

Security Audit of Safeplug “Tor in a Box”

Abstract

We present the first public third-party security audit of Pogoplug’s Safeplug device, which markets “complete security and anonymity online” by using Tor technology to protect users’ IP addresses. We examine the hardware, software, and network behavior of the Safeplug device, as well as the user experience in comparison to other forms of web browsing. Although the Safeplug appears to use Tor as advertised, users may still be identified in ways they may not expect. Furthermore, an engineering vulnerability in how the Safeplug accepts settings changes would allow an adversary internal or external to a user’s home network to silently disable Tor or modify other Safeplug settings, which completely invalidates the security claims of the device. Beyond this problem, the user experience challenges of this type of device make it inferior to the existing gold standard for anonymous browsing: the Tor Browser Bundle.

1 Introduction

Privacy on the Internet is becoming increasingly important as users realize how vulnerable they are to surveillance and theft of their data. A recent Pew study listed compromised emails/accounts, harassment, and stolen Social Security Numbers, bank information, and credit card numbers as some of the results of online visibility; according to the study, 86% of internet users have tried to become more anonymous online [19]. Despite this, users do not believe that they have the tools to solve this problem.

In December of 2013, the cloud storage company, Pogoplug, released the Safeplug, which is a small box that plugs into a user’s home router. It claims to: conceal your identity, hide where you live, shield your surfing habits, and make you anonymous online by routing all traffic through Tor [18]. We conducted the first public third-party security audit of the Safeplug by analyzing

the hardware, software, network behavior, and usability of the device. The following are some of our findings:

- The Safeplug functions as a HTTP proxy for the browser, which then uses Tor for outgoing traffic.
- Despite the use of Privoxy as an ad-blocker, the Safeplug does nothing to prevent users’ browsers from collecting both first- and third-party tracking cookies, allowing users to be de-anonymized across websites despite the presence of Tor [20].
- Safeplug users are vulnerable to a Cross-Site Request Forgery (CSRF) attack that allows an attacker external to their home network to modify the Safeplug settings (including silently turning off the use of Tor).
- A malicious user within the network can modify the Safeplug settings without notifying any other devices on the network.
- The Safeplug has a higher web request latency than that of the Tor Browser Bundle.

Pogoplug made use of good security principles by using auditable open source software on their device, and has the laudable goal of making online security the standard for more users. However, there is little reason to use the Safeplug over the Tor Browser Bundle; in addition to the Tor network technology used in the Safeplug, the Tor Browser Bundle contains protections against tracking cookies and fingerprinting, making it an improvement over the privacy offerings of the Safeplug.

2 Background

Safeplug. Safeplug [18], is a product that launched in December of 2013 from the cloud storage company Pogoplug, that offers any user the option of using Tor [3] without having to know about it or how it works. It allows users to browse the web from their own standard

web browser with complete security and anonymity for the cheap price of \$49. Safeplug offers Tor out of the box, with no additional software installation, by sitting between a user’s router and the Internet [21]. Pogoplug’s marketing pitch centers around the protection of users’ IP addresses by using Tor [9].

Pogoplug. Pogoplug, a subsidiary of Cloud Engines, also offers secure file server or file storage technologies for home networks, called Pogoplug. Based on our examination of the box discussed below, we suspect that a recent version of the Pogoplug hardware was relabeled as Safeplug and simply uses different software to achieve its newly branded purpose.

3 Design and Operation of Safeplug

3.1 Hardware

Before investigating the software, we took apart one of the two Safeplug devices we purchased in order to see the physical components. Figure 1(a) shows the top of the board and Figure 1(b) shows the bottom. The board incorporates: (A) an SD card slot, (B) a power connector, (C) a USB slot, (D) an ethernet connector, (E) ethernet transceiver, (F) lan transformer, (G) the Marvell 88F6192 integrated controller, and (H), (I) flash memory. The Marvell controller is from 2008 and is an ARM SoC [12].

