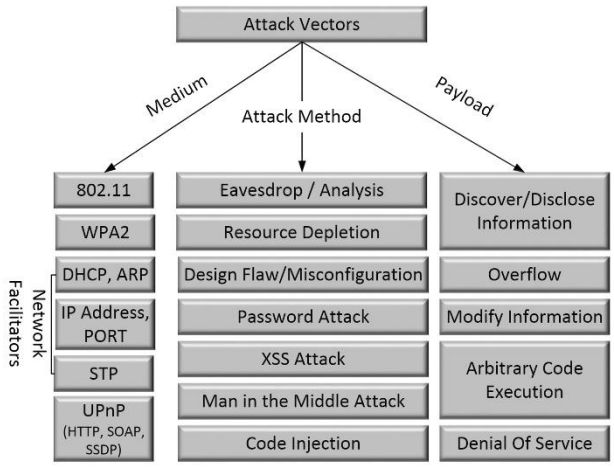

Figure 1. Attack vector classification

The above classification is based on existing and reported vulnerabilities (e.g. [10], [11], [12], [13], [14], [15], [16], [17], [18]), as well as the type of the vulnerability aspects and attack implementations that exist. This is actually the basis of this paper's methodology and research which also specifies the scope of potential research points to be covered. In order to illustrate these research points, Table 1 was created. This table maps medium and attack methods to a potential payload, thus defining a potential attack paths or heat map.

Table 1. Mapping medium, attack method and payload

			Network Facilitators				UPnP		
	802.11	WPA2 /WPS	DHCP	ARP	IP Ad. /Port	STP	HTTP	SOAP	SSDP
Eavesdrop/ Analysis		I			I				
Resource Depletion			D	D		D			
Design Flaw/ Misconfiguration	D	I	D	D		O	I,M,X,D	I,O	I,O
Password Attack		I					I		
XSS Attack							X	D	D
Man in the middle (MITM)		I,M		D,I					
Code Injection							D,X		O,X

Legend: I: Discover/Disclose Information M: Modify Information D: Denial of Service
X: Arbitrary Code Execution O: Overflow

2.1 Attack Vector Medium 1: 802.11

According to the standard Wi-Fi implementations, there are 11 different channels for the Americas and 13 for Europe. The operating frequency range is between 2.4 and 2.5GHz. An attack on the physical layer can be a simple signal jamming from other devices operating on the same frequency range, or specifically crafted electronic devices that block the signal, known as signal jammers. It must be noted that the latter are illegal on most regions. An intentional jamming can be executed though, using a computer system equipped with 802.11 hardware that has the ability send de-authentication packets to the intended target, using a packet injection attack via specific hardware and software.

ISBN: 978-1-941968-03-1 ©2014 SDIWC

2.2 Attack Vector Medium 2: Wireless Protection

It is clear that a successful connection to a wireless network makes it vulnerable to eavesdropping, data packet replay [19], as well as Denial of Service (DoS). Even if a successful connection cannot be established, wireless networks remain exposed to DoS attacks because of their connection medium that allows jamming attacks on the lowest OSI layer [20].
The initial efforts of creating a protection scheme for the wireless networks were proven substantially insufficient and problematic, since the WEP Protocol did not provide a secure enough implementation, as it used an RC4 encryption with weak keys that had insufficient length [21]. This allowed successful brute-force attacks and recovery of the encryption key within few hours or even minutes [22], packet forgery, as well as the decipherment or falsification of data packets, even without knowledge of the cryptographic key [23].
The next security implementation, which was the Wi-Fi Protected Access (WPA), was also proven to have security gaps regarding the Temporal Key Integrity Protocol (TKIP) under specific conditions [24]. Using a Man in The Middle attack, the exploitation was expanded further, resulting the packet forgery of small packets on any WPA implementation in a very short time [25]. Subsequently these attacks were perfected, thus rendering implementations that they use protocol TKIP vulnerable in attacks [26], ordaining the replacement of WPA and TKIP, with the WPA2 and CCMP that it is based in the strengthened AES encryption [27]. Still, even these implementations can be forced if they are not applied correctly [28], although they remain the safest alternative to date.

