Understanding the detection of fake view fraud in Video Content Portals

Miriam Marciel
NEC Labs Europe
miriam.marciel@neclab.eu

Ruben Cuevas
Universidad Carlos III de Madrid
rcuevas@it.uc3m.es

Albert Banchs
IMDEA Networks Institute
Universidad Carlos III de Madrid
albert.banchs@imdea.org

Roberto Gonzalez
NEC Labs Europe
Roberto.Gonzalez@neclab.eu

Stefano Traverso
Politecnico di Torino
stefano.traverso@polito.it

Mohamed Ahmed
NEC Labs Europe
Mohamed.Ahmed@neclab.eu

Arturo Azcorra
IMDEA Networks Institute
Universidad Carlos III de Madrid
arturo.azcorra@imdea.org

ABSTRACT

While substantial effort has been devoted to understand fraudulent activity in traditional online advertising (search and banner), more recent forms such as video ads have received little attention. The understanding and identification of fraudulent activity (i.e., fake views on videos) in video ads for advertisers, is complicated as they rely exclusively on the detection mechanisms deployed by video hosting portals. In this context, the development of independent tools able to monitor and audit the fidelity of these systems are missing today and needed by both industry and regulators.

In this paper we present a first set of tools to serve this purpose. Using our tools, we evaluate the performance of the fake view detection systems of five major online video portals. Our results reveal that YouTube's detection system significantly outperforms all the others. Despite this, a systematic evaluation indicates that it may still be susceptible to simple attacks. Furthermore, we find that YouTube penalizes its users' public and monetized view counters differently, with the former being more aggressive. In practice, this means that views identified as fake and discounted from the public view-counter are still monetized. We speculate that even though YouTube's policy puts in lots of effort to compensate users after an attack is discovered, this practice places the burden of the risk on the advertisers, who pay to get their ads displayed.

Keywords

YouTube, advertising fraud, fake views, detection mechanism

1. INTRODUCTION

Online advertising sustains a large fraction of Internet businesses. It was recently reported by The International Advertisement Bureau (IAB) to have generated a revenue of $49B in 2014, in the U.S. alone. This figure corresponds to 15.6% increase in revenue with respect to 2013. Following the trends in technology, mobile and video advertising have become an important and growing fraction of the overall online advertising market. Of particular interest to this work is video advertising. Surveys indicate that in 2013, 93% of online marketers used video to advertise their products [1], and of these, 65% used YouTube specifically to deliver adverts. This activity is estimated to have generated $3.3B in 2014, in the U.S., representing approximately 7% of the total revenue generated by online advertising [2]. This figure has risen from $2.2B in 2012, and is expected to grow to $8B by 2016 [2, 3].

Given its size, it is no surprise that online advertising attracts fraud. Current estimations indicate that 15-30% of ad-impressions are fraudulent [4, 5, 6] leading to losses in the order of billions of dollars for advertisers [7]. With respect to video ads, recent media and industry reports indicate that fraud is especially endemic in online video, c.f. [8, 9, 10]. The U.S. based marketing community body, the Association of National Advertisers (ANA), reported in Dec. 2014 that (on average) 23% of video ad views across different studies are fraudulent [11].

Researchers and practitioners have previously analyzed and proposed solutions to address click fraud in search and display ads. Notable examples include [12, 13, 14, 15]. While the goal of click fraud is in general to inflate user activity counters at a particular target, such as a webpage, online video ads offer new motivations, attack paths and revenue streams. First, the prestige associated with owning/producing popular videos is now commonly commoditized, for example it was reported that YouTube removed more than 2B suspected fraudulent views from accounts associated with the music industry [16, 17]. Second, online videos advertisers are forced to delegate the detection of fraud to the online video portals, that host their content, and have to rely on the high-level statistics these offer. In contrast, search and banner advertisers can collect partial information from their click/impression sources to identify fraudulent activity. For instance, they can register data from the delivered users (e.g., session times), ad-location, etc. Third, while ads in banner search are sold at either *Cost-per-Impression (CPI)*, or *Cost-per-Click (CPC)*, video ads are typically sold at *Cost-per-View (CPV)*,[1] which is on average more expensive than the equivalent CPIs (sold in sets of 1000, and referred to as Cost Per Mille (CPM)) [18]. This is an obvious motivation for fraudsters to prefer video-ads.

The common attack in online video ad fraud consists of inflating the view count of one (or more) video(s), using botnets or viewers recruited from cheap-labour countries [19]. In fact, it is easy to find paid services that offer to generate tens of thousands of views to videos hosted on popular platforms (YouTube, Dailymotion or Vimeo) at a low price.[2] If the goal of the attacker is to simply increase the popularity and visibility, this is enough. In the case of monetized views, the attacker translates the view inflation to revenue by attempting to have ads served to its fake viewers.

In response to the scale of the video-ad fraud, the online advertising market has consistently published the need for solution in the media [20], and the IAB has recently formed a working group to address the issues [15]. It has so far published a white-paper report on anti-fraud principles and proposed a reference taxonomy [19]. At the same time, online video platforms have acted to forestall the damage by strengthening their fake-view detection methodologies [21, 22], and publicizing their activity.

As things stand today, we lack standard methods to auditing or monitoring the performance of the fraud detection algorithm used by video platforms. This is reflected in the IAB working group white-paper which states that "*[Supply sources] are challenged by a lack of consistent and independently measurable principles on how they each should identify and expunge fraudulent traffic*" [19].

In this paper, we first propose a measurement methodology to aid in filling this gap. Employing a modular active probe, we evaluate the performance of the fraud detection mechanism (for public and/or monetized views) of 5 online video portals, namely YouTube, Dailymotion, Vimeo, Myvideo.de, and TV UOL.

Being YouTube the only portal deploying a sufficiently discriminative fake view detection mechanism, we deepen our analysis to study its key parameters. We look at the impact of manipulating only parameters that are directly accessible to viewers, including: the behavior of an IP address, such as varying the number of visited videos per-day, the views per-video, and the duration per-view. We also look at the impact of changing the browser-profile of viewers, such whether or not cookies are enabled, and the impact of mixing viewer activity in NATed traffic.

Our main findings can be summarized as follows:

Source	YouTube	Vimeo	Dailymotion	Myvideo.de	TV UOL
SYSOMOS [23]	81.9%	8.8%	4%	-	-
DATANYZE [24]	65.1%	11.1%	0.6%	-	-
NIELSEN [25]	84.2%	-	1.16%	-	-
Statista [26]	73.6%	0.9%	1.6%	-	-
Alexa	3	145	84	3236 (DE: 153)	101 (BR: 5)
Views/day (x1M)	1200	1	60	7	6

Table 1: Market share and rank of the studied portals as reported by different public sources. Views per-day estimates are collected from Wikipedia [27].

(1) Of the 5 portals we test, YouTube in the only portal to deploy a significantly discriminative fake view detection mechanism. All other portals do not sufficiently discount view counters under the simplest fake views generation configurations.
(2) A deeper analysis reveals that the view detection mechanism of YouTube's public view counter is susceptible to simple fake views generation strategies such as: using multiple values in the HTTP connection attributes (e.g., User-agent or Referrer), distributing views across multiple IP addresses, or routing views through NATs.
(3) We find a consistent and significant discrepancy between the counters reported for the same content by public and monetized view counters in YouTube; whereby monetized view counters report *at least 75% more* fake views than public view counters.

Organization of the paper
The rest of the paper is organized as follows. Sec. 2 presents the basic background on the business models and statistic reporting tools for the five online video platforms considered in this study. In Sec. 3 we present the measurement tools and the performance metrics used in this study. Sec. 4 compares the performance of the fake view detection algorithms of the online video platforms under analysis. Sec. 5 and Sec. 6 present more detailed analyses of the fake view detection algorithms used by YouTube to discount views from the public view-counter and the monetized view-counter, respectively. Finally, Sec. 7 discusses the related work and Sec. 8 concludes the paper.

2. BACKGROUND

In this work, we focus on user-generated video platforms, the most widely used and, therefore, the most susceptible to video advertising frauds. Table 1 summarizes the online-video market share of the portals considered in this study. Since YouTube is reported by all sources to be the dominant player of online video market, it will serve as the reference portal in our study.