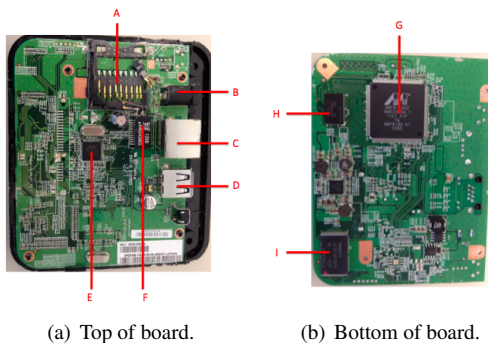


Figure 1: Safeplug circuit board after the teardown. The explanation of the components is in the text.

3.2 Software on the Safeplug

Table 1 shows the software used by the device, the corresponding version of the software, as well as the date of that versions release and if it is up to date. There are many known vulnerabilities in the pieces of software that are not up to date.

When a user modifies the settings page, Javascript on the settings page generates a POST request to `xspctrl`,

Table 1: Software and the corresponding version numbers on the Safeplug.

Safeplug Software	Version	Date	Up To Date (as of May 2014)
Linux Kernel	2.6.31.8	2009	No
Lighttpd Web Server	1.4.33	2013	No
Privoxy Proxy	3.0.21	2013	Yes
Tor	0.2.3.25	2012	No
Dropbear sshd	v0.52	2008	No

which is a script running via CGI in Lighttpd. The CGI handler copies a number of environment variables, and then forks and runs `xspctrl` via `execve` (in the constructed environment). `xspctrl` is a shell script with a method for each settings change; these methods execute any necessary Safeplug binary files (`go_update`, `go_upgrade`, `go_sshd`, or `go_updateexceptions`) and then return a HTTP response.

3.3 Networking

The Lighttpd, Privoxy, and Dropbear software give some indication of the networking services on the device. We used `nmap` to scan the ports. Before the activation of the Safeplug, only ports 80 (HTTP) and 3333 (unknown purpose) are open. After the activation, port 22 for SSH, port 80 for settings access and port 8080 for http-proxy are open. When Tor is turned on as a relay, port 9001 is opened as well. This corresponds with the note on the settings page about Tor relays discussed in Section 4.3 below.

3.4 Configuration on the Safeplug

The Safeplug configuration files can be found in `/opt/xce/etc` and include `sp.conf` and `sp_version` and `sp_torexceptions`. The first contains all of the configuration details: whether to use Tor, whether to block ads, and whether to act as a Tor relay or exit relay. The version file is used during the call to check for updates in the `xspctrl` script, and the exceptions file is used by the Privoxy configuration to control the whitelist of sites not to connect to via Tor. These configuration files are read by the scripts in `/opt/xce/etc/init.d` which enable Lighttpd, Privoxy, and Tor. As expected, Privoxy looks at the `tor`, `ad-block` and `exceptions` configurations, and Tor reads the `sp.conf` file to determine the correct Tor configuration file (regular, relay, or exit relay).

4 Usability

Several aspects of the user experience of the Safeplug also affect the security of the device. The Safeplug advertisements highlighted the ease of use, but omit important details.

4.1 Information Prior to Using the Device: Terms of Service

The TOS are never presented to the user: they aren't presented in the Safeplug package or shown during the activation process, and are only available through a small link at the bottom of the Safeplug website[18].

The Safeplug uses several pieces of open source software. A standard term of many open source licenses states that a company that uses the open source software must list the software used and its license, as well as the open source code of their own software that uses the license. The TOS contains a link to a page that would supposedly comply with this requirement: <http://pogoplug.com/home-en-developers-open-source.html> but the link is dead; instead, the reader sees a 404 error [18]. This dead link was noted by some members of the Tor-talk mailing list in November 2013 and has not been fixed as of May 2014 [2]. Pogoplug does maintain an open source page at <https://pogoplug.com/opensource>, which describes several pieces of software for the Safeplug: the bootloader, Linux kernel, XCE kernel support driver, glibc, BusyBox, and a Safeplug source bundle. However, there is no way to find this page from the Terms of Service or the Safeplug website.