2.3 Using WPS as a Backdoor to Wireless Protection

The Wi-Fi Protected Setup (WPS) standard was added, from 2007 onwards, to almost all SOHO routers, in order to provide an easy way for simple and safe interconnection of devices, so as to encourage end-users to connect to wireless networks, using a form of encryption.
The WPS interface can be implemented in two ways: Pressing a physical or a virtual button on the device (PBC - Push Button Connect) or using a predetermined 8-digit identification code (i.e. PIN) [29].
This implementation is erroneous by design [16] and in order to promote usability, has made serious compromises on security. The design flaw that has been discovered, reduces the range of possible combinations, from 10^7 to $10^4 + 10^3$ (i.e. 11.000) [16]. Furthermore, physical exposure of the PBC can lead to exposure of the Wi-Fi password. As there are a lot of implementations using WPS that do not have any countermeasures to prevent a brute-force attack, WPS should be considered a serious threat in terms of compromising a SOHO network.

2.4 Attack Vector Medium 3: Networking Facilitations

Most SOHO devices use well known traditional network protocols in order to make installations easier for the end user and reduce manual configuration at a minimum level.
Following that principle, most of the modern home devices use DHCP to obtain a network address and connect to the network. By design, DHCP and ARP protocols (used in conjunction) do not bear any security mechanisms [30] rendering them vulnerable to certain attacks like Man-in-the-Middle (MiTM) [31], data collection and analysis [32] as well as resource depletion [33]. The payload of such attacks can cause a DoS, but it could also appear in the form of an injection of a rogue DHCP server [17]. These attacks can easily result a complete collapse of a SOHO network.
Most devices are also equipped with wireless and wired networking. On those cases, the Spanning Tree Protocol (STP) which is being used to prevent network loops, has inherent weaknesses by design, if not implemented

properly. As all of the above are interrelated and can be attacked sequentially, they can be grouped as one entity, although they might have a different threat level.

2.5 Attack Vector Medium 4: The UPnP Architecture

Consumers' need to interconnect different electronic appliances and devices in an easy and effective way, formed the need of creating the UPnP architecture which leverages common technologies and protocols for both networking and Web in order to achieve usable and flexible connectivity, control and data transfer among these devices [34].

However, due to the fact that these implementations present multiple vulnerabilities in various structural elements of UPnP, there are plenty of cases that these can be attacked resulting DoS [35], Stack overflows [36] and unexpected behaviour [37].

According to [18], security flaws in UPnP enabled network devices resulted the exposure of the SOAP API over the Internet and the exposure of more than 80 million devices via SSDP. Also known attacks on both HTTP and XML [38] prove that UPnP enabled appliances are potential exploitation points on a network.

Moreover, combining both UPnP and Wi-Fi potential threats, it can be deduced that quality control as well as extended testing of these home devices against known exploits and vulnerable components and implementations should be a mandate to ensure an acceptable level of security. Any compromise in the product design, in order to improve the usability of these devices must also be under great consideration and any vendor has to ensure that they do not bargain with the security of the device in order to create a product that is just "easier to setup and use".

3 SONOS and SONOSNET

The Sonos system is a wireless streaming system for the transmission of high fidelity sound (Hi Fi), as well as Internet radio or any digital music stored on a local area network (LAN) devices, supporting various sharing methods. The Sonos device interconnection is implemented via the proprietary Sonos network called SonosNet. [39]

SonosNet is a peer to peer, mesh network, using AES encryption for the communication between Sonos connected devices. SonosNet version 2.0 uses MIMO technology and a wireless network interface based on the 802.11n standard.

More specifically, the second level of TCP/IP has been customized to create a maximum number of wireless connections between Sonos devices, creating maximum coverage, given the positions of the devices. SonosNet requires a wired network bridge to the remaining network and for the connection with the rest of the devices [40].

Sonos uses STP to avoid network loops, DHCP to obtain address for the Sonos devices and UPnP to interconnect both with the rest of the Sonos devices and all other devices that exist on the network and can provide relevant services.

In order for a device to be associated with the SonosNet, either as an active device or as a controller, there is an authentication process that is similar to the WPS PBC.

Moreover, Sonos devices bear a console implementation for their control and use specific network ports for enabling specific services.

Finally, Sonos software (since version 3.7) allows Android based devices to be connected on the SonosNet by enabling a specific parameter.

3.1 The Attack Tree Against Sonos

As stated earlier, the scope of this research is to identify vulnerabilities of equipment that can be found on a typical SOHO network and that can be exploited using tools that are publicly available. Combining the issues raised in the previous paragraphs (i.e. known vulnerabilities

and attack methods, specific architecture, implementation and unabridged extend of the attack vector classification) with the attack tree theory explained as well as the assumptions made, it's been concluded that an attack tree that would show plausible vulnerabilities can have the form of Figure 2.

