Click-fraud Monetizing Malware: A Survey and Case Study

Tommy Blizard, Nikola Livic
Microsoft
Tommy.Blizard@Microsoft.com
Nikola.Livic@Microsoft.com

Abstract

Malware monetization via fraudulent ad traffic is lucrative and relatively easy. Not surprisingly, there is a higher incidence of malware that monetizes in this way today than in the past. Although this type of attack is not new, current methodologies to make malware-generated ad traffic appear organic, at such large scales, while remaining virtually undetected are novel.

The on-line world of advertising is complex and not generally well understood. It is well funded and not well supervised. There is little transparency and there are conflicts of interest; thus creating a fertile ground for criminal activity.

This paper provides a brief survey of this space, outlines an example analysis of current click-fraud malware, and presents new approaches to combat this type of malware monetization.

1. Introduction

At a very high level in the world of on-line advertising, there exist publishers, ad networks and advertisers. The advertisers pay for their ads to be displayed. The publishers possess websites and/or apps that attract visitors via content or functionality. The publishers get paid to display ads and their visitors, who are the advertiser's potential customers, may click on an ad link. Ad networks act as brokers between the publishers and advertisers. For example, when Johnny Surfer surfs to a publisher's site, www.SomePublisherSite.com and clicks on an ad link for WaxWidgets, the advertiser, WaxWidgets pays the ad network for that click, and the ad network pays some fraction of that to www.SomePublisherSite.com. This is Pay-Per-Click (PPC) advertising.

In reality things are much more complex, ad networks can also be publishers (as is the case for Google and Microsoft ad networks, whose search portals are publishers). There are syndication and sub-syndication schemes where ads are shared by an ad network with a publisher who in turn shares those ads with another publisher. Each publisher down the line receives some fractional share of the click profit. There are also affiliates of publishers who supply traffic to ads. These affiliates can purchase traffic from other affiliates that purchase traffic from other affiliates, and so on, such that the origination and quality of the traffic becomes obscure. Quality is defined in this context as real human traffic that has conversion potential. Conversion is defined by advertisers as the desired action taken by the potential customer after clicking an ad link. This desired action could be to buy a product, fill out a form, submit feedback, or any possible action behind the ad campaign. The complexity and profit in this structure has created some fruitful opportunities for crime.

Click fraud could be defined as the intentional misappropriation of ad revenue by generating a click that doesn't originate from a potential customer or the rightful publisher. Since advertisers pay for each ad click- in a PPC system, there is an economic incentive for the publisher and the ad network to generate as many clicks as possible. This is tempered, however, by the fact that publishers and ad networks must maintain a certain level of click quality to remain in business. Though short-term gain is realized for click-fraud, the longevity of ad networks and publishers depends upon maintaining an acceptable return on investment (ROI) for advertisers.

978-1-4673-4879-9/12/$31.00

As ROI is diminished by click-fraud, over time ad networks stand to earn less for placing ads. Also advertisers, though they may want high quality clicks for their own ads, may not be so concerned for their competitors. For example, advertiser A could repeatedly click on competitor advertiser B's ads, by flooding the publisher with fake clicks, B's ad budget is effectively depleted. Typically with search ads, advertisers bid on keywords relevant to their ad campaign so there is economic motivation to compromise a competitor's higher bid. In addition to minimizing ROI on a competitor's ad campaign this would also allow publisher A's ads to be displayed at a lower bid price now that B's ads have been "clicked".

There exist many variations of click fraud today, but they all generate improper charges for an ad click, by either charging for a click that has 0% chance of conversion or subverting the payment of the true publisher to a false one.

Ad networks and advertisers have filtering/detection methods designed to prevent click fraud. Because the algorithms are not disclosed there are data scarcity challenges to accurately gauge: the current level of click fraud, the defense methodologies and the prevention success rates. Some estimates have pegged annual click fraud on search engine results to be worth $100 million [1]. Others have estimated 22% of all ad clicks are fraudulent [2]. While other estimates are significantly less, only .02% of clicks are fraudulent which need to be refunded to advertisers according to [3]. The wide variation in these estimates could be ascribed to different underlying motivations of people who made these estimates. Ad networks have a vested interested in having a perception of minimal click fraud numbers, while click forensic companies have an opposite interest.