4.2 Activation and Setup

The first step was to plug Safeplug into our router and activate our device. We followed the simple instructions, activated our device, and then ended on the configurations page. The configurations page shows a variety of platforms and browsers, shown in Table 2, and links to proxy configuration instructions for each one. The platforms presented include many popular desktop and mobile platforms. For every browser on every client device that the user wants to protect, they must set up the Safeplug as the browser's HTTP proxy and ensure that the proxy stays set up when the browser is updated or re-installed.

After we finished configuring our Safeplug and browser, we were taken to the Safeplug settings page. This page is shown in Figure 2. This page allows the user to set a number of options on the device, including turning Tor on/off, turning ad-blocking on/off, and turning relay node ability on/off (and if on, an additional option

Table 2: The platforms and browsers that the Safeplug settings page provides instructions for.

Platform	Browsers
Windows	Internet Explorer, Chrome, Firefox
OSX	Safari, Chrome, Firefox
iOS	Safari
Android	Chrome

appeared to allow the device to be an exit relay). Additionally, the user can specify “white-listed” domains that will be connected to directly even if Tor is turned on, without going through the Tor network. The Safeplug FAQ website describes this as an option for sites like banks that prevent access from foreign IP addresses[18].

4.3 Explanation of Tor Configuration Options

One of Safeplug's configurations is the use of the device as a Tor relay node in the Tor network. When the device is initially set up, the default setting is to not use it as a relay node.

The settings page describes Tor in an extremely basic way:

“Safeplug uses the Tor network to secure your Internet connection. Tor works by routing your Internet through a series of random destinations, much like driving a twisty, complicated route to throw off someone who is following you, making it impossible for websites and organizations to identify the source or destination of Internet traffic.”

On the other hand, it assumes a much higher level of knowledge about networks for users trying to configure a relay:

You must forward inbound traffic on port 9001 to [IP-of-Safeplug]. For help configuring your router, click here.”

The link provided is to an advertisement-filled and confusing website.

Using Safeplug as an exit node has possible legal repercussions. As an exit relay, all traffic that exits the node can be traced back to the Safeplug's IP address; it is likely that some of this traffic contains illegal information or is part of illicit activities. The Tor Project recommends not running an exit relay from a user's home: “If law enforcement becomes interested in traffic from your exit relay, it's possible that officers will seize your

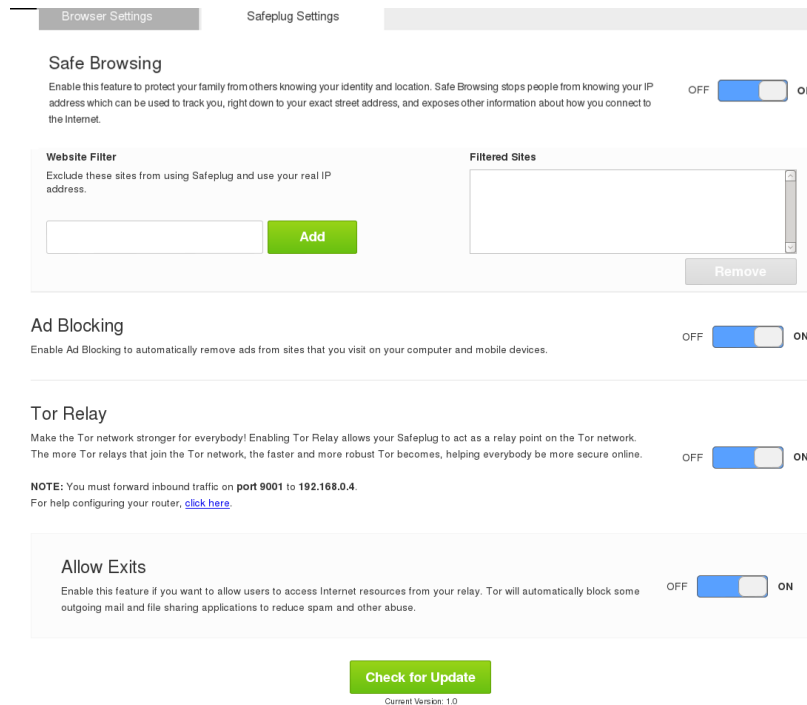


Figure 2: Safeplug settings page. The last button “Allow Exit” only appears if the relay option above has been turned on.

