The weakest link on the network: exploiting ADSL routers to perform cyber-attacks

Anastasios Stasinopoulos, Christoforos Ntantogian, Christos Xenakis
Department of Digital Systems, University of Piraeus
Piraeus, Greece
{stasinopoulos,dadoyan,xenakis}@unipi.gr

***Abstract*— *ADSL routers are an integral part of today's home and small office networks. Typically, these devices are provided by a user's ISP and are, usually, managed by people who do not have any special technical knowledge. Often poorly configured and vulnerable, such devices are an easy target for network-based attacks, allowing cyber-criminals to quickly and easily gain control over a network. In this paper, we systematically evaluate the security of ADSL routers and identify the potential of attacks, which attempt to compromise the vulnerabilities of their web interface. More specifically, we present common vulnerabilities and attacks that occur in websites on the Internet, and project them on the special characteristics of the web management interface of ADSL routers. To put this analysis into a practical context, we investigate the security of a popular ADSL router provided by a Greek ISP. In this security assessment, we have discovered two 0-day vulnerabilities in the web management interface of the tested router. In particular, we discovered an operating system (OS) command injection and stored Cross-Site Scripting (XSS) attack. A malicious may exploit these vulnerabilities to perform several large-scale attacks. Specifically, he/she can perform DNS hijacking attack and redirect the users to fake web sites for phishing; mount a Distributed Denial of Service (DDoS) attack using the compromised routers as zombie machines; or even spread a malware. Finally, we discuss some well-known security practices that should be followed from developers and users to enhance the security of ADSL routers.*

Keywords*—ADSL routers, web interface vulnerabilities, command injection, XSS, DNS hijacking, phishing.*

I. INTRODUCTION

In recent years, there has been a significant increase in broadband Internet access in several countries, including Greece. According to the Hellenic Telecommunications and Post Commissions, the penetration of broadband Internet access in June 2012 has reached 2.560.414 subscribers (22.6% of the total population of Greece) [1]. There are seven major Internet service providers (ISPs) operating in Greece, such as OTE Conn-X, Forthnet, Hellas On Line, WIND Hellas, Cyta hellas, On Telecoms and Vodafone Hellas. It is a common practice among ISPs, to provide to each new subscriber an ADSL router for free. This router is the most critical part in a Small Office / Home Office (SoHo) network, since it controls the traffic flow between the Internet and the internal network. ADSL routers include a web-based administration interface, allowing network setup and configuration. This interface is often written using a combination of HTML, Javascript, PHP and Perl-CGI (Common Gateway Interface) programming languages and can be accessible through a log-in process.

Although this web interface is an effective solution from a usability point of view, it is an Achilles' heel in terms of security and the overall system's robustness. Having acknowledged this, some individual security activists have investigated the security of many popular ADSL routers, and proceeded in publishing their finding as well as the discovered vulnerabilities, mainly, in a form of exploits [3] [4] [5] [6] [7]. Moreover, a collective study of vulnerabilities, discovered in the web interface of embedded devices, was presented in [13]. This satisfies the objectives of activists and may facilitate the manufactures and ISPs to patch the security holes of their products; but it is not fruitful in understanding the reasons why these weaknesses continue to occur, and what will be the impact of a possible attack that exploit such vulnerabilities. This is the motivation of this paper, which attempts to fill the above mentioned gap in an holistic and technically sound approach by: (i) discussing the basic weaknesses that occur at ADSL routers; (ii) presenting a manual methodology (not an automated one) that we have followed to investigate the security of an ADSL router; (iii) analyzing the impact of the discovered vulnerabilities; and (iv) presenting some well-known security practices that should be followed during routers implementation and usage.