Most user-generated video platforms monetize the content uploaded by their users through advertising. YouTube, Dailymotion, Myvideo.de, TV UOL all deliver adverts to the videos streamed to the viewers. YouTube is the only platform that allows its user to explicitly indicate whether or not to enroll her videos into the monetization programme. Users

[1] see https://support.google.com/adwords/answer/2472735?hl=en
[2] e.g., http://www.viewbros.com

who decide to participate, share the revenue from monetized views. Dailymotion implements a different revenue sharing scheme. It allows third-party web-masters to enroll in the "Dailymotion Publisher Network", so that they can monetize views to videos hosted by Dailymotion, and are embed in their sites.[3] While YouTube and Dailymotion share the advertising revenue with their users, to the best of our knowledge, Myvideo.de, and TV UOL do not. Vimeo is still different, as it runs a graded subscription based model, whereby only users paying for the 'Plus' or 'Pro' accounts are able to generate revenue from their uploads; either by using a "Tip Jar" service that enables other viewers to tip to the user (available for 'Plus' and 'Pro' accounts) or a "Pay-To-View" service in which viewers pay to watch (only available for 'Pro' accounts).

Under these revenue models, malicious users have incentives to inflate their view counters. When there is revenue sharing based on view counts, as in the case of YouTube and Dailymotion, view inflation is typically linked to the generation of fraudulent revenue. However, user view inflation is not limited to defrauding the ad system; as mentioned, numerous examples have been documented, showing that users can, and do, trade on the popularity of their uploads, cf. [16].

To enable their users to understand how viewers interact with their content, video platforms record and report several statistics. More specifically:

- YouTube provides two main sources of data on user activity (counted views): public statistics (public view counter, number of comments, likes, dislikes, number of subscribers), which are available on the video webpage, and private statistics, which include the number of counted and monetized views and are only available to the video uploader. The public view-counter is updated in real-time, while the video has less than 301 views. Once this threshold is reached, the view-counter is paused and a background verification process starts. After this initial verification process successfully ends, public view-counters are updated once every 30-120 minutes [28], and only verified views are counted.

YouTube provides separate statistics for counters on monetized content. Users have to create an AdSense account, and enrol their YouTube channel, to monetize their content. Users can then view monetization statistics in both their YouTube analytics and the AdSense accounts. In this paper we use the monetization statistics from the YouTube Analytics service, which is claimed to provide an error of less than $\pm$ 2% with respect to the actual number of monetized views [29]. In particular, the monetization statistics offered by YouTube Analytics are: (i) the estimated number of monetized views, i.e., the number of viewers who watched an associated video-ad, (ii) the estimated revenue based on the Cost per Mille (CPM), and (iii) the total gross revenue the video generated. In order to enable users to better target their contents, these metrics are available by country, date and type of ad.

Finally, YouTube Analytics service collects, and reports several finer grained statistics for viewer behaviour, including, the number of views grouped by day, country, viewer age and gender, playback location (if video is embedded in third party websites), etc. YouTube Analytics also provides users with summary reports on their channel subscribers, including their likes and dislikes, comments, etc. These statistics are updated once a day [30]. Based on our experiments Analytics counts only validated views.

- Dailymotion provides public view counts on each video's webpage, and users can access similar statistics to those offered by YouTube Analytics: number of views filtered by country, and playback location, over a selected time window. Web-masters registered with the Dailymotion monetization service can access monetization statistics: the number of impressions, estimated revenue and CPM. However, these statistics are aggregated across all videos associated to a web-master's account. Hence, statistics per-video are not available.

- Myvideo.de and TV UOL provide public view counters only. This data can be accessed through either the website associated to the video or the user account.

- Vimeo offers public statistics for each video including the number of views, likes and comments as well as their weekly evolution. Vimeo Basic accounts have only access to the public statistics, whereas Vimeo Plus and Pro accounts have access to advanced statistics [31], including geographical information about the views, information about users commenting or liking the video, etc.

In summary, different platforms offer statistics with different level of detail on their counted and monetized views. Overall, YouTube is the one offering most extensive statistics whereas Myvideo.de and TV UOL provide very limited statistics.

3. MEASUREMENT TOOLS, PERFORMANCE ASSESSMENT METHODOLOGY AND DATA PROCESSING

In this section we present the tools that we have implemented to generate views and fetch statistics automatically, as well as our methodology to evaluate the effectiveness of video portals in detecting fraudulent views. We note that these tools and methods are a valuable contribution in themselves, and can be easily adapted to evaluate new video portals beyond the ones considered in this paper.

3.1 Measurement Tools

To evaluate the performance of the fake-view detection mechanisms deployed by the different online video portals, we implement probes that generate automatic views under well defined constraints, and log the results of their activity. In addition, we build tailored web crawlers to collect

[3]Note that users cannot monetize their videos directly. However, web-masters can embed any video hosted by Dailymotion in their website and monetize the views attracted

Module	Description	Default Value
User-agent	Set the user-agent for a session (e.g., Firefox or Chrome).	Linux/Firefox
Referrer	Set the referrer for a session. Options are Facebook, Twitter, YouTube Search (specific for the case of YouTube), and Direct Link.	Direct Link
Cookies	When enabled, all the views have the same cookies.	Disabled
View duration	Duration of a video view (in seconds). Options are i) fixed time or ii) samples from an exponentially distributed random variable with mean the duration of the video.	Fixed (40 secs.)
Wait time between views	Vary the view inter-arrival time (seconds). Options are a Poisson process, or a constant. Zero indicates a burst.	Constant factor of the number of daily views

Table 2: Description of our software modules and their default setting

the number of views counted and monetized by the different video portals.

Automatic Views Generation: We implement a Selenium [32] based modular probe to simulate the actions of viewers on the different portals. This tool is able to load a given video-page, and we can easily configure it to perform certain viewer-like actions such as interacting with the objects in the page or varying the duration of video views. Viewer-like behaviors are implemented as dynamically configured modules, which enable us to flexibly explore the parameter space that might be used to categorize the actions of viewers. Table 2 summarizes the list of available modules and their default settings.

In order to accurately monitor and isolate the impact of our experiments on the portals tested, each probe generates views only to videos that we upload for our experiments. So as to reduce background noise resulting from real users stumbling upon the videos, the names and descriptions of all experiment videos are set to random hashes, and all external links to them are removed. We measure the scale of the background noise resulting from our approach by uploading 209 videos to YouTube. We observe that they only attract 21 views from external users over a three-month measurement period.

To scale our experiments, where specified, we utilize transparent Squid [33] proxies set up in ~100 public IP addresses in Spain and Germany as well as in 300 PlanetLab nodes [34]. These proxies relay views generated by probes running in our local datacenter. From our experiments, we determine that YouTube treats direct and transparently proxied requests equally.

In the rest of the paper, we will indicate the specific configuration of the probe used to conduct each reported experiment.

Fetching Statistics from Video Portals: In order to retrieve the statistics reported by each portal, we implement tailored web crawlers. These allow to: (i) collect the information from the videos' public view counters and (ii) login in the user (i.e., uploader) account and retrieve the number of counted views for the video as well as the number of monetized views (if available).[4] In particular, for Myvideo.de, TV UOL and Vimeo we collect information on the number of counted views whereas for YouTube and Dailymotion we retrieve also the reported statistics on the number of monetized views.

3.2 Performance Assessment Methodology

To establish the reliability of the view counters, and compare the different portals, we measure their accuracy in detecting fake views and report their false positive rates. For some specific portals we also report their false negative rates.

False positives: From the perspective of a view counter detection mechanism, a *false positive* is defined to be a 'fake view' that is misclassified and counted in the view counter (public or monetized). To measure the rate of false positives for a given portal, a probe generates views to a given video on a given portal and retrieves the number of counted views from the statistics offered by the portal. The false positive rate (R_{FP}) for the given platform is then defined as:

$$R_{FP} = \frac{\text{\# counted views}}{\text{\# `probe' generated views}}$$

False negatives: Again from the perspective of a view counter detection mechanism, a *false negative* is defined as a 'real user' view labelled as fake from the system, and not counted in the view counters. To measure the rate of false negatives for a given portal, we embed videos into webpages that we can control and monitor. We then request collaboration from real users to visit these pages, and watch the videos. The experiment tracks the number of visitors accessing each webpage, and the number of videos they watch. The false negative rate (R_{FN}) for the given platform is then defined as:

$$R_{FN} = 1 - \frac{\text{\# counted views}}{\text{\# `real-user' generated views}}$$

3.3 Data Processing

The results obtained from measuring systems in the wild typically show a high level of variability. Our case is not an exception. The sources of variability in our study may come from our measuring architecture, (e.g., a failing proxy) or from temporal transient behaviors of the detection mechanisms of video portals. While identifying failures in our probes and removing them from the data is (relatively) easy, identifying temporal transients in the behaviour of the detection mechanism and filtering them is challenging. Indeed our experiments show evidence of the presence of systematic and random temporal transients in the detection mechanism of some portals. For instance, when our probe generates automatic views to a YouTube (our reference video portal)

[4] The crawler also login in the web-master account in the case of Dailymotion.