computer. For that reason, it’s best not to run your exit relay in your home or using your home Internet connection [1].” Considering that the Safeplug is intended for home use, it is a poor design choice to allow the user to use it as an exit relay without providing the user with any contextual information.

4.4 Internet Use

Next, we examined popular webpages, with an eye for changes in the user experience. A user would probably decide not to use Safeplug if they could not read their web pages (due to a different language), or if they could not log into their accounts (some web sites will lock a user out if they try to log in from multiple countries in a short time period). While browsing the Internet, we experienced some pages in German and Swedish; if a user is not familiar with Tor, they may not understand why this is happening. We were also prompted with a request to “Enter the name of the city where you usually sign in” when we tried to log into our Google account, indicating that Google noticed we were trying to login from an “unusual” location.

4.5 Cookies

While browsing the Internet in a fresh browser session, we logged into a Google Account in a tab, and then subsequently went to a website that had the Google+ widget in a different tab. When we clicked on the Google+ widget, it knew who we were; so we can confirm that a cookie associated with the login was sent to the second Google+ page. We experienced the same situation with Facebook and its corresponding “like” button. This was confirmed by using FourthParty, a plugin developed by Jonathan Mayer to collect information about cookies and other browsing data [13].

This is clearly not the perfect “anonymity” that Safeplug promised in their publicity. Although normal use does follow this model of persisting logged in status across browser tabs and sessions, a user concerned about anonymity and seeing each session coming up differently (e.g., German then Swedish) might mistakenly believe that they do not have to log out of their social media accounts to preserve their anonymity when accessing other websites.

More interesting and damaging to the user’s control over their anonymity would be third-party cookies because the user cannot remove those just by logging out. Most browsers require a trip to the browser settings to clear cookies (or not have them set in the first place).

When collecting data on the existence of third-party cookies, we analyzed two separate browsing sessions; they were both new sessions with no cookies. One of the sessions used the ad-block feature of the Safeplug and the other did not. After analyzing data from Fourth-party, we found many third-party cookies in both sessions. These included cookies from: abmr.net, bizographics.com, krx.net, and bluekai.com among many others. The ad-block functionality on the Safeplug reduced, but did not eliminate, these third-party cookies.

Although Safeplug has a warning about clearing cookies on their FAQ page, it only mentions clearing cookies after a browser session. Of greater concern would be tracking a user during their browser session from website to website; preventing this requires knowledge and constant vigilance from the user, or a browser that does not accept third-party cookies, such as the one provided in the Tor Browser Bundle.

4.6 Browser Fingerprinting

We used Panopticlick [8] to examine the fingerprint of a freshly installed Firefox browser running through the Safeplug proxy. Panopticlick found the browser to be unique, which means that websites that did fingerprinting could very accurately track, correlate, and de-anonymize user traffic without knowing the IP address or even storing a cookie. The presence of the Safeplug, as an HTTP proxy, should be completely undetectable by the fingerprinting service because HTTP is designed to make proxies transparent. Unlike the Tor Browser Bundle, Safeplug users do not even have the fingerprints of other users of the service to help hide their fingerprints. Allowing the user to use their own browsers significantly increases the amount of variation and customization between users and therefore the likelihood of having a unique fingerprint. This danger to anonymity (unlike cookies) is not even mentioned in the Safeplug FAQ page [18].