A recent systematic approach to measure large-scale click fraud has placed it at 26% or greater [4]. This work includes disclosure of the methodology used to derive its figures, so its objectivity can be scrutinized and its results can be reproducible. Provided the actual rate is somewhere in between the high and low estimates, with online advertising totaling $32B in 2011, with 48% of that coming from search advertising [5] , there is still significant monetary gain for malware doing search click fraud.

As more malware today engages in click fraud, effective automatic techniques to detect and remediate become more important. This is important to consumers who want a high-quality network experience and the 'free' websites/apps supported by advertising , it is critical to the overall health of the on-line commerce ecosystem and it is a key link for the criminal to profit.

2. Background

Click fraud encompasses a variety of maliciously intended link clicking. One basic motivation is for direct monetary gain, but other motivations include click-spamming where clicks are generated to deplete a competitor's ad budget; or to inflate a site's search engine optimization (SEO) ranking; or to frame a publisher for traffic fraud; or for pure mischief-without any perceivable benefit. The spectrum of methodologies for click fraud span a single user publishing ads and manually clicking from a single IP, to a global distributed botnet using state-of-the-art algorithms interacting with the user to generate organic clicks, to sophisticated sham sites doing cost per click (CPC) arbitrage, to click farms where humans are paid minimally to click on ad links. Each of these scenarios presents unique challenges in detection and policy. This paper only focuses on a small subset of the click-fraud space; namely, malware generating click-fraud for remunerative gain by interposing upon the user's browsing. It should be noted that this focus differs from previous work in this area [6] that analyze clickbots automatically generating clicks without user interaction. Though the techniques mentioned, by Miller et al, of obfuscating the origination of the traffic through layers of indirection in syndication channels is relevant to our example analysis below.

Pay-Per-Click (PPC) has had a somewhat checkered past. The online PPC ad space is about 15 years old, and some prominent malware have used click fraud to monetize. The following highlights some key points (from our point of view) in its history:

- **1996:** OpenText financed their search portal by selling search results to "preferred" customers.

This may have been the first attempt away from banners into a more granular ad click model.

- **February 1998**: GoTo.com, now part of Yahoo, originated the Pay-Per-Click (PPC) internet advertising model for search engine results.
- **2000:** Google created its own PPC ad network (Adwords).
- **2001:** Chase Law Group was one of the first documented cases of click fraud when a competing law firm was guilty of spuriously clicking on Chase's $3-$10 ad links [7].
- **2005:** Yahoo settled a $4.5 million class action lawsuit for not doing enough to stem click fraud on their ad network. (In 2006, Google settled similar a case for $90 million.)
- **2006:** Microsoft deploys its own PPC system.
- **May 2006:** Clickbot was discovered, the first low-noise click fraud against syndicated search engines with roughly 100,000 infected machines [8].
- **2006 to 2011:** DNSChanger (Alureon) a resilient bootkit and sophisticated malware executing platform, employed click fraud as one of its monetizing vehicles. It tampered with DNS so as to redirect traffic to ad sites. There were approximately 4 million machines infected at its peak, with estimates of $14 million illegally obtained [9].
- **2009:** A Koobface variant was found to be infecting users (roughly 2.9 million infections) through social media and netted over $2 million in click fraud in a 12 month period [10].
- **2011:** Android malware, HongTouTou, was found doing click fraud on Chinese search ad networks [11]. (With the increase of mobile devices and the existence of click fraud in the nascent, "wild-west" mobile ad space today [4], this will likely be a target for future malware.)
- **2012:** Flashflake was discovered with roughly 700,000 OS X infections, doing click fraud on search [12]. Sierefef, also known as ZeroAccess, automatically generated 32GB/month in fraudulent click traffic on each infected machine [13].

This list is far from comprehensive, but merely illustrates the assertion that some percentage of malware monetizes this way.

Since 2000, advertisers and 3rd party traffic quality companies have used data analysis to detect click fraud. The exact techniques and heuristics are not disclosed. But some basic standard metrics could include: average daily clicks, average page views per click, date/time/referrer of the click, conversion rate per click (CRC), anomalies in click trends [7]. Advertisers can get more data on their landing page to better determine click fraud such as IP, geo-location, OS, browsing behavior (viewing seconds, mouse movements, etc.) In addition to passive data monitoring, advertisers can proactively target likely conversion audiences. They can avoid geographic locations with higher incidence of known click fraud; they can target keywords that get less click fraud; they can select times slices that improve conversion rates.

Ad networks also have heuristics, also not disclosed. Though they do have rules to detect double clicks and they do monitor statistical anomalies in over hundreds of metrics [3], there is limited data available.