Trace	Period	Length	IP addresses	Views	Videos
YT-1	01/03/13-30/04/13	2 months	28071	3.94M	1.37M
YT-2	01/05/13-30/11/13	7 months	16781	15.9M	3.95M

Table 3: The measurement traces containing YouTube video sessions we obtain from a residential network of an European ISP, and their main characteristics.

video, we observe that there is a trend (very clear if the rate of configured rate of views per day is high) for YouTube to count more views in the first few days of the experiment, then to later adjust to a lower stable false positive ratio in the rest of the experiment. Moreover, we also observe some experiments with a low false positive ratio in all the execution of the experiment that show a peak in the false positive ratio in an unique day.

In order to identify the standard behavior of fake view detection mechanisms of video portals and get rid of the most common transient events as those described above, we compute for each experiment the daily false positive ratio and calculate the median across the days of the experiment. Note that the median is robust to outliers that in our case are represented by temporal transient events. For simplicity we refer to this metric as R_{FP} across the paper. Finally, we take care of repeating all experiments numerous times to provide statistical confidence in the results reported.

4. COMPARISON OF FAKE VIEW DETECTION MECHANISMS OF MAJOR PORTALS

In this section, we investigate how views are counted in the different portals. We first compare the fake view penalization methods employed by the different portals in their public view counters. We then study how the two portals that share their revenue (YouTube and Dailymotion) count views for monetized content.

4.1 Counting views in public view counters

Rate of False positives: We start by looking at the rate of false positives for public view counters. We set up a simple experiment whereby a single probe, with default parameters, from a fixed IP address, varies only the per day number of views it generates to a given video on a given portal. In particular for each of the five portals, we generate 100, 400 and 500 views per day, to a targeted video. Each experiment runs for eight days and we repeat it three times.

In order to understand whether the number of views that we generate corresponds to normal user behavior, we collect information from a large number of YouTube sessions to give us a baseline for the normal user behavior in the network. To do this, we replicate the methodology described in [35] and collect two independent datasets (from the residential network of an ISP) that contain millions of YouTube sessions. We refer to these datasets as *YT-1* and *YT-2* and we summarize their main characteristics in Table 3. Our traces

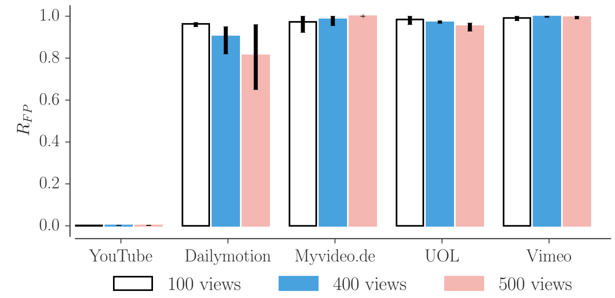

Figure 1: Comparison of the false positive ratio among the studied portals.

indicate that none of the IP addresses in *YT-1* and *YT-2* perform more than 100 daily views to a single video. Since YouTube is the most popular among the studied portals, we assume that the configured number of views per day represents an aggressive setup for all the portals.

The results of this experiment is reported in Figure 1. Main bars represent the average R_{FP} value from the three experiments and error bars report the max and min R_{FP}, while the different colors correspond to the different daily view rates. Our results indicate that YouTube operates the most discriminative detection mechanism, and that it is significantly more effective than the other portals. It penalizes **all** the views from the probes. In contrast, Dailymotion counts as valid almost all the views when the daily rate is 100, and 93% (85%) of them when the rate is 400 (500) views per day. Myvideo.de, TV UOL and Vimeo deploy detection mechanisms that capture $<$ 5% of probes fake views, even for the most aggressive configuration.

In summary, we observe that YouTube implements the most discriminative fake view detection mechanisms, and is able to easily detect obviously aggressive behaviors. Surprisingly, other portals have serious problems to discriminate fake view even under the considered aggressive setups.

Rate of False Negatives: To evaluate the R_{FN} for the different portals, we embed videos from the different portals into webpages we control, and track the number of real users accessing the pages and watching the embedded videos as well as the duration of their views. Further, we compare the impact of sourcing users via social media and online crowdsourcing platforms. In the case of social media, we announce the URL of the webpage on Facebook and Twitter and request collaboration from our contacts and friends to visit our pages and watch the embedded videos. In the second case, we use a crowd-sourcing website to recruit viewers. Finally, given the cost of recruiting large numbers of users, and the results of the previous experiment indicating that only YouTube and Dailymotion are significantly discriminative in updating public view counters, we evaluate the R_{FN} only for these two portals.

The results of this experiment are summarized in Table 4. We find that the R_{FN} is reasonably small for the two portals

Platform	Experiment	# performed real views	# counted views	R_{FN}
YouTube	Social Media	330	322	2,4%
	Crowd-sourcing	599	537	10,3%
Dailymotion	Social Media	325	290	10.9%
	Crowd-sourcing	587	515	12.2%

Table 4: False Negative Ratio for the two conducted experiments for YouTube and Dailymotion.

under both user sourcing approaches ($< 12\%$). Hence, we conclude that the fake view detection algorithms of YouTube and Dailymotion are fairly effective at identifying views generated by real users. However, we note that Dailymotion shows a larger R_{FN} for the first experiment.

Finally, it is worth noting that the data provided by YouTube and Dailymotion in both experiments shows a typical spatially localized distribution of viewer visits. Viewers are mostly localized in a specific geographical region; for the social media experiment, most of the views come from Spain, whereas, for the experiment that uses crowd-sourcing, most of the views come from India and Bangladesh. Interestingly, we observe that the social media experiment for both portals presents a smaller R_{FN}. This difference may be due to the fact that many of the views come from Spain, a country that is reported to present a very low volume of fraudulent activities in online advertising [36, 37].

4.2 Counting views in monetized view counters

Having established a baseline for how views are discounted in public view counters for (i) the actions of real users, and (ii) obviously aggressive fake view patterns. Now, we look at how this discounting is reflected on monetized content. More specifically, we focus on content we enrol in YouTube and Dailymotion monetization programmes, and we look at how actively it is monetized. Note that a given view is considered monetized only if a video advert is shown in the session and the viewer spends a given time watching the advert. Thus, to evaluate the effectiveness in counting monetized views, we count the views we generate that actually encounter an advert, and, then, using the analytics tool provided by each portal, we compare this number against the views that the portal has monetized.

To carry out monetization experiments we register several accounts and their associated videos in the monetization program of each portal. We remark that Dailymotion only monetizes videos embedded in external webpages. Therefore, we create external webpages to embed our videos. The webmaster accounts for these pages are then associated with uploader accounts, and the associated videos are then enrolled into the monetization programme. For YouTube, the process is more straightforward. To monetize videos, the uploader must register her channel to AdSense, Google's monetization platform, and indicate which videos to enrol. In the case of YouTube, our software generates views to the URL of the

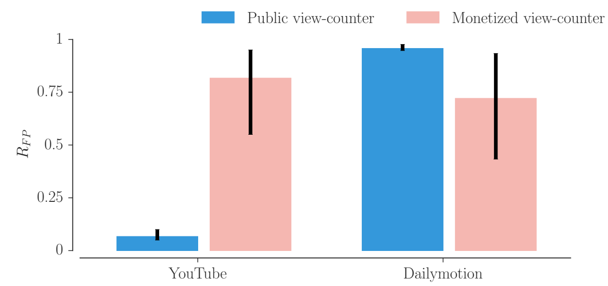

Figure 2: Comparison of false positive ratio for the number of views in the public and monetized view counters for YouTube and Dailymotion.

video webpage. For Dailymotion, views are directed to the external webpages we build to embed our videos. Based on preliminary experiments, we observe that Dailymotion does not show an advert to every automatically generated view. Then, to properly log the number of generated views subject to be monetized, we have developed a tool, based on optical character recognition (OCR), to detect whether the view received an advert.[5]

We run these experiments using the default configuration of our probe, which emulates a deterministic behaviour, and is set to perform 20 views per day to a targeted video from a single IP address. We repeat the experiment 4 times with a duration of 20 days for both portals.

From our *YT-1* (*YT-2*) traces we know that less than 0.04% (0.01%) of monitored IPs perform more than 20 views per day to a single video. We therefore consider that our setup is aggressive and expect our fake views to be easily identified. Moreover, as monetized fake views translate to direct costs to advertisers, we expect both portals to be strict in the identification of fraudulent views for monetized content. We check this assumption by comparing the R_{FP} between YouTube and Dailymotion. Note that for each trail, we configure the probe to view the video and any video advert it encounters completely.