4.7 Latency

If the latency of web requests using the Safeplug is noticeably longer than that of normal Internet use, users may be deterred from using the device. Similarly, if turning Tor on, but not ad-blocking, adds a significant time delay, the user may only turn on the ad-blocking feature (without Tor). We recorded the time for a web request on the following settings:

- Plain Firefox (no use of the Safeplug device)
- Firefox, no Tor, no ad-blocking (traffic running through the Safeplug device)
- Firefox, Tor, no ad-blocking

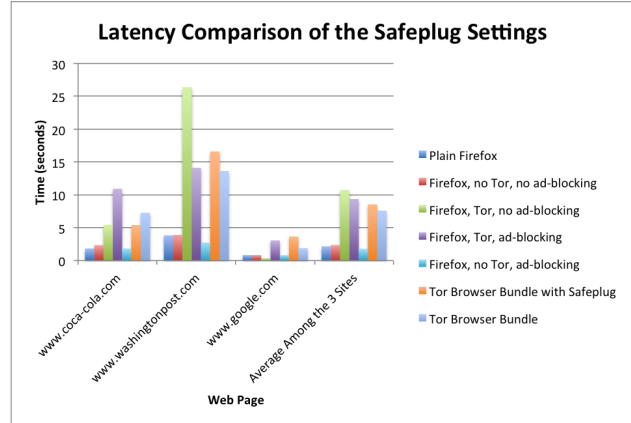


Figure 3: Latency of web requests.

- Firefox, Tor, ad-blocking
- Firefox, no Tor, ad-blocking
- Tor Browser Bundle with Safeplug (all settings off)
- Tor Browser Bundle

For each of the settings, we took 20 measurements; Figure 3 shows the average time of a web request on each of the specified settings for three different web pages. When taking these measurements, we loaded the page, but did not scroll; in many cases more objects are loaded when scrolling down a page. The differences between web pages can most likely be attributed to the amount of advertisements and content running in plugins (such as video) on the web page.

5 Vulnerabilities

As we discovered during our software analysis, the Safeplug has a remote procedure call (RPC) capability. This is a script called `xspctrl` found in `/opt/xce/html/svc`. Functional calls to this script include the ability to enable and disable all of the Safeplug settings, including Tor, ad block, and Tor relay. None of them require any arguments in the POST string.

5.1 Insider Attack

The Safeplug has no validation or authentication on the settings page for these RPC calls, so any malicious party inside the home network can easily modify the settings. Unlike many home routers, there is no username/password combination necessary to access the settings page. A more technically advanced user could send the call directly to the RPC server. Figure 4 shows an example of the RPC version of this attack. If the adversary can get into the network they can perform these attacks. For

```
File Edit View Search Terminal Help
forensics:~/safeplug-audit/rootfs/opt/xce/sbin$ curl --data "" 192.168.
0.4/svc/xspctrl/disableTor
{"version":"1.0","tor":0,"torrelay":0,"torexitrelay":0,"adblock":1,
"pacurl":"http://192.168.0.4:80/wpad.dat"}
forensics:~/safeplug-audit/rootfs/opt/xce/sbin$
```

Figure 4: Disabling Tor as an RPC attack.

example, if the user has an open WiFi network, then anyone nearby can launch this attack - potentially performing a sort of drive-by deanonymization.

Since the RPC version of the attack just involves basic Internet tools (the ability to send a POST request), the attacker could also be any kind of device on the local network, or the local gateway if it is compromised.

5.2 CSRF Attack

Any external website can also perform the above attack by returning a correctly formatted POST string via an internal user's browser. This executes the same functionality as the Insider Attack, but the attacker does not need to be on the local network. This Cross Site Request Forgery (CSRF) attack requires a nonmalicious insider user to visit a web page controlled by the attacker, allowing the attacker to send the POST to the Safeplug. If the attacker does not know the IP address of the Safeplug, he can perform an exhaustive search on the address space. We implemented this attack with less than 20 lines of Javascript code. The following steps are necessary for the attack:

1. Set up a web page with the Javascript code, which will send the POST request of the following format to all addresses in the common ranges: `http://<IPaddress>/svc/xspctrl/disableTor`.
2. Send the malicious link to a user in the targeted private network.
3. Once the user clicks the link and loads the malicious site, the correctly formatted POST request will be sent to every IP address in the ranges.
4. Tor is disabled silently. The user must check or refresh her settings page to learn that Tor is off.