In general, the security holes of ADSL routers are attributed due to the poorly written software of these devices. It also seems that the ISPs are not aware of the security impacts and harm that these vulnerabilities may cause. Recently, it was discovered that several Brazilian ISPs have fallen victims of a series of domain name system (DNS) hijacking attacks, which compromised a whopping number of 4.5 million ADSL routers [2]. The attack exploited a security hole in the routers' web interface, which allowed a Cross Site Request Forgery (CSRF) to be performed in their administration panel, allowing the attacker to make changes in the DNS servers. Once compromised, users were redirected to specially crafted phishing domains that mainly targeted users' online banking credentials. Another security issue of the ADSL routers has to do with the fact that once installed and configured, end-users will probably overlook to update their firmware, since it is a manual and tedious process. Moreover, many home users simply do not have any technical knowledge to perform software updates in the ADSL routers. Thus, even if the vendor of a vulnerable router releases, eventually, a security patch, it will not be applied in many devices.

In this paper, we systematically evaluate the security of ADSL routers and identify the potential of attacks, which attempt to compromise the vulnerabilities of their web interface. More specifically, we present common vulnerabilities and attacks that occur in websites on the Internet, and project them on the special characteristics of the

web management interface of ADSL routers. We pinpoint that the majority of the presented attacks can be attributed due to the fact that developers fail to properly validate the input provided by end-users. Moreover, to put this analysis into a practical context, we investigate the security of a popular ADSL router provided by a Greek ISP as a case study. In this security assessment, we have discovered two 0-day vulnerabilities in the web management interface of the tested router. In particular, we discovered that it is vulnerable to Operating System (OS) command injection and stored Cross-Site Scripting (XSS) attacks. A malicious may exploit these vulnerabilities to perform several large-scale attacks. Specifically, he/she can perform DNS hijacking attack and redirect the users to fake web sites for phishing; mount a Distributed Denial of Service (DDoS) attack using the compromised routers as zombie machines; or even spread a malware. Finally, we discuss some well-known security practices that should be followed from developers and users to enhance the security of ADSL routers. The work in this paper should also be viewed as a warning to end-users and ISPs of the possible attacks that can be performed in case that a device, even of limited functionality and capabilities such as a SOHO router, has been compromised.

The rest of the paper is organized as follows. Section 2 presents the attacks found in ADSL routers, categorized as server-side and client-side. Section 3 aims at investigating the security of a specific ADSL router, using a manual methodology. Section 4 analyzes the impact of the discovered vulnerabilities and provides some generic security practices that should be followed by developers and end-users. Finally, section 5 concludes the paper.

II. Attacks to ADSL Routers

The attacks that could occur in ADSL routers are divided into two categories: server-side and client-side attacks. Server side attacks target the router aiming at altering its normal behavior or disclosing sensitive information, such as root passwords. On the other hand, client-side attacks target an unsuspecting user connected to the ADSL router. This type of attacks is triggered by an end-user action using a browser, which interacts with a compromised or malicious web site, forcing it to execute malicious code or process data.

A. Server-Side Attacks

1) Authentication bypass.

Through authentication bypass attacks, an attacker is able to perform administrative changes without possessing the administrator password. A successful authentication bypass entails disclosure of sensitive information and allows execution of arbitrary commands with administrative privileges. Most of the security holes found on the web interface of embedded devices, which may lead to authentication bypass, fall into one of the following categories:

- Multiple representation of valid URLs
- Knowledge of “post-authentication” URLs
- Unchecked HTTP methods
- Unprotected cgi scripts

In the first category, an attacker tries to find alternative ways to represent a URL that would grant access to administration functionality. That is, there are multiple ways that a URL can be represented, where it is still valid by the web application, without the latter requesting from the user to enter the username and password. This happens because developers do not filter and validate properly the URLs that a web application receives. Therefore, the web application may accept multiple representations of the same URL as valid, without requiring authentication. For example, assume that the URL for accessing the firewall settings is: http://homehub/cgi/b/firewall/ and requires an admin account username and password. The attacker can try the following alternatives for the above URL that may provide access to the firewall settings, without however requiring the administrator password.