Figure 2 compares the R_{FP} in the number of fake views counted by the public and monetized view counters, in each experiment, for both YouTube and Dailymotion. Again, the main bar depicts the average value across the experiments, and the error bars give the min-max value of the R_{FP} across the experiments.

Dailymotion shows the expected behavior, i.e., it discounts a larger number of fake views from the monetization view counter (avg. R_{FP} = 72%) as compared to the public view counter (avg. R_{FP} = 97%). Despite this improvement, the detection mechanism for monetized fake views still presents a poor performance since roughly 3 out of each 4 fake views are monetized even under the aggressive configuration of our experiment. Surprisingly, YouTube results

[5]As ads display some text indicating the remaining time of the ad, by detecting the presence of such text we can identify that the ad has been displayed.

are in contradiction with our initial expectations. Indeed, YouTube applies a weak control to remove fake monetized views (avg. R_{FP} = 82%) despite having in place a relatively efficient mechanism to remove fake views from the public view counter (avg. R_{FP} = 7%). This unexpected result has been reported previously by YouTube users.[6] The YouTube support team has explained that discrepancies may be due to users watching the video advert but not the video, and, when this happens, the view is monetized but not counted by the public counter. However, this is not our case, as we instrument the software in these experiments to load both the video advert and the video completely. Another potential reason could be that YouTube performs part of its view validation procedure in a post-hoc manner, rather than in real-time [38]. However, in our experience, 6 months have passed since the time at which we run the experiments, and we have not observed any compensation in the statistics provided by YouTube. Furthermore, while such an approach would provide the flexibility of enabling YouTube to retroactively compensate for attacks that are discovered, we speculate that it would place the burden of the risk on the advertisers, who would have to bid to get their adverts displayed to customers without relying on stable and validated statistics.

From the above experiments, we would like to highlight that: First, we have not received any money while running these experiments, and all the statistics we report are those we retrieve from the YouTube Analytics channel page and the Dailymotion Publisher page; Second, we have reported our findings to YouTube and Dailymotion and plan to present their feedback and explanations, once we receive them.

In summary, the analysis conducted in this section shows that, among the studied online video portals, YouTube is the only one implementing a sufficiently discriminative fake view detection mechanism. However, our results reveal that YouTube only applies this mechanism to discount fake views from the public view counter and not from the monetized view counter. In the rest of the paper, we present advanced methodologies to further understand the functionality of the fake view detection algorithm of YouTube for public (Section 5) and monetized (Section 6) view.counters.

5. YOUTUBE'S FAKE VIEW DETECTION MECHANISM FOR COUNTED VIEWS

In this section we explore different aspects we believe YouTube may employ to detect fake views in order to unveil some fundamental components of the fake view detection system. We acknowledge that the spectrum of parameters and their interactions within YouTube's detection mechanism may be too large to make a full exploratory exercise. Instead, we analyze the impact of a subset of meaningful parameters, that other detection mechanisms can monitor, and are easy to configure for a user.

[6] https://plus.google.com/100368302890592068600/posts/1sEuu94EjuV

5.1 Parameters used in the detection

We leverage the modularity of the probe to define different configurations to isolate the impact of different parameters on the fake view detection mechanism. In our experiments, each probe instance uses a single public IP address chosen from our pool and performs 20 views per day to the same video for 8 consecutive days. Based on our results in Section 4.2, we expect that this behaviour is aggressive enough to trigger the fake view detection mechanism of YouTube.

Next we describe the probe configurations we use in our experiments. Note that unless specified, we set all parameters to their default values given in Table 2.

- Deterministic (D): The goal of this configuration is to define a simple, deterministic, and, thus, atypical view pattern. The configuration eliminates any randomness by setting to constant values the view time (40 secs.) and the time between views (72 mins.). All other parameters take their default values from Table 2. We expect this configuration to be easily identified.

- Vary view burst (B): Having observed the impact of changing the number of views that a video receives, we design this configuration to look at the impact of making views in bursts. In particular, the probe runs the **D**eterministic configuration zeroing the time between consecutive views, thereby generating a burst of 20 consecutive views every day. The time between bursts is configured to 24 hours. Since views in bursts from a given IP address are atypical, we expect this configuration to show a false positive rate lower than the deterministic configuration.

- Vary inter-view wait time (P): The goal of this configuration is to measure the impact of varying the time between views over a day. In particular, this probe runs the **D**eterministic configuration, but varies the time between two consecutive views to make it follow a Poisson process (λ = 20). In essence, this configuration aims to determine whether simply adding noise to the views per day rate of the **D**eterministic configuration has impact.

- Short Views (SV): The goal of this configuration is to measure the impact of making very short views to videos. This is a behaviour that is known to be utilized by bots and paid-for mass view services. In particular, this probe runs the **D**eterministic configuration, but sets the duration of video views to 1 sec. Since consecutive short views is atypical of users, we expect to see this configuration to be heavily penalized.

- Cookies (CK): The goal of this configuration is to measure to what extent the detection systems rely on cookies to identify users. Therefore this configuration performs all views using the same cookie.

- Complete (C): The goal of this configuration is provide a baseline by emulating some real-user like features. As such, it enables all the modules in Table 2, but cookies. Specifically, the view duration (λ = duration of the targeted video) and wait time between views (λ = 20) are selected from Poisson processes. The Referer and User-agent fields are se-

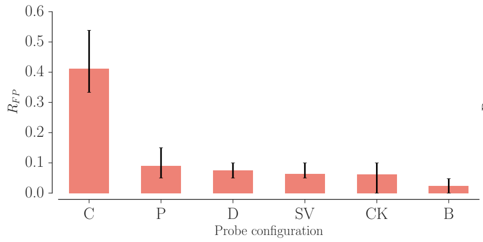

Figure 3: False positive rate obtained for each one of our experiment configurations.

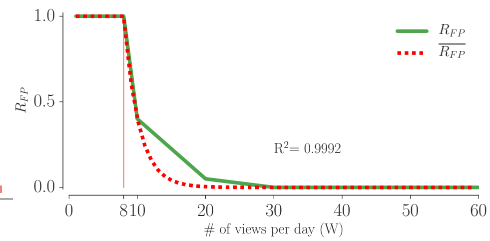

Figure 4: The false positive rate and its fitting model when increasing the number of daily views performed by a single IP address to a single video.

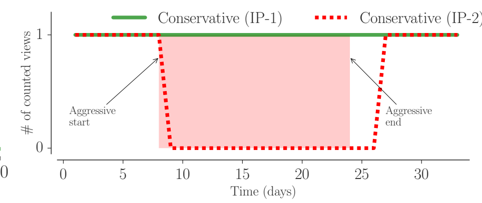

Figure 5: Number of views counted by YouTube for both IP-1 (running only a *conservative* probe) and IP-2 (running a *conservative* probe and an *aggressive* probe).

lected randomly. Given the variation in the parameters, we expect this configuration to be the least penalized.

For each configuration we run the experiment 5 times and evaluate the R_{FP} (again from the perspective of the view detection mechanism). The results are given in Figure 3; the main bar and the error bars represent the average, and max/min R_{FP}. Each bar represents a different configuration. As expected, the **C**omplete configuration yields the highest false positive rate (≈40%), which is on average more than 4 times larger than the other configurations, with average $R_{FP} < 10\%$. This indicates that adding randomness to basic HTTP connection parameters such as the User-agent, or the Referer makes it significantly harder for YouTube to detect fake views.

Looking at the impact of varying the wait-time between views (**P**, **D** and **B**), we observe some differences. Specifically, the detection mechanism penalises bursty behavior the most heavily, discounting 98% of the views from the **B**urst configuration. Moreover, comparing the **D**eterministic and the **S**hort **V**iews configurations, we observe that the detection mechanism ignores the view duration despite our expectation that very short view duration would be an obvious candidate for the identification of fake or (at least) not worthy views. Finally, the detection mechanism does not leverage browser cookies in the detection of fake views since the differences in R_{FP} in configurations with (**CK**) and without (**D**, **SV**, etc.) cookies are negligible.

In summary, *the results in this subsection reveal that attacks performed using a static configuration of HTTP connection parameters are easily identified by the fake view detection mechanism of YouTube, which effectively removes more than 90% of fake views in this case. Instead, adding some variability in these parameters would increase the effectiveness of the attack by ∼30%.* While these results explain the false positive rate difference between the considered configurations and our benchmark, they do not explain the 60% discounted fake views common to all our configurations. The only aspect which is common to all our configurations and which may be responsible for such large penalization is that they perform their views from a unique public IP address. This, along with the fact that IP addresses are one of the strongest online users identifiers [38] and one of the key parameters many security online services rely on [13, 39, 40], leads us to conclude that the video viewing pattern from an IP address is a key element for the fake view detection mechanism of YouTube. We analyze this hypothesis further in the next subsection.