While this attack requires a greater amount of time because the local IP address of the Safeplug must be guessed via search, the number of private address spaces is small, and the space likely to be occupied by a Safeplug on a home network is even smaller.

The largest observed time to send requests to the 192.168.0.0/24 space was approximately 400 milliseconds; the entire attack costs approximately 800 milliseconds for sending requests to both 192.168.0.0/24 and

192.168.1.0/24 ranges - even when the website was being loaded over Tor. In the case of a private network in the range of 172.16.0.0/16, the attack took less than 12 minutes (this generates script timeout warnings in most major browsers, which affects the timing of this attack). This means that it would take a few hours to send requests to the full 172.16.0.0/12 range, which is commonly used in business networks. The final private network space is 10.0.0.0/8 which is too large for an exhaustive search, but some simple optimization might make it feasible as well. For example, using a GET request to get and parse the Safeplug settings page would allow the script to positively identify the Safeplug and stop the search. However, the 192.168.0.0/24 and 192.168.1.0/24 ranges are much more common in home networks; because Safeplug is geared towards home network use, in most cases the script will take less than a second.

In addition to disabling Tor, the attacker can modify any other settings on the device. This includes: enabling/disabling Tor, enabling/disabling ad-blocking, enabling/disabling the use of the device as a Tor relay node [note: enabling requires the user to do additional setup], enabling/disabling the use of the device as an exit node (if it is already a relay). Lastly, the attacker can also modify the user's whitelist of sites that should not be routed through Tor. This whitelist attack is particularly dangerous because the change is silent and much harder for the user to notice the addition of a single website to the whitelist than a global loss of Tor. (Removal of a webpage from the whitelist is likely to produce visible usability problems.)

5.3 Gaining Access through SSH

Another command available to the RPC server is enabling SSH to the device. SSH instructions for Pogoplug's other device (called Pogoplug) are widely available online and an email in the Tor-talk mailing list confirmed the instructions [5]:

```
curl --data '' http://<IP>/svc/xspctrl/enableSSHD
ssh root@<IP-of-Safeplug>
password: ceadmin
```

Having a publicly available root password means that SSH is done effectively without authentication. Once

the box was activated and had Lighttpd installed, the SSH procedure was available and any adversary on the home network could log into the box and install malware, surveillance software, or virtually anything they desired.

5.4 Spoofing the Update Server

An additional dangerous class of vulnerabilities comes from the use of insecure TCP rather than any kind of encrypted communication during the software update process. We discovered that the update script is downloaded via TCP from an IP address provided by the Pogoplug servers. This update script (run as root) then downloads the Safeplug's software (Tor, Privoxy, Lighttpd, and wget) and checks it against MD5 hashes, but it is not clear whether there is verification of the update script itself. An adversary could use DNS spoofing or compromise the Pogoplug server and force users to download a malicious update script - for example, something that turns the Safeplug into a surveillance box while appearing to provide the correct functionality. Because the activation occurs over TCP rather than anything more secure, an adversary who can spoof DNS replies to the user can install arbitrary software onto the Safeplug box. This turns a device that does not live up to expectations, but is otherwise harmless, into something that actively harms security on the network.

6 Related Work

To our knowledge, there has been no other study analyzing the security of Pogoplug's Safeplug device. However there has been much prior research on the primary technology that the device uses, Tor, as well as the vulnerability of fingerprinting.