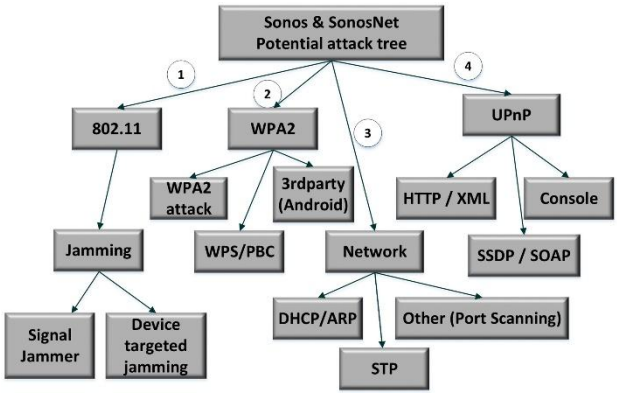

Figure 2. Sonos and SonosNet Attack Tree

The above attack tree could potentially have more branches as a result of using customised tools and equipment but this is considered to be out of the scope of this research.
The individual technologies used both in Sonos devices and the SonosNet are common to the majority of home multimedia devices. So, in order to simulate a real SOHO environment, common SOHO components were used together with Sonos devices. For the purposes of the attacking phase, resources that can be easily found and used were chosen, in order to simulate, as much as possible, the skills and effort needed for an actual attack to be performed to a SOHO network.

3.2 Attack Vector 1: Jamming the Sonos 802.11

While SonosNet is a proprietary 802.11n type wireless network, with mesh capabilities to prevent interference or drops, and signal jammers are illegal on most regions, thus interference can be emitted from numerous devices operating at the 2.4GHz band, such as wireless phones or AV wireless receivers. Furthermore, a home computer equipped with a wireless card, offering packet injection capabilities, can be used as a targeting jammer by sending de-authentication packets to an intended target. While SonosNet copes very well with those threats using its mesh networking capability, a successful DoS attack can be caused by targeting the Sonos Bridge device, i.e. the Sonos device that connects SonosNet to the SOHO router. From version 5.1 onwards, Sonos changed this process allowing alternatively direct connection to a wireless network. Previous topology still stands though.

3.3 Attack Vector 2: Attacking the Sonos WPA2-PSK

The level of difficulty in breaking in on a WPA2-PSK based Wi-Fi is that the creation of the PMK requires 4096 iterations of the SHA-1 algorithm, a time consuming process that practically prevents brute force attacks if used properly.
The use of the SSID on the process mentioned above ensures that the level of difficulty for finding a password does not only depend on the password length and complexity itself, but it also depends on the SSID. To accelerate these attacks, pre-computed tables – known as "rainbow tables" that contain values for pre-defined SSIDs and passwords [41] – are used. It can be expressed that the level of difficulty on finding a password (in this case the WPA2-PSK password) is a product of the probability P to find the correct password and the set of possible SSIDs:

$$\Delta = P(\text{wifi password}) * \Omega(\text{SSID}) \qquad (1)$$

Attacking the SonosNet WPA2-PSK using brute force techniques has been proven inefficient, as neither the SSIDs are common, nor the existing dictionaries made possible to reveal the password used by Sonos devices.

However, since SonosNet is available to Android based devices, a rooted Android device can be used to reveal that information. Rooting is a technique that allows end users to have access to the underlying Linux of an Android base device including all configuration files as well [42]. As shown in Figure 3, connecting such devices to the SonosNet can reveal both the Sonos hidden SSID and the WPA2-PSK password, simply by examining the correct configuration file.

```
ssid="Sonos_IQMXvxd4rbI7EIB84P35xVu
3iZ"
  scan_ssid=1
psk="7DE1D257A8CFD3807F367D8A9006
349E"
```

Figure 3. Revealing the Sonos WPA2-PSK password

Repeated tests have confirmed that Sonos WPA2-PSK passwords are always 32 characters long and consist of either digits or the capital letters A through F, thus representing some hexadecimal value. At this length, even with this very limited character set, it can be easily calculated that the Sonos WPA2-PSK password cannot be Brute-Forced. Nevertheless, infiltrating the SonosNet needs only a weak link, such as a rooted Android Device.