5.2 Influence of Video Viewing Pattern in the detection

In this section we analyze the response of YouTube's detection mechanism to the fake view patterns of an IP address. We first look the impact of fake view patterns when aimed at a single video, then explore the cases for a single IP viewing multiple videos, and finally a single video receiving views from multiple IP addresses.

One video, One IP address

We start our analysis by examining how YouTube discounts the views a single IP address generates to a single video. In particular, we are interested in understanding how the view discounting threshold(s) are triggered, when varying the number of views per day. We conduct a simple experiment, in which our probe generates $W = [1, 4, 7, 8, 9, 10, 20, 30, 40, 50, 60]$ views per day to a given video for 8 days. We use the previously defined **D**eterministic configuration for this experiment.

The results of this experiment are presented in Figure 4.[7] It reports the R_{FP} for the different values of W. We observe that the detection system counts all the fake views up to a rate of 8 per day. From 9 views on, the R_{FP} decays exponentially and zeroes for more than 30 daily views. We can model the R_{FP} with respect to the views per day (W) as an exponential decay function with the following parameters, with an R^2=0.999:

$$\overline{R_{FP}}(W) = \begin{cases} 1 & \text{if } W \leq 8, \\ e^{-0.454517n} & otherwise \end{cases}$$

In the above experiment, all videos are new, uploaded for the experiment. In order to understand whether this has any

[7] We repeat this experiment 3 times from IP addresses located in Spain and Germany obtaining similar results.

impact on the results obtained, we look at the response of the detection mechanism when the videos are already known by the system and moderately popular. We repeat the experiment using two videos,[8] with roughly 12K (100 in the last month) and 300K (5K in the last month) registered views over few years. To differentiate the activity of our probe, we configure it to use very rare user-agents (specifically, Bada, HitTop, MeeGo and Nintendo 3DS). Before starting the experiment, we validate (using YouTube Analytics) that the targeted videos have not received views from the selected user-agents in the previous 6 months.

Repeating the above experiment, with $W = [8, 9, 10, 20]$ views per day, with new IP addresses, we find that the fake view detection mechanism again starts discounting views from 8 views per day, for a given IP address. This suggests that the threshold chosen by the fake view detection algorithm of YouTube is fixed regardless of the popularity of the video.

Multiple videos, One IP address

Having observed how the views from a single IP address to a single video are penalized, we now look at the response of the fake view detection system when a single IP address distributes its views over several videos. Given the previous result, we expect that aggressive IP addresses will be heavily penalized, independently on the number of videos they target.

In the following we first validate this hypothesis with a simple experiment. We then present the results of a large scale measurement study we run to characterize the R_{FP} as function of the overall behavior of an IP address, with respect to the number of visited videos, and views performed.

We start defining two types of simple probe behaviors, with respect to the number of videos, and views per day, which we refer to as *conservative* probe and *aggressive* probe. The *conservative* probe performs 1 daily view to a given video from a single IP address whereas the *aggressive* probe performs 30 views per day. We set up the following experiment: in two IP addresses, IP-1 and IP-2, we launch an instance of the *conservative* probe to a different video for 34 days. Moreover, in IP-2, we also launch an instance of the *aggressive* probe, starting at day 8 and stopping at day 24, while there is no aggressive probe in IP-1.

Figure 5 gives the number of views the detection mechanism counts over time, for the *conservative* probes in the two IP addresses. Since these probes perform just 1 view per day to the video, we expect to see either 1, if it is counted, or 0, if its penalized. We find that the detection mechanism counts all the views from the *conservative* probe in IP-1, while it penalizes the *conservative* probe in IP-2 from day 9 to day 26. We observe that view penalization starts 24 hours after we launch the *aggressive* probe in IP-2, and ends two days after we stop the probe. Repeating the experiment twice more, we obtain the same results. From this, we conclude that YouTube's fake view detection mechanism penalizes IP

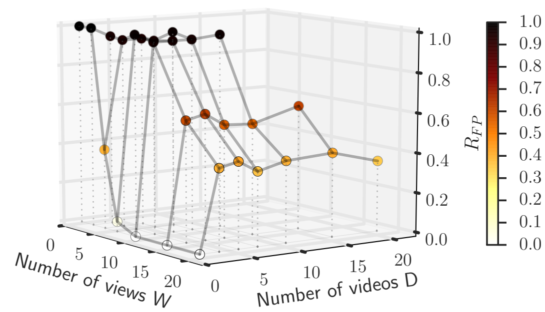

Figure 6: R_{FP} for several combinations of the number of views W and the number of watched videos D.

addresses globally for their behavior, rather than for individual videos.

Observing that the behaviour of an IP address is tracked across video views, we now look at characterizing the penalization factor YouTube applies to an IP address when it varies the number of videos viewed. We conduct a large scale experiment in which we perform $W = [1, 3, 5, 7, 10, 15, 20]$ views per day, uniformly distributed across $D = [1, 3, 5, 7, 10, 15, 20]$ videos over a period of 7 days.[9] In total, we ran 28 trails combining the different numbers of views and of watched videos. For each experiment we use a different PlanetLab node as proxy and use the **D**eterministic configuration of the probe.

Figure 6 reports the R_{FP} for each one of the 28 considered combinations. Looking at the evolution of R_{FP} for a fixed number of videos, we observe the exponential decay found earlier in every case. However, we notice a variation of the threshold in the number of overall daily views that defines the start of the exponential decay. It seems to be set to (at least) 10 views for $D \geq 1$, whereas we know from our previous results that it is 8 views for $D = 1$. If we now consider the evolution of R_{FP} for a fixed number of daily views, we observe that when we concentrate all views in a single video, the punishment is much more severe than when they spread across two or more videos. Moreover, the differences in R_{FP} for the cases of 2 or more videos are relatively small ($\leq 7\%$).

One video, Multiple IP addresses

Having established, that an IP address is tracked across the views it makes, we now look at the response of the fake view detection mechanism, when distributing the views to a given video across several IP addresses. To this end, we use 70 different PlanetLab nodes, and divide them in 3 independent sets of different size $N = [10, 20, 40]$. We assign each set of nodes a different video on YouTube, and configure each PlanetLab node to generate views to its correspond-

[8]We obtain video uploader permissions to conduct this experiment.

[9]Note that we only run experiments for $W \geq D$. For instance, in the case of $W = 5$ we run experiments for $D = 1, 3, 5$.

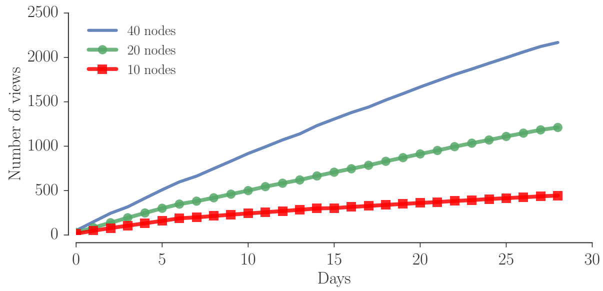

Figure 7: Number of views in the public view counter for three different experiments using 10, 20 and 40 PlanetLab nodes.

ing video. We again utilize the **D**eterministic configuration of the probe, and report the results with each PlanetLab node set to generate 3 views per day. Overall, our experiment generates 30, 60 and 120 views per day to a video, which should result in $R_{FP} = 0$, if coming from a single IP address.

Figure 7 gives the temporal evolution of the aggregate number of views counted by the YouTube public view counter for each of the three experiments. As shown, the growth in number of views over time is linear for all configurations, and we observe an overall $R_{FP} > 73\%$ in the three experiments. This indicates that distributing activity across multiple IP addresses results in a substantial increase in the R_{FP} enabling attackers to inflate view counters easily.

This experiment suggests that *YouTube is vulnerable to attacks that employ many IP addresses* (like, e.g., the ones performed using botnets), as such attacks can apparently achieve an arbitrarily large number of views. Indeed, it is easy to find paid services that offer to inflate the view counter of YouTube (and other video platforms) videos up to tens of thousand views in a short period of time and at a low price (e.g., http://www.viewbros.com).

5.3 Impact of NATed IP addresses on the detection mechanism

As NAT devices aggregate traffic, they typically contain the video viewing activity coming from multiple, usually private, IP addresses. In large NATed networks such as campus networks, corporate networks and, in some cases, ISP networks, this activity may be significantly large. Therefore, in these sets of experiments, we investigate how the fake view detection mechanism of YouTube penalises the views coming from NATed networks. To this end, we install our probe on three machines accessing the Internet from NATed networks located at three different institutions and we configure them to perform 20 (Institution 1), 75 (Institution 2) and 100 (Institution 3) daily views for a period of 8 days. We remark that we use the **D**eterministic configuration, in which we disable the usage of the cookies to ensure that the fake view detection system has no means (under our control) to identify the viewing sessions our probes generate.