So for malware today to monetize it must bypass these unknown heuristics and detection analyses; in other words it must appear human generated.

3. Approach

The purpose of our investigation was to ascertain, by dynamic analysis, the mechanics of how a specific malware executed click fraud. This analysis would also help answer who benefitted from the malware's activity and how much click fraud occurred on the ad network because of this malware. This was also a prototypical case by which we could evaluate whether this merited further investigation and what type of structure, processes and collaborations we would need in order to create an automatic click-fraud prevention system.

4. Hapili - a case study

Hapili is the internal research name we gave a particular malware, that was labeled as part of the Tracur family, which was found in April 2012. Hapili comes from the name of the publisher that benefited most from the subverted clicks. Though this code is obfuscated, encrypted and employs anti-

emulation and anti-debugging techniques, its operation is straightforward. It consists of user-mode DLL, *dpvdx.dll*, resident on the file system. There exists a registry entry, *HKCU\Software\Microsoft\CurrentVersion\Run\Update*, that executes rundll32.exe on *dpvdx.dll* when the machine boots.

After the machine boots, *Dpvdx.dll* is running in a non-hidden process that uses SetWindowsHookEx to inject itself into IE. *Dpvdx.dll* unpacks/decrypts itself and gets data about the victim's machine including its name and volume information. It uses these data and creates a GUID which it stores in the registry under *\Software\Nktwlbksk*, this is used to identify the victim to the C&C. *Dpvdx.dll* hooks internet connections so when a victim searches via Bing or Google it sends an encrypted blob containing the GUID of the victim machine, the search portal, the query string of the search and an affiliate ID. It seems this affiliate ID is used to track whence the malware got on the victim's machine, so the fraudulent clicks generated by this instance of the malware get credited to a particular affiliate. The C&C sends a response back that includes a referrer, a URL and an agent. These are then used when the user clicks on an ad link from the search results and one of five things can occur:

A. The user actually reaches the legitimate landing page of the ad. This seems to happen for several reasons: a debugger is attached so no malicious redirect, certain periods of time produced no redirect or this search was already redirected a few times prior. Tests were not conducted to see if the geo-location was a factor in non-redirects.
B. The redirect goes to a sub-syndicated search portal, where the user is presented the same ad links, and they have to click once more to get to where they intended.
C. The redirect goes to a non-germane site that has nothing to do with the sponsored link they clicked, but for "Work at Home", "Make $10,000 in 10hrs a week" type of sites. These appear to be advert arbitrage sites [4].
D. The redirect goes to another site which was other than what was intended but had similar content.
E. The redirect breaks, due to a C&C database bug and the user is presented with an error.

Over the course of a few weeks, by manually executing queries and clicking on sponsored ad links from Bing searches, we observed about 30% go to actual ad landing page, 20% go to sub-syndicated search portal results, 30% go to a non-germane site, and 15% to a different site of similar content. 5% were returned errors.

For the non-savvy computer user, the degradation in their search activity maybe a small enough inconvenience to go unnoticed. Since roughly 50% of the time they get search results and land where they intended with maybe an extra click, as in scenario B. Scenario B is effective and appears organic from the ad network and advertiser perspective because it is human generated and the human was interested in the ad and got to the landing page of the ad; the only problem is the wrong publisher profits from this click. Instead of the search engine rightfully earning this revenue the attacker's publisher gets the credit. The malware coopts the user's interaction to produce clicks that falsely originate from a sub-syndicate instead of from the right publisher. In cases D the malware more aggressively interferes with the user's click while trying to preserve the intent of the user's request by ending up at another site that has similar content. For C the redirect is very much non-sequitur to what the user intended.

Self-selection could be a possible reason why very little effort was seemingly spent on hiding this malware from the victim [14]. A savvy computer user would be alerted that something was amiss on the very first redirect, and would not be an easy "mark". So it is reasonable that the target for this malware is a user who would not be aware or inclined to prevent click redirection and would actively contribute to generating more clicks. The indicators of a file on the file system, a running process, a registry entry and sporadic search anomalies would go unnoticed with this target audience for years.