3.4 Attack Vector 2: Alternate Attacking to the Sonos Association Process

In order to connect a device to SonosNet, a certain process must be followed. This requires physical access to the device and pressing a button combination when prompted. In order to examine if this authentication process can be falsified using a replay attack (i.e. replaying the PBC packet sequence from an unauthorized device), the PBC transmission sequence has been recorded using a packet analyser. Analysis showed that this PBC packet sequence were UDP broadcasts. When replayed from an unauthorized source that had different network configuration from the original one (not the same IP or MAC address), the unauthorized device was successfully connected to the SonosNet, without having physical access to the device or using the authorized PBC process. This proves that there is a serious security gap within the Sonos association process. Proof of a successfully connected unauthorized device is shown in Figure 4.

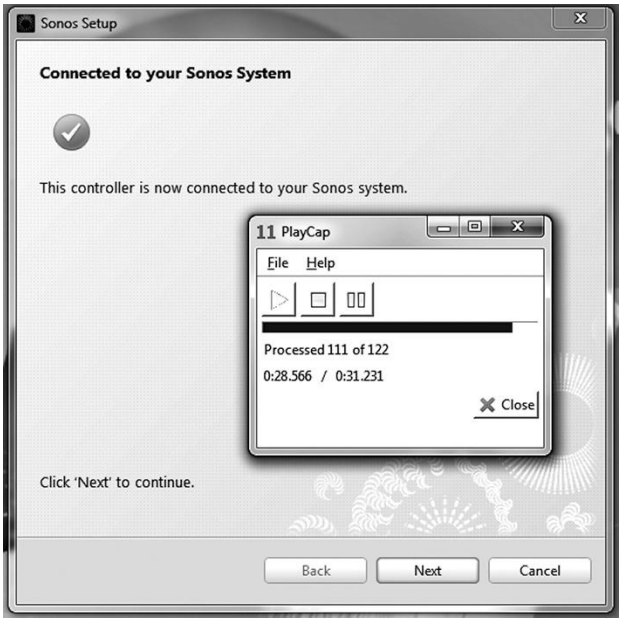

Figure 4. Connecting to Sonos using a repeat attack

3.5 Attack Vector 3: Successful Attacks to Network Facilitations

When a Sonos device is attached or restarted, it receives a network address via DHCP. Sonos' design philosophy seeks a simple installation for the end user, thus not offering a possibility to manually configure the network. By depleting all the available network addresses, Sonos devices remain functional until restarted or until they ask the DHCP server for an address renewal. In those cases, and within the starved DHCP Server, Sonos devices do not get a DHCP address corresponding to the SOHO network. Instead, they get a random automatic address (169.255.x.y) and they become unavailable. To achieve DHCP address

depletion, a workstation running Kali Linux was used, where the application "dhcpstarv" (i.e. a DHCP starvation utility) has been installed.
Furthermore a series of port scanning sessions were performed using the above mentioned equipment and the "nmap suite" with various program switches. Apart from the already advertised ports from Sonos for serving various services, port scanning revealed that TCP port 1400 is also open on Sonos devices. Further investigation revealed that this port is used for serving the http console.

3.6 Attack Vector 4: Attacking the Http Console

Checking the related paths and web pages through the use of a web browser or specialized tools (e.g. OWASP ZAP) to reveal website's structure, it has been revealed that the pages of the Sonos' Web Console have no authentication mechanism of any sort, thus they are freely accessible.
To obtain this information a workstation with OWASP ZAP (Zed Attack Project) application has been used.
It has been found that on those pages lie important information but also control mechanisms and commands. For example, by accessing the following path, the specific device restarts without the use of any credentials, so as for an authorized user to prove his permission:

```
http://<sonos_ip>:1400/reboot
```

A screenshot of the resulted behaviour is shown in Figure 5.

Figure 5. Rebooting a Sonos device without authorization

Additional pages were also revealed during the research, including:

```
http://<sonos_ip>:1400/status
http://<sonos_ip>:1400/support/review
http://<sonos_ip>:1400/advconfig.htm
http://<sonos_ip>:1400/unlock.htm
```

The first link provides a tree of 66 other links that reveal a great deal of important information regarding the device's properties, configuration as well as the underlying embedded Linux that the Sonos System relies to. The second link contains another 60 links that can execute specific commands on the underlying Linux, check the contents of log files and also reveal more information not only for a specific Sonos device but for all of the Sonos devices that are part of the specific SonosNet implementation.
The unprotected console that has been revealed leaves further room for traditional Web Attacks like Cross Site Scripting (XSS). By modifying the XML code of an existing page it is possible to execute arbitrary code on the Sonos' underlying Linux, as shown in Figure 6.