Table 5 gives the R_{FP} for each experiment along with some information of the different NATed scenarios. Note that, although our probe generates views rather aggressively, the R_{FP} is surprisingly large in all cases. This suggests that the YouTube's fake view detection mechanism has problems to properly identify malicious activity coming from NATed networks. To confirm this finding, we separately analyze working days and days off (i.e., weekends and holidays) in Institution 2, running the experiment for 194 days. Note, that working days register a high volume of traffic in the NATed network whereas in days off the traffic is typically low. Figure 8 shows the distribution of daily false positive rate for working days and days off using boxplots. The results confirm that YouTube discounts almost all views during days off, i.e., when our traffic is more exposed, but have serious problems to discount views (median R_{FP} = 60%) during workdays, i.e., when our views are hidden by larger volumes of traffic. Hence, this suggests a fraudster can dramatically increase the efficiency of its activity by gaining access to machines located behind large (active) NATed networks, e.g., a public campus network.

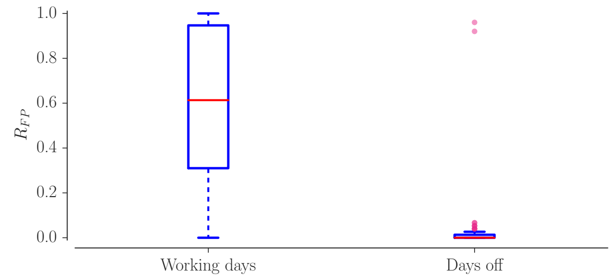

Figure 8: Distribution of daily R_{FP} for working days and days off for Institution 2

Experiment	W (views/day)	# U (users behind the NAT)	U/W	R_{FP}
Inst. 1	20	∼50	∼ 2.5	0.9
Inst. 2	75	∼100	∼ 1.33	0.82
Inst. 3	100	∼50	∼ 0.5	0.36

Table 5: R_{FP} and information about the four scenarios for the experiments we conduct from NATed IP addresses.

In summary, *the fake view detection mechanism of YouTube implements an exponential discount factor of the number of views performed from a single IP address that increases with the rate of views. However, our results show that some simple modifications in a fraudster's strategy can increase very substantially the false positive rate: i) adding some randomness in the HTTP connection attributes such as the User-agent or the Referer, ii) distributing the malicious activity across multiple IP addresses, or iii) performing fake views from NATed networks.*

6. YOUTUBE'S FAKE VIEW DETECTION MECHANISM FOR MONETIZED VIEWS

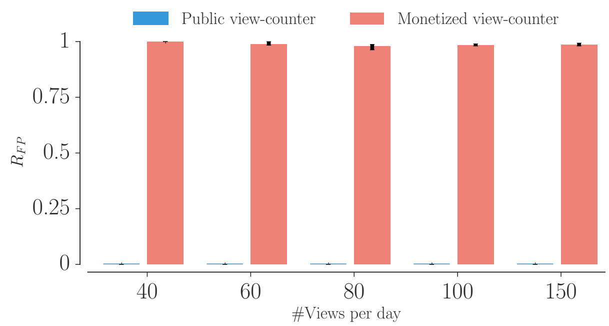

Figure 9: Comparison of false positive ratio for the public and monetized view counters of YouTube for different daily rates of generated views W.

The results in Section 4 surprisingly imply that YouTube monetizes (almost) all the fake views we generate, while discounting them from the public view counters. To further understand this seemingly anomalous behavior, in this section we study in more detail the detection mechanism applied to monetized views.

We re-use the configuration described in Section 4.2 for YouTube, and conduct a new set of experiments, whereby we increment the number of daily views our probe generates from a single IP address, to a single video. In particular, we set $W = [40, 60, 80, 100, 150]$ to cover a wide range of aggressive configurations. We conduct each experiment for 10 days and repeat it three times. Figure 9 reports the R_{FP} for both the public and the monetized view counters. Again, the main bars and error bars represent the average and the max/min R_{FP}, respectively.

We find that the monetized view counter's detection mechanism penalizes a negligible portion of views in all the considered configurations (avg. $R_{FP} > 83\%$), while the public view counter's detection mechanism penalizes most of the fake views. These results confirm the preliminary observation in Section 4; that YouTube applies different penalization schemes to the fake views in the monetized and public view counter, with the former being much more permissive than the latter.

6.1 Counting monetized views from the advertiser's perspective

To gain insight into the monetary implications of the above finding, we design a tailored experiment in which we assume the role of an advertiser exposed to fraudulent views. To this end, we first create an advertiser account using the Google AdWords service. AdWords enables us to configure advertising campaigns in YouTube, so that our video ads can target YouTube videos whose uploaders participate in the monetization programme. We then create a video ad and build an advertising campaign to target our own videos that we previously uploaded to YouTube. In this way, we play both the role of the advertiser and the publisher in our campaigns, and can build a complete picture of the trade.

Adwords offers a wide range of tools to aid in the design of video advertising campaigns. Advertisers can tailor campaigns to reach specific YouTube viewer demographics (per interests, country, language, gender, age), or target specific YouTube videos. With the aim of checking if YouTube actually charges advertisers in presence of fake views, we configure a campaign to target the views from the countries where our proxies are located (accepting all languages, genders and ages) and headed to our experiment videos. Then, we use our probe to generate views to these videos.

YouTube deploys a sophisticated bidding algorithm that selects in real-time the ad to target to a specific video. Briefly, this algorithms implements a variant of a Vickrey auction, named Generalized Second-Price auction [41] for which the winner (advertiser) pays the price of the second highest bid. Note that winning bids vary over time and targeted videos. In addition to the bid price, the algorithm also considers other parameters including the profile of the viewer watching the video, the advertiser's daily budget, the price of the advertiser's bid, etc.

In setting up this experiment, we faced several challenges to configure a successful campaign able to target a large number of ad views. Our initial trails were unsuccessful; we used a small daily budget of 50€ and our campaign had an unusual configuration, since it targeted few very specific and relatively unpopular videos. To overcome this, we took advice from an online advertising expert to: i) increase our bidding prices per ad view up to 10-15€ (the recommended bid price for YouTube was 0.04-0.05€); ii) configure the video uploader's Adsense account to accept only the specific type of ads defined in our campaign; iii) configure two or more campaigns that compete for the same videos (viewers); iv) to vary the pattern of views to the videos more.

Having done so, we launched new experiments, whereby we targeted a set of videos from different IP addresses and different rates of views per IP address (between 10 and 70 views per IP address). In particular, our campaigns target 14 videos, using the **D**eterministic configuration of our probe. Of the 14 trails, 5 videos were able to attract ad views from our campaigns, meaning that we bid and won; in affect our ads were targeted to our uploaded videos, and watched by our probes, which are configured to view in full any ad target, as well as the video.

Table 6 summarizes the main characteristics of the view-pattern configuration of these videos. Moreover, it shows the number of monetized ad views from our campaigns, as well as the number of counted views in the public view counter for the day of the experiment in which more ad views were delivered. We observe that the number of monetized views in Videos 1 and 2, is larger than the number counted for the public view counter, i.e., views considered suspicious are removed from the public view counter, but monetized. For Video 3, out of the 80 views performed, YouTube's public view counter recorder 36, and monetized (and charged us for) 29 views. While we cannot certify whether each one of these 29 monetized views belong to a counted or a penalized view, results suggest that some of the views actually monetized were not considered legitimate in the public counter.

At the time of writing this paper, in total 145 ads from our campaigns were shown in Videos 1-3 to our probe, for which we were charged 10.51€. To check whether the discrepancy between the public and monetized view counters is resolved when payment is made, we proceeded to pay our bill and certified that YouTube withdrew the amount from our credit card. This confirms, that YouTube appears to initially monetize views that are removed from the public view counter (e.g., those from Video 1 and 2). Nevertheless, Google indicates through its AdWords support website[10] that *"If we find invalid clicks that have somehow escaped our automated detection in the past two months, we'll give you credit for these clicks"*. However, it is still worth noting that it appears the burden of risk from fraud is being initially carried by the advertiser. We intend to report any changes to this circumstance in a revised version of this paper.