- http://homehub/cgi/b/firewall/%5C
- http://homehub/cgi/b/firewall//
- http://homehub/cgi/b/firewall/~

In the second category, the attacker exploits the following functionality common in many routers: when accessing a web interface of an appliance, the user is prompted to enter a password. Once, the admin user enters his/her username/password, the device verifies whether the provided credentials are valid. After a successful authentication, the web application reveals hidden URLs that are used for administration functionality. The problem stems from the fact that the authentication mechanism of some routers is so weak, that when an unauthenticated user requests a hidden URL, the web application does not ask for a password and delivers the hidden web page to the unauthenticated user. One may argue that this attack cannot be performed, because an unauthenticated user cannot know the hidden URL paths to access the related hidden web pages. However, this assumption is erroneous, because there are many ways that an attacker can discover hidden URLs. For example, the attacker can perform directory bruteforcing (i.e., try randomly various possible combinations of URLs) [14], since the URL paths are very easy to guess. To rectify this erroneous functionality, the web application should use session identities to check for each single HTTP request whether the user is in fact authenticated.

In the third category, the ADSL router performs an authentication check, only, when a request is performed using a certain HTTP method. In this case, the attacker can simply change the HTTP method from GET to POST or vice versa and gain access to the administrator’s functionality. Finally, in the fourth category, the authentication bypass is due to the following case. Many web applications use CGI scripts to process input data and generate dynamic content. In this case, authentication bypass can be performed because the web pages correctly accept HTTP requests only from authenticated users, but the respective CGI script, which performs the actual processing of the request, does not require an authenticated request. This happens because the developer overlooked the

fact that CGI scripts should also perform validation of HTTP requests.

2) Default authentication credentials

Most users neglect to change the default authentication credentials (i.e., the username and password) of their ADSL router. On the Internet, there are many sites such as the "Routerpasswords.com" [2], which freely provide a collection of the default credentials of several well-known ADSL routers. Moreover, vendors, usually, use the same or similar passwords across different models. Thus, it is easy for an attacker to guess the default credentials of an ADSL router, of which the administrator has not taken care to change them.

To prove this assertion, we examined the security of the default credentials used in BaudTec router series T263R* and TW263R* of the Greek ISP "OTE Conn-X". In both routers, we discovered that the default username of the administrator account is "admin". Even worse, we found no option to change it from the web management interface. On the other hand, the default password of administrator's account on T263R1U* router is "1234". In the case of T263R4* router, the default password is 12 digits and it is the MAC address of the ADSL router. For example, if the MAC address of the router is "00:13:33:1D:DD:62", then the administrator's default password is "0013331ddd62". The MAC address was found very easily, since it was written on a sticker underneath the ADSL router.

3) OS command injection

ADSL routers are mostly embedded Linux devices providing several basic Unix commands. This fact can be exploited from adversaries to perform OS command injection attacks. As its name implies, the main purpose of an OS command injection attack is to inject and execute arbitrary system commands, specified by the attacker, through a vulnerable web page of the ADSL router. In particular, since most ADSL routers are Linux devices, the web application of the router is essentially a system shell. Thus, an attacker may use the system shell to execute commands with the same privileges of the web application. The reason behind this attack is due to the lack of proper validation of the input data. Note that a web application, usually, takes input through forms (GET or POST) or other headers of HTTP requests (e.g., cookies, etc.). By replacing each possible input data with specific OS commands, the attacker can successfully perform this attack.

4) SQL injection

This well-known type of injection attack allows an attacker to inject code through the input data of a web application, in order to read or modify sensitive information from a database. This attack exists because the web application does not validate the provided SQL query from the user before its processing. In embedded devices such as routers, the related database usually does not have any useful information from an attackers' point of view. However, recently in [7] it was identified that an SQL injection attack targeted to a router can be used to pivot various critical attacks (i.e., buffer overflows) and gain remote access to the device with full privileges. Thus, SQL injections are considered an important attack vector in the context of ADSL routers, and developers should pay attention at validating the provided SQL queries.