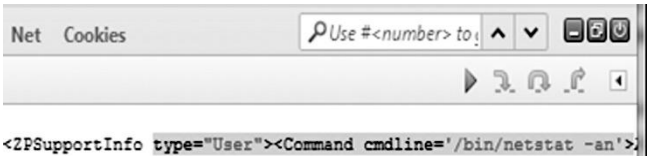

Figure 6. Modifying an existing xml to execute arbitrary code.

Furthermore one of the revealed links gives the exact location of the most recent firmware file for each type of the Sonos family. This link is:

```
http://<sonos_ip>:1400/status/jffs/
upgrade.log
```

As there is no protection of any kind even for this procedure, an attacker can potentially download the device's firmware, alter it so that it can include malicious code or a backdoor and then infect a device with the altered firmware using a MiTM attack and pretending that a

rogue host is the official Sonos firmware update page.

3.8 Not Everything is Vulnerable

Although it has been described earlier that ARP and STP protocols are a potential risk on all devices using them, this is not the case for Sonos devices. Several known attacks were proved inefficient on the SonosNet. During the related research, a workstation running Kali Linux, was used to perform an ARP poisoning by using the "Nemesis" application (i.e. a Packet Crafting/Injection tool). A similar MiTM based attack was tried in order to cause continuous root bridge elections on STP, using the "Yersinia" application (i.e. a framework for performing layer 2 attacks).
Apart from the networking part, the UPnP implementation was proved non-exploitable since it is based on library versions that are considered safe and do not have identified vulnerabilities on both SOAP and SSDP. Even though this framework was considered as one of the areas of potential risks for SOHO and multimedia devices, extensive tests with "Metasploit", as well as the specialized "ScanUPnPnow" utility that reveals vulnerable implementations, proved that this is not the case for Sonos devices.

4 OVERCOMING THE ISSUES

Vulnerabilities on devices intended for home use, given their volume and spread, should not be designed solely to perform functions that are visible to users. Device manufacturers will have to adapt their product development approach and provide both functional and secure solutions, since end users are not intended to have specialized equipment and security mechanisms to protect their SOHO network.
Although this was not the case for Sonos and SonosNet, UPnP implementations as well as the network protocols that facilitate most contemporary multimedia and smart home devices should be checked thoroughly by the manufacturers for known vulnerabilities and related attack methods. The vast numbers of flawed configurations or poor implementations confirm the latter.
On the following paragraphs, proposals to manage some of the identified problems are made. The selection criteria for these proposals remain the same as the ones that Sonos itself has already put. They must neither change the user experience and simplicity of use, nor affect significantly the production cost of the devices. It should be noted that not all issues can be solved through those parameters, nevertheless an improvement on these security issues could can be achieved.

4.1 The DHCP Problem

It has been found that Sonos devices can receive a network address only via DHCP protocol and demonstrated that this can lead to a DoS attack that can be implemented with a quite low difficulty level. This can be resolved by making address leases to the SOHO router. Still, having such an option is not always the case, despite the fact that this option is more difficult than configuring a manual network address on a device interface. In order to maintain simplicity and ease of setup, a proposal for developing an automated mechanism that discovers free static addresses on a SOHO network and assigns them to the devices is proposed.
Sonos could also develop a software tool that will be compatible with the operating systems and platforms that they already support (Windows, iOS, Android), which gives the device a static network address, according to the algorithm shown in Figure 7.