For the most aggressive experiment configuration of Video 4, we find that 15 of our video ad views were delivered. We notice that these views were initially added to the bill of our advertiser account. However, 5 days after the first ad view was delivered, YouTube rightfully labeled the probe's activity as suspicious and suspended the video uploader account in AdSense.In addition, YouTube has notified us via email of the suspension of the uploader's account due to suspicious activity. Finally, the ad views associated to fake views were removed from the advertiser account. We believe that the peculiar setup of our campaign, coupled with the aggressiveness of the experiment triggered some alarm in YouTube's detection mechanism. Note that as reported in Table 6, we repeated this experiment twice obtaining similar results.

In summary, we conclude that *YouTube uses a seemingly permissive detection mechanism to discount fake monetized views. This exposes advertisers to the risk of building their advertisement campaigns on unreliable statistics, and may make them initially burden the risk of fraud. Conversely, the public view counter is much more discriminative, demonstrating that YouTube has effective means to identify fake views. Our results also reveal that whenever the permissive threshold for the detection of fake monetized views is crossed, YouTube severely penalizes the uploader of the video by suspending her Adsense account, preventing the uploader from monetizing any of the videos associated to the suspended account.*

7. RELATED WORK

The research community has devoted an important amount of effort to the identification of malicious behaviors in online services and to the design of counter-measures to such behaviors [42, 43, 44]. Similarly to YouTube's fake view detection mechanism, most of the detection system designs rely on the IP address as the main id to track and identify malicious behavior. Some examples of such mechanisms are the classical monitoring tools looking for sources of attacks, such as port scanning [45] and DDoS attacks [46], or the detection systems which counteract malicious users in P2P applications [47]. Only those systems requiring the user registration to gain access to the service, e.g., Online Social Networks, implement detection mechanisms that use both the IP address and the user-id as basic units to detect inappropriate behaviors. For instance, Facebook traces the requests pattern from a given account and if it is unusual the user is warned and if the behavior persists the account is closed [48]. In the case of YouTube, both registered and non-registered users can access to the service, although as our results suggest it seems that the detection algorithm does not make distinction between both types of users (even when cookies are enabled).

[10]https://support.google.com/adwords/bin/answer.py?answer=2549113&ctx=tltp&hl=en-US

	# IPs	Daily Views per IP	Monetized view counter	Public view counter
Video 1	1	10	8	5
Video 2	1	20	16	2
Video 3	8	10	29	36
Video 4	2	70	15 (17)	0

Table 6: Experiments configuration of videos attracting ads from our advertising campaigns. The reported numbers of monetized and public counted views correspond to the days of the experiment in which more ads where shown. The number 17 for Video 4, reflects the second trail of the experiment.

More recently, the rapid proliferation of botnets and specialized bots to attack specific services has led the research community to work on the identification, characterization and elimination of botnets and bots [49, 50, 51, 52]. Additionally, following a similar methodology to the one we use in this paper, Boshmaf et al. [53] and Bilge et al. [54] have developed their own automatic software to evaluate the effectiveness of the detection of different social networks from different types of attacks such as user impersonation.

In the field of fraud detection and mitigation in online advertising, most of the literature focuses on traditional type of ads such as search or display ads. In this case, the fraud problem is referred to as click fraud since the fraudulent activity is associated to fake clicks on ads, typically performed from bots. Metwally et al. [13] present an early study in which they use the IP address as the parameter to detect coalition of fraudulent users or *fraudsters*. In a more recent work, Li et al. [55] propose to analyze the paths of ad's redirects and the nodes found in the content delivery path to identify malicious advertisement activities. Furthermore, Stone-Gross et al. [14] managed to get access to a command-and-control botnet used for advertisement fraud in which the bot master sends commands with fake referrers. On a complementary work, Miller et al. [56] study the behavior of two clicking robots: Fiesta and 7cy. Fiesta uses a middleman that probably shares its revenue with advertiser sub-syndicates. 7cy tries to emulate a human behavior and presents different behaviors depending on the location of the infected computer. Finally, Dave et al. [57] design an algorithm to identify click fraud from the advertiser perspective; to design this algorithm, the authors propose to measure different aspects of the user behavior in the advertiser webpage such as the mouse movements or the time spent in the website. Based on their initial work, the same authors propose, implement and test ViceROI [12], a solution to discount fake clicks from ad networks. The basis of ViceROI detection algorithm is the fact that click-spammers will lead to a higher ROI (Return of Investment) than a legitimate publisher, as the authors claim that a realistic ROI is difficult to obtain with robots.

All the above works establish a very solid basis for the design of tools to mitigate fraud associated to traditional ads. However, they are (in general) not applicable to fraud associated to video-ads due to the different nature of video-ads and click-based ads. To the best of the authors knowledge, there is only a very recent study that analyzes fraud in video-ads [58] . The authors of this study use traces from a video platform in China to identify statistically outlying video-viewing patterns and, based on these observations, suggest a fake view detection algorithm built on parameters such as the number of views made from an IP address to a video or the number of different IP addresses watching a given video. Unfortunately, as the authors acknowledge, they do not count with a ground truth dataset to validate their designed solution as legitimate views cannot be distinguished from fake ones in their dataset. In contrast to this work, our study focuses on five major video portals, including YouTube, the most important video platform worldwide, and pursues a different goal. We propose a methodology to generate ground truth scenarios so that we can evaluate the performance (and unveil basic functionality principles) of different video portals' fake view detection mechanisms for both the number of counted and monetized views. As our methodology is extensible to other video platforms, the authors from [58] could use it to validate their proposed solution in their considered video platform. Finally, although less related to our work, it also is worth referring the reader to a recent work by Krishnan and Sitaraman [59], which presents a large scale analysis of the different factors that influence the effectiveness of video-ads.

8. CONCLUSIONS AND FUTURE WORK

To the best of the authors' knowledge, this work is the first one to propose a set of tools to monitor and audit the fake view detection mechanisms of online video portals, and enable independent and external parties to measure their performance. The application of our tools and methodology to the view counting behavior of five different video portals has highlighted some interesting observations. We find that only YouTube deploys a sufficiently discriminative fake view detection mechanism. All the other portals studied are susceptible to very naïve view inflation attacks. Clearly, this raises a problem for users with regard to the accuracy of the numbers that are reported by these portals.

A more careful analysis of YouTube's fake view detection mechanisms has revealed that, it is susceptible to attacks that introduce some randomness to the viewer behavior, including the use of multiple user-agents, referers, multiple IP addresses, or machines within a large NATed network. These are traits that a knowledgeable user would be able to configure easily, and we assume must be common in large scale attacks using botnets. YouTube is consistently more permissive in the counts for monetized views, when compared to the public view counters. Specifically, fake views are penalized and not counted by the public view counter, but are monetized, i.e., have paid-for ads delivered in them, and counted in the the video owner's monetized views. While YouTube is shown to strive to protect its users and clients, for example by reacting quickly when suspicious behavior is identified, we speculate that its setup seems to place an unnecessary burden of risk on clients. For example, fake views can be discounted equally for public and monetized counters, but they are not.

Finally, our analysis in this paper reinforces the call by industry for "consistent and independently measurable principles on how [Supply sources (SSPs/exchanges, ad networks, and publishers)] should identify and expunge fraudulent traffic". In future work, we intend to refine the tools developed here, and explore the possibility of making them available to the wider community.

9. REFERENCES

[1] eMarketer, "As Barriers Tumble, Video Marketing Adoption Grows." http://www.emarketer.com/Article/Barriers-Tumble-Video-Marketing-Adoption-Grows/1010374, 2014.

[2] "IAB internet advertising revenue report, 2014 full year results." http://www.iab.net/media/file/IAB_Internet_Advertising_Revenue_Report_FY_2014.pdf.

[3] eMarketer, "Online Video Advertising Moves Front and Center." http://www.emarketer.com/Article/Online-Video-Advertising-Moves-Front-Center/1009886, May 2013.

[4] Will Luttrell, "Only The Buy-Side Can Solve Our Fraud Problem." http://www.adexchanger.com/data-driven-thinking/only-the-buy-side-can-solve-our-fraud-problem/, 2013.

[5] Douglas de Jager, "Display Advertising Fraud is a Sell-Side Problem." http://www.spider.io/blog/2013/04/display-advertising-fraud-is-a-sell-side-problem/, 2013.

[6] S. Vranica, "A 'Crisis' in Online Ads: One-Third of Traffic Is Bogus." http://online.wsj.com/news/articles/SB10001424052702304026304579453253860786362, 2014.

[7] "Solve Media Survey." http://news.solvemedia.com/post/74832974631/solve-media-bot-survey-2014, January 2014.

[8] "Fraud Alert: Millions of Video Views Faked in Sophisticated New Bot Scam." http://www.adweek.com/news/technology/fraud-alert-millions-video-views-faked-sophisticated-new-bot-scam-156883, 2014.