Tor. Prior security evaluations of the Tor network reveal a myriad of potential vulnerabilities. A significant area of research on Tor relates to diversity of autonomous systems (ASes). Researchers argue that a user's anonymity may be compromised by using geographically diverse ASes [7, 15]. There has also been proven traffic correlation attacks that are efficient on the Tor network [14, 16]. Johnson, Wacek, Jansen, Sherr, and Syverson found that in a period of six months, 80% of all users may be deanonymized by a reasonably realistic Tor-relay adversary [11]. While Safeplug does not introduce or modify how Tor is used, it routes all traffic through the Tor network; Safeplug is also vulnerable to the attacks found in prior research on Tor.

Fingerprinting. Website fingerprinting attacks as well as remote physical device fingerprinting attacks have shown they can identify users, even when specific defenses have been used in order to prevent them. Previous research has shown that web page fingerprinting

attacks are possible [6, 10, 17]. Cai, Zhang, Joshi, and Johnson found that their fingerprinting attack is successful 83.7% of the time when the defense is the use of Tor [4]. These results can be extended to the security of Safeplug. Because Safeplug uses Tor to anonymize users, it may be susceptible to this type of fingerprinting attack.

7 Discussion

7.1 Necessary Fixes

The most critical engineering fix is authentication in the POST calls to prevent the CSRF attack. A typical approach to preventing CSRF attacks is using a cookie and a hidden form field set in the settings page of the Safeplug hosts; the cookie must be returned by the browser when making the POST request to the RPC server [22]. Although someone doing a CSRF attack such as the one described above could get the cookie sent, because of the same-origin policy, the adversary would not be able to examine the contents of the cookie to determine what to put in the form field.

7.2 Structural Problems

However, there are much more significant structural problems with implementing a Tor connection via an HTTP proxy. Several of the usability problems involve user awareness and vigilance. For example, cookies and fingerprinting problems mean that users could still be tracked across websites, regardless of whether the ad-block functionality on the Safeplug is enabled.

One type of client that deserves special attention is a mobile phone user. Safeplug provides proxy functionality and instructions for Safari on the iPhone and Chrome on Android. However, while this does supposedly give the option for more mobile users to make use of Tor, proxy configuration on mobile devices comes with significant usability issues. For example, proxying would only work while the user is on the same wifi network as the Safeplug. If any data is sent over the cellular network or another wifi network, then the security of Tor is lost. Additionally, the user may have to disable the proxy whenever they move to a different network, and remember to re-enable it when they want to use Tor. This is certainly not transparent usability. Users who are truly concerned about anonymity online should eschew the Safeplug and purchase a device that supports the Tor Browser Bundle or other Tor Project software.

7.3 Opportunities

Despite all the structural problems, is there a market for a Torifying piece of hardware? Given the security pitfalls in comparison to a piece of software such as the Tor Browser Bundle, there seems to be no reason for a user who can run the Tor Browser Bundle to purchase the Safeplug or any other device. For mobile phones, the proxy problems with mobility contribute to an already high usability cost. However, there are an emerging class of “smart home” devices which may connect to the Internet. It is possible that some of these devices can be configured to use an HTTP proxy or some other middle box to Torify their traffic to an external service provider. For them, is the benefit of some anonymity via a Safeplug-like device worthwhile? Since the data sent by these devices to the service provider likely contains identifying information, the use of Tor would only protect the user’s location at the expense of connection time and load on the Tor network. Since proxy configuration on such devices is likely to be difficult and the amount of information hidden is unlikely to be worth the effort, a Torifying box that functions as a proxy is of questionable value in this space as well.

8 Conclusion

Ultimately, the structural concerns of the Safeplug “Torifying proxy-in-a-box” strategy indicate that this is problematic as a method for security and anonymity online. It is critical for Safeplug to correct their security errors, particularly the vulnerability to silently disable Tor, in order to protect customers who have already made use of the device, but users who are truly concerned about safety and anonymity online should make use of technologies directly from the Tor Project.