ISBN: 978-1-941968-03-1 ©2014 SDIWC

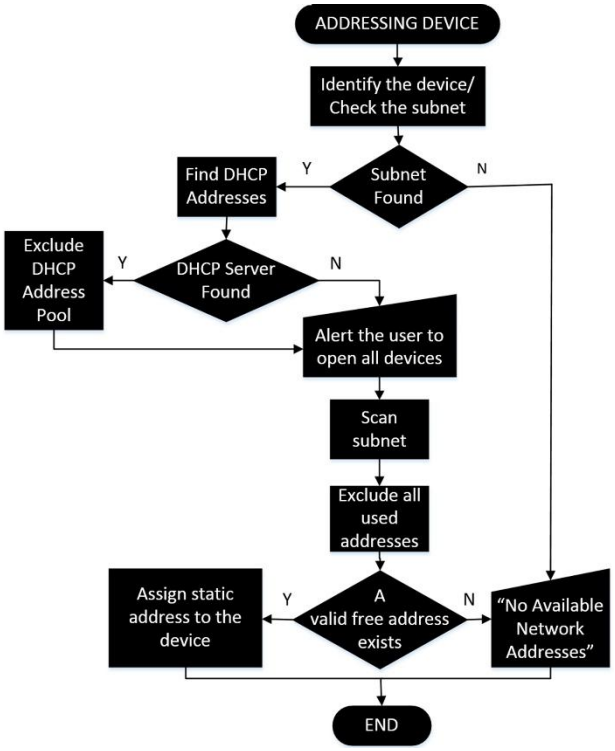

Figure 7. Addressing a device algorithm

4.2 Securing the HTTP Console

It has been found that Sonos devices have a "hidden" website that acts as a management console and can be accessed through TCP port 1400. While any of these information has not been found published or mentioned in any accompanying documentation or any other official channel, it appears that through this console, critical configuration changes can be performed on the device and a great deal of information can be revealed. This fact leads to the identification of a series of vulnerabilities, such as those identified and proved. This is actually one of the most vulnerable points on Sonos devices, since the complete lack of any form of access control makes them vulnerable to any other device connected to the network and has just a web browser.

Therefore, it is suggested that the http console should be secured with an authentication process and any Sonos program or device control process that uses the console, must be suitably modified. As a mandatory security control, the web communication must be switched to using the https version and an authentication system, transparent to the end user, must be developed for the authorized and paired control devices.

This could include an authentication process using a token created from the device itself in a unique way. The access level should also be segregated to user level that could only perform end-user tasks, as well as administrator level that could also modify any configuration parameters.

Furthermore all the webpages that can be potentially disclose critical information, should be suitably modified to have security controls that can prevent commonly used attack methods such as XSS and code injection, as presented on this paper.

All above issues are not opposed in any way to the vendor's user-centric philosophy aiming for simplicity and usability for the consumers. Nevertheless, they should also drive an extensive and solid reengineering and development, so that the devices will remain equally user-friendly but also secured at a reasonable level.

4.3 Propositions Considering Other Identified Vulnerabilities

It has been also become apparent that the Sonos Association Process has significant design flaws that make it exploitable. This model can change on the software level, so that the packets transmitted within the Sonos environment, could determine the transmitting device in a reliable manner. Altering this mechanism to something similar to the WPS could be more secure, but still having some limitations. However this approach should not introduce complexity to the users' experience mechanism and can generally remain disabled, unless selected by the entitled end-user. If the manufacturer also chooses to provide a process delay after a number of failed attempts (lockout), together with a de-activation mechanism when not needed, it should narrow significantly an attacker's time frame.

Considering the interconnection issues with third party devices, the Sonos software could be improved by adding some checks, so that it will not be possible for altered devices such as "rooted" android-based devices to be connected to the SonosNet.

5 CONCLUSIONS

Throughout the previous sections of this paper significant security flaws were presented and successful attacks were demonstrated confirming the discussed vulnerabilities related to a very popular multimedia SOHO solution, i.e. Sonos and SonosNet. From these results it became evident that vulnerabilities can arise not only from erroneous designs or omissions in protocols and technologies being used, but also from poor implementations.

The Sonos approach for oversimplification of installation and maximization of usability creates a significant impact on the security of the devices, the related network as well as any data communicated there.

The main issue with the Sonos devices is that they can be a permanent point of intrusion on a SOHO network, even if all other potential vulnerabilities are covered. Once SonosNet allows access to other devices, the problem becomes much larger, since any form of data can potentially use it as a medium.

Since an attacker is able to choose among different vulnerabilities, the ones that are less demanding in resources, can be implemented easily and have the greatest impact, would be preferred. Under that assumption the vulnerabilities that should be addressed with high priority and criticality are: Secure the http console, redesign the association procedure and change or better yet remove the feature of connecting rooted Android based devices to the SonosNet.

This paper aimed to make clear that the level of security of the SOHO multimedia and smart devices and appliances networking should be increased. Manufacturers must use information security techniques and technologies already available and operating at other information-centric sectors (e.g. Finance, Trading). Additional techniques and technologies to be considered, are related to quality controls of their products. Manufacturers must also upgrade their service standards, as well as the control and security levels, so that required actions to resolve specific security issues must be completed within well specified timeframes.