[9] A. Kantrowitz, "Ad-Fraud Operation Fools Detection Companies, Nets Millions." http://adage.com/article/digital/ad-fraud-operation-fools-detection-companies-nets-millions/293929/, 2014.

[10] J. Kirk, "Malware campaign inflated views of pro-Russia videos." http://www.techworld.com.au/article/574002/malware-campaign-inflated-views-pro-russia-videos/, May 2015.

[11] ANA and White Ops, "The Bot Baseline: Fraud in Digital Advertising." https://www.ana.net/getfile/21853, December 2014.

[12] V. Dave, S. Guha, and Y. Zhang, "Viceroi: Catching click-spam in search ad networks," ACM CCS, 2013.
[13] A. Metwally, D. Agrawal, and A. El Abbadi, "Detectives: Detecting coalition hit inflation attacks in advertising networks streams," ACM WWW, 2007.
[14] B. Stone-Gross, R. Stevens, A. Zarras, R. Kemmerer, C. Kruegel, and G. Vigna, "Understanding fraudulent activities in online ad exchanges," ACM IMC, 2011.
[15] The Interactive Advertising Bureau (IAB), "Trustworthy Supply Chain: Anti-Fraud Working Group." http://www.iab.net/about_the_iab/recent_press_releases/press_release_archive/press_release/pr-091614, 2014.
[16] D. Gayle, "YouTube cancels billions of music industry video views after finding they were fake or 'dead'." http://www.dailymail.co.uk/sciencetech/article-2254181/YouTube-wipes-billions-video-views-finding-faked-music-industry.html, 2012.
[17] C. Hoffberger, "YouTube strips Universal and Sony of 2 billion fake views." http://www.dailydot.com/news/youtube-universal-sony-fake-views-black-hat/, 2012.
[18] " Video Advertising Playbook ." https://www.tubemogul.com/marketing/TubeMogul_Video_Ad_Playbook.pdf, 2014.
[19] The Interactive Advertising Bureau (IAB), "Anti-Fraud Principles and Proposed Taxonomy." http://www.iab.net/media/file/IAB_Anti_Fraud_Principles_and_Taxonomy.pdf, September 2014.
[20] A. Wright, "Advertising industry wages war on bots to combat online ad fraud." http://www.theguardian.com/media-network/olswang-partner-zone/2015/feb/18/advertising-industry-online-fraud-bots, 2015.
[21] "Keeping YouTube Views Authentic ." http://googleonlinesecurity.blogspot.co.uk/2014/02/keeping-youtube-views-authentic.html, February 2014.
[22] S. Dredge, "Google goes to war on 'fraudulent' YouTube video views." http://www.theguardian.com/technology/2014/feb/05/youtube-fake-views-counts-google, 2014.
[23] "A Look Inside Online Video Engagement - Part I." https://www.sysomos.com/reports/video, 2009.
[24] "Online Video market share in the Alexa top 1M." http://www.datanyze.com/market-share/online-video/, 2015.
[25] "May 2012 - Top U.S. Online Video Sites." http://www.nielsen.com/us/en/insights/news/2012/may-2012-top-u-s-online-video-sites.html.
[26] "Leading internet multimedia portals in the United States in August 2014, based on market share of visits." http://www.statista.com/statistics/266201/us-market-share-of-leading-internet-video-portals/, 2014.
[27] "Comparison of video hosting services." https://en.wikipedia.org/wiki/Comparison_of_video_hosting_services, 2015.
[28] "Reference Guide: Data API Protocol." https://developers.google.com/youtube/2.0/reference.
[29] "Ad Performance report for partners." https://support.google.com/youtube/answer/2423005?hl=en.
[30] "Views report." https://support.google.com/youtube/answer/1714329.
[31] "Vimeo - Get Advanced Statistics." http://vimeo.com/stats.
[32] "Selenium webdriver." http://docs.seleniumhq.org/projects/webdriver/.
[33] "Squid proxy server." www.squid-cache.org/.
[34] B. Chun, D. Culler, T. Roscoe, A. Bavier, L. Peterson, M. Wawrzoniak, and M. Bowman, "Planetlab: An overlay testbed for broad-coverage services," *ACM SIGCOMM CCR*, vol. 33, July 2003.
[35] A. Finamore, M. Mellia, M. M. Munafò, R. Torres, and S. G. Rao, "Youtube everywhere: Impact of device and infrastructure synergies on user experience," ACM IMC, 2011.
[36] Barry Schwartz, "Report: Click Fraud Rate For Q2 2010 28.9%." http://searchengineland.com/report-click-fraud-rate-for-q2-2010-28-9-45838, July 2010.
[37] John Brown, "The Click Fraud Report Infographic: First Half 2013." http://www.adometry.com/blog/click-fraud-report-infographic/, July 2013.
[38] L. Chen, Y. Zhou, and D. M. Chiu, "Fake view analytics in online video services," ACM NOSSDAV, 2013.
[39] M. P. Collins, T. J. Shimeall, S. Faber, J. Janies, R. Weaver, M. De Shon, and J. Kadane, "Using uncleanliness to predict future botnet addresses," ACM IMC, 2007.
[40] A. Ramachandran and N. Feamster, "Understanding the network-level behavior of spammers," *ACM SIGCOMM CCR*, vol. 36, Aug. 2006.
[41] B. Edelman, M. Ostrovsky, M. Schwarz, T. D. Fudenberg, L. Kaplow, R. Lee, P. Milgrom, M. Niederle, and A. Pakes, "Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords," *American Economic Review*, vol. 97, 2005.
[42] F. Soldo, K. Argyraki, and A. Markopoulou, "Optimal source-based filtering of malicious traffic," *IEEE/ACM Trans. Netw.*, vol. 20, no. 2, 2012.
[43] Z. Chen, C. Ji, and P. Barford, "Spatial-temporal characteristics of internet malicious sources," IEEE INFOCOM, 2008.
[44] S. Venkataraman, A. Blum, D. Song, S. Sen, and O. Spatscheck, "Tracking dynamic sources of malicious activity at internet-scale."
[45] S. Staniford, J. A. Hoagland, and J. M. McAlerney, "Practical automated detection of stealthy portscans," *J. Comput. Secur.*, vol. 10, July 2002.
[46] T. Peng, C. Leckie, and K. Ramamohanarao, "Proactively detecting distributed denial of service attacks using source ip address monitoring.," in *NETWORKING*, vol. 3042 of *Lecture Notes in Computer Science*, 2004.
[47] R. Cuevas, M. Kryczka, R. González, A. Cuevas, and A. Azcorra, "Torrentguard: Stopping scam and malware distribution in the bittorrent ecosystem," *Comput. Netw.*, vol. 59, Feb. 2014.
[48] M. Gjoka, M. Kurant, C. Butts, and A. Markopoulou, "Practical recommendations on crawling online social networks," *IEEE JSAC*, vol. 29, October 2011.
[49] A. Karasaridis, B. Rexroad, and D. Hoeflin, "Wide-scale botnet detection and characterization," UNSENIX HotBots, 2007.
[50] Y. Xie, F. Yu, K. Achan, R. Panigrahy, G. Hulten, and I. Osipkov, "Spamming botnets: Signatures and characteristics," *ACM SIGCOMM CCR*, vol. 38, Aug. 2008.
[51] K. Lee, J. Caverlee, and S. Webb, "Uncovering social spammers: Social honeypots + machine learning," ACM SIGIR, 2010.
[52] J. Zhang, R. Zhang, Y. Zhang, and G. Yan, "On the impact of social botnets for spam distribution and digital-influence manipulation," CNS, 2013.
[53] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu, "The socialbot network: When bots socialize for fame and money," ACM ACSAC, 2011.
[54] L. Bilge, T. Strufe, D. Balzarotti, and E. Kirda, "All your contacts are belong to us: Automated identity theft attacks on social networks," ACM WWW, 2009.
[55] Z. Li, K. Zhang, Y. Xie, F. Yu, and X. Wang, "Knowing your enemy: Understanding and detecting malicious web advertising," ACM CCS, 2012.
[56] B. Miller, P. Pearce, C. Grier, C. Kreibich, and V. Paxson, "What's clicking what? techniques and innovations of today's clickbots," Springer-Verlag DIMVA, 2011.
[57] V. Dave, S. Guha, and Y. Zhang, "Measuring and fingerprinting click-spam in ad networks," *ACM SIGCOMM CCR*, vol. 42, October 2012.
[58] L. Chen, Y. Zhou, and D. M. Chiu, "Analysis and detection of fake views in online video services," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 11, 2015.
[59] S. S. Krishnan and R. K. Sitaraman, "Understanding the effectiveness of video ads: A measurement study," ACM IMC, 2013.