5. Discussion

We used three data sets for detecting this fraud: 1. Microsoft Malware Protection Center (MMPC) telemetry from a number of Microsoft security services (MSE, MSRT, Defender, etc.), 2. COSMOS

data from AdCenter and 3. data obtained via our manual analysis of hapili. Through manual investigation, we identified a few publishers and their ad units who benefitted from the search click fraud. We used data in COSMOS to further identify specific type tags being used for these manually verified fraud clicks. We then looked at IPs obtained from MMPC telemetry data of machines with *dpvdx.dll* present on the file system. These IPs, for a 21 day period between when the malware was first discovered until a signature was released, were intersected with the publishers ID, ad units and type tags, to determine the amount of misappropriated revenue perpetuated by this version of hapili. The exact number is confidential but on the order of hundreds of thousands of dollars for a 3 week period of time.

We also created a milker, not too dissimilar from that developed by [6], but much more rudimentary. We basically automated a random playback of the malware asking the C&C for URL directives. This was an effort to automate the most laborious portion of our approach. Though we were able pin-point fraudulent clicks, further work is necessary to go beyond just the first URL programmatically, since we have found this is mostly just a JavaScript redirect. Future work would entail devising a milk-crawler, which would milk the directives and then crawl the URLs to the actual landing page of the beneficiary.

Our methodology has identified click fraud for a specific piece of malware. It is those clicks that occurred during a precise period of time to publisher's ad units with type tags that we have manually confirmed as benefiting from fraud. In addition, these clicks also emanated from an IP of a machine running an MMPC product that detected the malware on the file system. While false positives (clicks identified as fraudulent which in fact are not) are possible, more investigation is necessary to quantify. A Bayesian approach, incorporating additional data points such as- the decrease in ad clicks to a particular publisher after the malware signature is released, could quantify FPs. This would quantify the likelihood that a click is not fraudulent, given an IP associated to a machine that has the malware, clicks on the same type tags of the ad units of publishers known to be benefitting from this malware during a time when the malware is active and not clicking on the same type tags after the malware was removed.

6. Conclusion

The percentage of malware engaging in some form of click fraud is significant. Data indicate this problem is far from contained, especially in the mobile space [4]. There is a need for understanding in the security community of malware that monetizes via click fraud. There are opportunities to join disparate data sets today to build prevention systems for malware-generated click fraud. This is important to consumers as well as the entire online commerce ecosystem. Since it is a large vector by which criminals can profit, it is a key area to address for a comprehensive cyber crime disruption strategy.

7. References

[1] L. Sinclair, "Click fraud rampant in online ads, says Bing," *The Austrailian,* p. 1, 16 May 2011.

[2] C. Forensics, "Click Fraud Rate Rises to 22.3 Percent in Q2 2010," p. 1, 20 Oct 2010.

[3] S. Ghosemjumder, "Stealing Clicks," *Forbes,* p. 1, 24 Sept 2007.

[4] S. G. Y. Z. Vacha Dave, "Measuring and Fingerprinting Click-Spam in Ad Neworks," *SIGCOMM,* 2012.

[5] J. S. A. M. T. R. Kenny Olmstead, "Digital: By the Numbers," *The State of the News Media 2012, An Annual Report on American Journalism,* 2012.

[6] P. P. C. G. C. K. V. P. Brad Miller, "What's Clicking What? Techniques and Innovations of Today's Clickbots".

[7] J. Stricchiola, "Click Fraud in PPC & Google," [Online]. Available: http://www.alchemistmedia.com/click-fraud.html.

[8] M. S. Neil Daswani, "The Anatomy of Clickbot.A".

[9] S. Gallagher, "How the most massive botnet scam ever made millions for Estonian hackers," 10 Nov 2011. [Online]. Available: http://arstechnica.com/tech-policy/2011/11/how-the-most-massive-botnet-scam-ever-made-millions-for-

estonian-hackers/.

[10] Imperva, "Botnets at the Gate," 2011.

[11] M. J. Schwartz, "Android Trojan Practices Click Fraud," 16 Feb 2011. [Online].

[12] S. Golovnav, "The anatomy of Flashfake," 24 May 2012. [Online]. Available: http://www.securelist.com/en/analysis/204792232/The_anatomy_of_Flashfake_Part_2.

[13] K. S. Labs, "Malware Analysis Report: New C&C Protocol for ZeroAccess/Sirefef," 2012.

[14] C. Herley, "Why do Nigerian Scammers Say They are from Nigeria?," WEIS, 2012.

[15] PwC, "Internet Advertising Revenues Set First Quarter Record at $8.4 Billion," 11 June 2012. [Online]. Available: http://www.iab.net/about_the_iab/recent_press_releases/press_release_archive/press_release/pr-061112.