6 REFERENCES

[1] L. Zhou, , D.Wu, B. Zheng and M.Guizani, "Joint Physical-Application Layer Security for Wireless Multimedia Delivery" IEEE Communications Magazine, Vol. 52, Issue 3, pp. 66-72, March 2014.

[2] S. Lee, and S. Kim, "Smart TV Security - #1984 in 21st century", CIST(Center for Information Security Technologies), Korea University, March 08, 2013.

[3] A.Hemel, "Universal Plug and Play: Dead simple or simply deadly?", upnp-hacks.org, April 07, 2006.

[4] Y. Oren and A. D. Keromytis, "From the Aether to the Ethernet – Attacking the Internet using Broadcast Digital Television", Columbia University, May 19, 2014.

[5] N. A. Millington and P.V. Hainsworth, USA patent No US8326951 B1, Dec 4, 2012.

[6] W. Ahmad, Z. Hayat, B. Zafar, F.A. Khan, F.U. Din and I. Shah, "A Survey on Taxonomies of Attacks and Vulnerabilities in Computer Systems", International Journal of Computer Science and Telecommunications Volume 3, Issue 5, pp.93-97, May 2012.

[7] N. Yoshioka, H. Washizaki and K. Maruyama, "A survey on security patterns", Progress in Informatics, vol. 5, issue 5, pp. 35-47, October 2008.

[8] C. Simmons, C. Ellis, S. Shiva, D. Dasgupta and Q. Wu, "AVOIDIT: A Cyber Attack Taxonomy", Technical Report: CS-09-003, University of Memphis, August 2009.

[9] B. Schneier, "Ch. 21 - Attack Trees in Secrets & Lies: Digital security in a networked world", Indianapolis, USA: Wiley Publishing Inc, ISBN: 978-0-471-45380-2, pp. 318-333, January 2004.

[10] F. T. Sheldon, J. M. Weber, S. Yoo and W. D. Pan, "The Insecurity of Wireless Networks", IEEE Security and Privacy Magazine, Vol.10, Issue 4, pp. 54-61, August 2012.

[11] C. Heffner, D. Yap "Security Vulnerabilities in SOHO Routers", http://www.exploit-db.com/wp-

content/themes/exploit/docs/252.pdf, September 2009.

[12] B. Konigsberg, "Auditing Inside the Enterprise via Port Scanning & Related Tools", SANS Institute InfoSec Reading Room, http://www.sans.org/ reading-room/whitepapers/auditing/auditing-enterprise-port-scanning-related-tools-75, 2002.

[13] L. Davi, A. Dmitrienko, A.R. Sadeghi, and M. Winandy, "Privilege Escalation Attacks on Android in Information Security", pp. 346-360, Springer Berlin Heidelberg, 2011.

[14] G. Ou, "How to jam your neighbor's Wi-Fi legally" , ZDNet, http://www.zdnet.com/blog/ou/how-to-jam-your-neighbors-wi-fi-legally/247, June 15, 2006.

[15] L. A. Trejo, R. Monroy and R. Lopez Monsalvo, "Spanning Tree Protocol and Ethernet PAUSE Frames DDoS Attacks: Their Efficient Mitigation", Technical report, Department of Computer Science, Tecnológico de Monterrey, Campus Estado de México, Carr. Lago de Guadalupe, Km. 3.5, Estado de México, 52926, Mexico, 2008

[16] S. Viehböck, "Brute forcing Wi-Fi Protected Setup Version 3", Wi-Fi Protected Setup, December 26, 2011

[17] Y. Bhaiji, "Understanding, Preventing, and Defending Against Layer 2 Attacks", Cisco, http:// www. nanog. org/meetings/nanog42/presentations/ Bhaiji_Layer_2_Attacks. Pdf, 2007.

[18] H. Moore, "Security Flaws in Universal Plug and Play: Unplug. don't play.", Rapid7 Ltd., January 2013.

[19] Z. Feng, J. Ning, I. Broustis, K. Pelechrinis, S. V. Krishnamurthy and M. Faloutsos, "Coping with packet replay attacks in wireless networks", 8th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, pp. 368-376, June 2011.

[20] D. R. Raymond and S. F. Midkiff, "Denial-of-Service in Wireless Sensor Networks: Attacks and Defenses", IEEE Pervasive Computing, Vol. 7 , Issue 1, pp.74-81, January 2008.