References

- [1] The legal faq for tor relay operators. <https://www.torproject.org/eff/tor-legal-faq>.
- [2] Tor mailing list. <https://lists.torproject.org/pipermail/tor-talk/2013-November/031199.html>.
- [3] The tor project. <https://www.torproject.org/>.
- [4] CAI, X., ZHANG, X. C., JOSHI, B., AND JOHNSON, R. Touching from a distance: Website fingerprinting attacks and defenses. In *Proceedings of the 2012 ACM conference on Computer and Communications Security* (2012), ACM, pp. 605–616.
- [5] COLLETON, L. Fwd: Ssh on safeplug. <http://archives.seul.org/or/talk/Jan-2014/msg00003.html>.
- [6] DYER, K. P., COULL, S. E., RISTENPART, T., AND SHRIMP-TON, T. Peek-a-boo, i still see you: Why efficient traffic analysis countermeasures fail. In *Security and Privacy (SP), 2012 IEEE Symposium on* (2012), IEEE, pp. 332–346.
- [7] FEAMSTER, N., AND DINGLEDINE, R. Location diversity in anonymity networks. In *Proceedings of the 2004 ACM workshop on Privacy in the electronic society* (2004), ACM, pp. 66–76.
- [8] FOUNDATION, E. F. Panoptick. <https://panoptick.eff.org>.
- [9] HALFACREE, G. Pogoplug launches tor-powered safeplug. <http://www.bit-tech.net/news/hardware/2013/11/25/pogoplug-safeplug/1>.
- [10] HERRMANN, D., WENDOLSKY, R., AND FEDERRATH, H. Website fingerprinting: attacking popular privacy enhancing technologies with the multinomial naïve-bayes classifier. In *Proceedings of the 2009 ACM workshop on Cloud computing security* (2009), ACM, pp. 31–42.
- [11] JOHNSON, A., WACEK, C., JANSEN, R., SHERR, M., AND SYVERSON, P. Users get routed: Traffic correlation on tor by realistic adversaries. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security* (2013), ACM, pp. 337–348.
- [12] MARVELL. 88f6190 and 88f6192 integrated controller hardware specifications. http://www.marvell.com/embedded-processors/kirkwood/assets/HW_88F619x_OpenSource.pdf.
- [13] MAYER, J. Fourthparty. <http://fourthparty.info>.
- [14] MURDOCH, S. J., AND DANEZIS, G. Low-cost traffic analysis of tor. In *Security and Privacy, 2005 IEEE Symposium on* (2005), IEEE, pp. 183–195.
- [15] MURDOCH, S. J., AND ZIELIŃSKI, P. Sampled traffic analysis by internet-exchange-level adversaries. In *Privacy Enhancing Technologies* (2007), Springer, pp. 167–183.
- [16] OVERLIER, L., AND SYVERSON, P. Locating hidden servers. In *Security and Privacy, 2006 IEEE Symposium on* (2006), IEEE, pp. 15–pp.
- [17] PANCHENKO, A., NIESSEN, L., ZINNEN, A., AND ENGEL, T. Website fingerprinting in onion routing based anonymization networks. In *Proceedings of the 10th annual ACM workshop on Privacy in the electronic society* (2011), ACM, pp. 103–114.
- [18] POGOPLUG. Safeplug. <https://pogoplug.com/safeplug>.
- [19] RAINIE, L., KIESLER, S., KANG, R., MADDEN, M., DUGGAN, M., BROWN, S., AND DABBISH, L. Anonymity, privacy, and security online. *Pew Research Center* (2013).
- [20] REISMAN, D., ENGLEHARDT, S., EUBANK, C., ZIMMERMAN, P., AND NARAYANAN, A. Cookies that give you away: Evaluating the surveillance implications of web tracking (draft: April 2, 2014).
- [21] SOLON, O. Safeplug makes it super-easy to harness tor’s anonymity at home. <http://www.wired.co.uk/news/archive/2013-11/22/safeplug-tor>.
- [22] ZELLER, W., AND FELTEN, E. W. Cross-site request forgeries: Exploitation and prevention. *Bericht, Princeton University* (2008).