B. Client-Side Attacks

1) Session Hijacking

In session hijacking, the attacker takes control of a user session and impersonates a legitimate user, after successfully stealing or guessing his/her authentication session ID or cookie. The vulnerability is attributed due to the fact that in several ADSL routers, session IDs or cookies are conveyed in plain-text. Thus, an attacker can simply perform packet sniffing (passive or active) and read sensitive data, including session IDs or cookies. To defend against session hijacking attacks, all sensitive data conveyed in the wireless interface should be encrypted using the SSL protocol (i.e., HTTPS).

2) XSS and CSRF attacks

The XSS attack is a common type of code injection attack on web-based applications. There are two primary flavors of XSS: non-persistent and persistent. The non-persistent (or reflected) XSS show up in case that the data provided by a user, most commonly in HTTP query parameters or in HTML form submissions, is used immediately by server-side scripts to parse and display a page of results. On the other hand, the persistent (or stored) XSS is more devastating, because it occurs when the code injected by the attacker is saved by the server, and then permanently displayed on the vulnerable web page. The main idea behind XSS attacks is the attacker first to inject malicious HTML or Javascript code in a web page of the router's web management interface. Next, when the unsuspecting user accesses this web page, the malicious code will be executed from the user's browser, aiming at session hijacking or compromising sensitive data from the user's computer. XSS attacks rely on the lack of properly sanitization of the input surface. Moreover, due to the fact that the Javascript security model is weak, attackers can discover various XSS attack vectors to achieve this attack. The vulnerability stems from the fact that certain web pages are trusted more than others in the context of the web browser allowing code to run with higher privileges.

Finally, CSRF is another common type of code injection attacks in web applications. An attacker forces an already authenticated user (e.g., with administration privileges) to execute unwanted actions through HTTP requests, which are not validated from the web application. These actions include changing administrator password or DNS settings, restoring router to default settings, rebooting the router, etc.

III. CASE STUDY: ZTE ZXV10 H108L ADSL SECURITY TESTING

The aim of this section is to investigate the security of a popular ADSL router named "ZTE ZXV10 H108L ADSL 2+ Wireless Router", provided by the Telecommunication Company "WIND Hellas". The router under investigation is an embedded device with MIPS CPU architecture. It includes a custom-made web interface for the device management, written in HTML and Javascript. During our security assessment, the ADSL router under testing had the latest

available firmware (I.e., V1.0.01_WIND_A01). After the security testing, we discovered two 0-day vulnerabilities in the web interface of the router. In particular, we discovered that the ADSL router is vulnerable to OS command injection (see figure 1) and persistent XSS attacks. It is important to mention that these vulnerabilities were discovered by manual testing. On the other hand, all automated security checks, using well-known security tools failed to discover any vulnerability.

First, we discovered a stored XSS vulnerability in a specific page of the web interface. For the successful exploitation of the XSS vulnerability, we added a specially crafted Javascript code in a field named "Host Name". Since this XSS is stored, every time the user visits this specific vulnerable page, the malicious Javascript code will be executed.

We used the following procedure for the discovery of the OS command injection vulnerability. First, we discovered that the ADSL router had the port 8083 open, which is used for remote access to the ADSL router through the Internet. This probably happens because the technical department of the specific ISP wants to have remote Internet access to the ADSL routers for troubleshooting purposes. Next, we tried to establish a remote telnet connection using the default credentials for the admin account, but the telnet server of the ADSL router rejected the provided credentials. This means that the technical department of the ISP uses different credentials to establish a telnet connection with the ADSL routers.

Continuing our testing, we found a web page that performs diagnostic functionality to discover broken connections. We explored this page and discovered that this page, essentially, was executing the *Ping* command. After several trials and specially crafted input combinations to perform OS command injection, we succeeded to perform arbitrary command execution (see Fig. 1). At this point, we were able to execute all the command line tools provided with busybox.