[21] N. Borisov, I. Goldberg and D. Wagner, "Intercepting Mobile Communications: The Insecurity of 802.11", Proceedings of the 7th annual international conference on Mobile computing and networking, New York: ACM, pp. 180-189, 2001.

[22] G. Lehembre, "Wi-Fi security – wep, wpa and wpa2", hakin9, January 2006.

[23] A.H. Lashkari, M.M.S. Danesh and B. Samandi "A Survey on Wireless Security protocols (WEP,WPA and WPA2/802.11i)", Computer Science and Information Technology, ICCSIT 2009. 2nd IEEE International Conference on, Beijing: IEEE, pp. 48-52, August 2009.

[24] E. Tews and M. Beck, "Practical attacks against WEP and WPA", Proceedings of the second ACM conference on Wireless network security, ACM, pp. 79-86, March 2009.

[25] T. Ohigashi and M. Morii, "A Practical Message Falsification Attack on WPA (white paper)", http:// packetstorm.wowhacker.com/papers/wireless/A_Practical_Message_Falsification_Attack_On_WPA.pdf, August 26, 2009.

[26] M. Vanhoef and F. Piessens, "Practical verification of WPA-TKIP vulnerabilities", Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security, Hangzhou China, ACM, pp. 427-436, May 2013.

[27] M. Bellare, J. Killian and P. Rogaway, "The Security of the Cipher Block Chaining Message Authentication Code", Journal of Computer and System Sciences, Volume 61, Issue 3, pp. 362-399, December 2000.

[28] A. Tsitroulis, D. Lampoudis and E. Tsekleves, "Exposing WPA2 security protocol vulnerabilities", International Journal of Information and Computer Security Volume 6 Issue 1, pp. 93-107, March 2014.

[29] B. Slavin, "Wi-Fi Security – The Rise and Fall of WPS", http://www.netstumbler.com/2013/01/18/wi-fi-security-the-rise-and-fall-of-wps/, Netstumbler. com, January 2013.

[30] M. Patrick, "DHCP Relay Agent Information Option", RFC 3046, January 2001.

[31] S. McClure, J. Scambray and G. Kurtz, "Chapter 7: Network devices - Hacking Exposed 6th edition", pp. 388-412, McGraw-Hill, 2009.

[32] J. Cache, J. Wright and V. Liu, "Chapter 5 : Attack 802.11 wireless clients - Hacking Exposed Wireless 2nd edition", pp. 155-201, McGraw-Hill, 2010.

[33] K. Lauerman, and J. King, "DHCP Consumption Attack and Mitigation Techniques (white paper)", http://www.cisco.com/c/en/us/products/collateral/switches/catalyst-6500-series-switches/white_Paper_ C11_603833.html, Cisco Press, 2010.

[34] UPnP Forum. "UPnP™ Device Architecture 1.1", http://upnp.org/specs/arch/UPnP-arch-DeviceArchite cture-v1.1.pdf, Contributing Members of the UPnP Forum, October 15, 2008.

[35] A. Hemel, "Universal Plug and Play: Dead simple or simply deadly? (white paper)", http://www.upnp-hacks.org/fosdem2008-presentation.pdf,upnp-hacks .org, February 24, 2008.

[36] J. Barnaby, "Exploiting Embedded Systems. Blackhat Conference", Amsterdam: eEye Digital Security, March 2006.

[37] J. Squire, "Universal Plug and Play IGD - A Fox in the Hen House (white paper)", https://www. blackhat.com/presentations/bh-usa-08/Squire/BH_US_08_Squire_A_Fox_in_the_Hen_House%20White%20Paper.pdf, Black Hat USA 2008 Archives, August 8, 2008.

[38] A. Chuvakin and C. Peikari, "Chapter 15. SOAP XML Web Services Security - Security Warrior", O'Reilly, pp 228-233, January 2004.

[39] Sonos, "SONOS System Setup Guide" Sonos Inc, 2014.

[40] S. Beckhardt, H. Goassain, N.A. Millington and J. M. Peters "US Patent No 0336499 A1", December 2013.

[41] P. Oechslin, "Making a Faster Cryptanalytic Time-Memory Trade-Off", Advances in Cryptology-CRYPTO 2003, Springer Berlin Heidelberg, pp. 617-630, 2003.

T. Vidas, D. Votipka and N. Christin, "All Your Droid Are Belong To Us: A Survey of Current Android Attacks", Proceedings of the 5th USENIX Workshop on Offensive Technologies, pp. 81-90, August 2011.