The Cookie Monster: From $0 to Hero

Cookie stuffing for profit

By Nik Telkedzhiev
nikola.telkedzhiev@tufts.edu

Mentor:
Ming Chow
https://github.com/mchow01

Abstract

A long time ago, online merchants started using the so-called “affiliate networks” to sell more of their merchandize. The way an affiliate network works is fairly simple. First, brands partner up with affiliates (people who have some influence over the target market of the brand because of their online presence). Then, the brands offer to pay the affiliates (usually between 8% and 12% of the purchase price) every time a person redirected by them purchases something from the brands’ online stores. Thus, both parties benefit. Brands get more sales and influencers monetize their social status. The way that merchants track whether a purchase is made through an affiliate is not complicated too. First they provide their affiliates with custom links to the online stores. Each such link contains the affiliate’s ID. Then, when a person opens the link and is redirected to the brand’s online store, a cookie containing that same affiliate ID is saved on the client’s browser. When a transaction is made, the brand affiliate system checks to see if a cookie with an affiliate ID is present, and if it is the affiliate is paid. However, there are ways to exploit the cookie setting/tracking process to financially profit.

1. Introduction

Cookies were invented with the clear purpose of storing information on the clients' browsers. Their goal is to introduce states into the stateless HTTP/HTTPS protocol. That means that things like your current log in state, ID, and other preferences can be stored in your browser with the goal of optimizing your user experience. For example, instead of logging in Facebook every time you open the site, if you choose, you can stay logged in. In that case, an authentication cookie is stored on your browser and next time you go on Facebook, Facebook's server reads your cookie and you are automatically logged in. The process of setting a cookie is extremely simple. A cookie is just a representation of a name-value pair with an expiration date. All that you have to do if you want to save a cookie into the client's side is one line of code, to set the name, value and the expiration date.

Historically, cookies have been blamed for degrading the security of the websites that use them. However, is this paper we are not going to focus on how we can use the information stored in the cookie to get unauthorized access. We are going to show ways that cookies can be

overwritten in a way to mislead merchants and allow exploiters to financially profit.

Considering the enormous size of the online retail industry and the money being paid to affiliates worldwide, we can imagine what sizable profits one can make by simply changing the cookie that stores the affiliate's ID. By overwriting it, a maliciously minded person, is basically telling the merchants that he is responsible for the sale. When a merchant is to examine the sale, a cookie with a affiliate ID will be present in the client's browser, so the owner of that affiliate ID, in this case the exploiter, will be given the commission the sale. The attacker is not responsible in any way for redirecting people to the merchant's site but since a cookie with his affiliate ID is found upon purchase, he is the one getting paid. Thus, he is either stealing another affiliate's money, or just getting unfairly paid from the retailer.

2. To the Community

This topic is of utmost importance to the online retailers world. If this vulnerability is exploited on a greater scale, a general distrust

towards this cost efficient way of marketing might be developed. That might make online retailers stray away from marketing through an affiliate network. This will financially damage both the merchants and the affiliates. Furthermore, even if is it exploited on a smaller scale, every time a purchase is made with corrupted cookie money are being transferred to the wrong party. Either affiliates are not getting paid or merchants are paying for something that didn't happen. There are big companies such as MyProtein, the leading UK online retailer of nutrition supplements, which are heavily dependent on this kind of marketing. Therefore, if we allow hackers to exploit this vulnerability, that might cause significant real world damages.

3. Exploitation

Since it is that easy to set a cookie, there are many ways that an attacker can exploit this vulnerability without the awareness of the victim. Various social engineering techniques can be applied to make potential clients visit a website which contains code that changes their cookies to the affiliate ID of the attacker. It is important to note that, an

attacker must also be member of the affiliate network of the merchant and have a real affiliate ID in order to get paid when clients make purchases with his ID. There are, however, more efficient ways of stuffing a greater number of cookies at once. Such an approach will be to just simply upload and execute the code that sets the visitors' affiliate ids to your ID on a website that gets many hits every day. Ideally, the people who go on this website will be same ones that shop at a certain online merchant exposed to this affiliate scheme. If this is successfully executed, then the cookies of all the people that visit that site will be corrupted. Every time any of those people make a purchase, money are being transferred to the bank account of the attacker.

For example, imagine that you decided to exploit this vulnerability on well known website such as bodybuilding.com. The perfect place to perform your attack will be the forum of the website. Since many people like to see reviews by others before placing an order, the chances that you will be corrupting a cookie of a potential buyer are extremely high. All you need to do is upload an image or an *iframe* and set the source to the referral link given to you as an affiliate. Then, when a person visits the pages, the link is automatically opened and the cookie is set to

contain your affiliate ID. You can literally load up a million affiliates IDs and should the page visitors buy something in the future, you are getting paid. Just to provide some perspective on potential earnings, the 2013 revenue of bodybuilding.com was $420.5 M. If an attacker successfully floods the forum with his affiliate id and let's say reaches 5% of all people shopping there, that would mean that he **has made $1,682,000** (420.5M x 5% x 8%) **only from one merchant**.

4. Defense

One way to protect against this type of cookie stuffing is to monitor referrers and conversion rates. For example, when an affiliate ID belongs to a certain site, but the referral came from another source such as a social network or some other high-traffic website, most probably that referral was not real. However, if the "affiliates" post their code on an SSL page, then no referrer header field is being transferred. That means that merchants can only see the affiliate ID (no referral source is included in the header). Without knowing the referral source, it is impossible to monitor for stuffed cookies.

5. Conclusion

Ironically, the SSL protocol that makes the Internet more secure allows people to unfairly profit by performing the above-described cookie stuffing technique. The only possible way to protect against it would be to be more selective when you are accepting new affiliates. Even then though, if the affiliate decides to exploit this vulnerability on a small enough scale so that his referrals would not looks suspicious, it is nearly impossible for merchants to catch the offenders. The only way to get caught will be to actually find the malicious code on the source code of a page that can be somehow connected to the owner of the affiliate ID. As no merchant can afford to scan the entire Internet, catching the bad guys after we have allowed them to enter our affiliate networks seems impossible. The only way to protect is to not allow when to join as affiliates in the first place.

6. References

Jeremiah Grossman. (2009, Feb 10) *Get rich or Die Trying.* Retrieved from https://www.youtube.com/watch?v=SIMF8bp5-qg

W3Schools. *JavaScript Cookies.* Retrieved from http://www.w3schools.com/js/js_cookies.asp

Bodybuilding. *World's #1 Online Fitness Website and Supplement Store.* Retrieved from http://www.bodybuilding.com/

IETF. *HTTP State Management Mechanism*. Retrived from http://tools.ietf.org/html/rfc6265 - section-3

Network Working Group. (1999) *Hypertext Transfer Protocol.* Retrieved from https://www.ietf.org/rfc/rfc2616.txt

Inc. Magazine. *Inc. 5000 2014.* Retrived from http://www.inc.com/profile/bodybuildingcom