By exploiting this vulnerability to perform command injection attacks, first we executed the command "*uname*" and we discovered that the specific ADSL router runs Linux OS, based on BusyBox v1.01 [10]. Next, we executed the command "*vsftpd start*" to activate the FTP service. After that, we successfully established an FTP connection using anonymous login credentials. After that, we downloaded for analysis several files of the busybox filesystem. In a file named "*db_default_cfg.xml*" we found in plaintext the secret password of the root account. Then, we logged in using the discovered credentials of the root account and we found that the ADSL router unlocked some hidden features, such as activating telnet services from LAN. We were also able to view hidden web pages that provided sensitive information about the Intranet network of the specific ISP, such as internal IP addresses.

The last testing that we made was to examine whether the same root password is used among other devices of the same model. For this reason, we have written and released a python script that automates the exploitation of the specific router [12]. We tested several other routers and in all cases we verified the assumption that all routers of the specific model share the same credential for the root account. This discovery is crucial, because it allows the attacker to perform generalized attacks as we analyze in section IV.

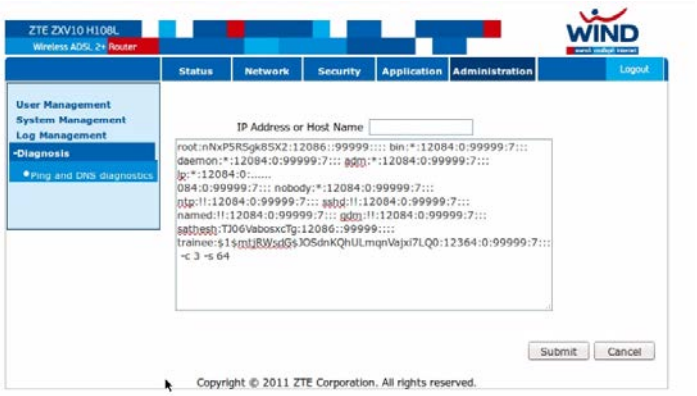

Fig. 1. Execution of the "*cat /etc/shadow*" command through the diagnostic web page of the router

IV. Analysis

A. Impact

Here we analyze the attacks that can be carried out from the discovered vulnerabilities. A malicious can exploit [12] the aforementioned vulnerabilities to perform a large-scale attack. In particular, he/she can mount an automated attack by scanning and discovering IP addresses of the specific ISP. Next, he/she can gain unauthorized remote access to the specific routers simply by using the root account credentials, which is the same for the routers of the same model. At this point, the attacker has three different choices to complete his/her attack:

(i) Perform a DNS hijacking attack. In particular, the attacker can replace the DNS settings of the ADSL router to point to a rogue DNS server, which is under the attacker's control. In this way, the attacker can achieve, for example, to direct the user of the ADSL router to a fake bank website instead of the legitimate bank website and steal his/her bank credentials. In other words, it can perform an effective phishing attack, which is undetectable.

(ii) Take advantage of the FTP connection to upload a sniffer that monitors the user's LAN traffic. A more devastating attack can be performed if the attacker uploads a malicious application that performs a DoS attack to a targeted server. In the last case, the attacker can combine several compromised and under his/her control ADSL routers to perform an orchestrated large-scale distributed DDoS attack.

(iii) Exploit the stored XSS to force the user to run a malicious java applet, which allows the attacker to have access to the user's personal computer, and through pivoting to exploit and gain access to the other devices or computers located in the local network of the compromised ADSL router.

B. Security Measures

Both developers and users of ADSL routers should follow some well-known security measures, in order to enhance security or eliminate the possibility of attacks. In particular, developers of the web management interface should take into account the following measures during implementation:

(i) Use strong authorization polices. The HTTP communication between user and the ADSL router must be secured using the SSL protocol to provide message confidentiality.

(ii) Validate and filter data coming from insecure sources, like user input, in order to protect from code injection attacks. Moreover, developers must use encrypted session negotiation to avoid session hijacking.

(iii) Sensitive information in the ADSL router must be stored in an encrypted database and not in plaintext files.

(iv) Perform systematic and continuous security audits and vulnerability checks on devices that are in the process of production (pre-market) both at the application layer (firmware) and at the device layer (hardware).

On the other hand, the users of the ADSL router should take into account the following guidelines:

(i) After the first log-in on the administration panel, the default administrator's password must be replaced with a new, strong and secure, at least 12 characters in length, consisting of upper / lowercase alphanumeric and special characters.

(ii) They should use the latest version of the activated browser and avoid visiting links that receive through suspicious e-mails or from social media.

(iii) After logging out from the router, users must clear browser's cookies.

(iv) They should always upgrade ADSL router's firmware regularly, only from the official manufacturer website.

V. Conclusions

This paper evaluated the security of ADSL routers by investigating vulnerabilities and analyzing possible attacks. As a case study we investigated the security of a popular ADSL router named "ZTE ZXV10 H108L ADSL 2+ Wireless Router", provided by the Telecommunication Company "WIND Hellas". After the security testing, we discovered two 0-day vulnerabilities in the web interface of the router. In particular, we discovered that it is vulnerable to Operating System (OS) command injection and stored Cross-Site Scripting (XSS) attacks. A malicious may exploit these vulnerabilities to perform a large scale attack. Specifically, he/she can perform DNS hijacking and redirect the end users to fake web sites for phishing attacks; mount a Distributed Denial of Service (DDoS) attack; or even spread a malware.

We have disclosed the discovered vulnerabilities to the affected vendor of the router and the ISP that provides it. We promptly received a confirmation from the customers' service department of the ISP that our request is being processed. However, at the time of writing this paper, we were not aware whether further actions have been taken from the ISP to patch the discovered vulnerabilities. We believe that many Greek ISPs provide ADSL routers that are vulnerable to the same or similar attacks. To prove this, we have investigated another ADSL router, that is a "Baudtec" router, provided by the Greek ISP "OTE Conn-X" and we have found similar vulnerabilities, such XSS, CSRF, information disclosure, etc. For this reason, we believe that more research is required to discover and patch critical 0-day vulnerabilities in ADSL routers.

References

[1] State of Broadband in Greece Second Quorter 2012, Hellenic Telecommunications & Post Commisions

[2] Brazilian hackers use DNS poisoning to infect users with banking Trojan, http://www.infoworld.com/d/security/brazilian-hackers-use-dns-poisoning-infect-users-banking-trojan-178421.

[3] Exploits Database, Offensive Security, http://www.exploit-db.com/

[4] Packet Storm, http://packetstormsecurity.com/

[5] RouterPwn framework, http://routerpwn.com/

[6] Router Exploitation, Felix "FX" Lindner, http://www.recurity-labs.com/content/pub/FX_Router_Exploitation.pdf, 2010

[7] Zachary Cutlip, "SQL Injection to MIPS Overflows: Rooting SOHO Routers", Blackhat 2012.

[8] OWASP, Cross-site Scripting (XSS), https://www.owasp.org/index.php/Cross-site_Scripting_XSS

[9] OS Command Injection, https://www.owasp.org/index.php/OS_Command_Injection

[10] BusyBox, http://www.busybox.net/

[11] Oriyano Sean-Philip, Robert Shimonski, Client-Side Attacks and Defense

[12] ZTExploit, https://github.com/stasinopoulos/ZTExploit

[13] Hristo Bojinov, Elie Bursztein, Eric Lovett, Dan Boneh, "Embedded Management Interfaces: Emerging Massive Insecurity ", BlackHat 2009, USA.

[14] DirBuster: a multi threaded java application designed to brute force directories, OWASP