It's Free for a Reason: Exploring the Ecosystem of Free Live Streaming Services

M. Zubair Rafique*, Tom Van Goethem*, Wouter Joosen*, Christophe Huygens*, Nick Nikiforakis†
*iMinds-DistriNet, KU Leuven
{*zubair.rafique,tom.vangoethem,wouter.joosen,christophe.huygens*}@*cs.kuleuven.be*
† Department of Computer Science, Stony Brook University
nick@cs.stonybrook.edu

***Abstract*—Recent years have seen extensive growth of services enabling free broadcasts of live streams on the Web. Free live streaming (FLIS) services attract millions of viewers and make heavy use of deceptive advertisements. Despite the immense popularity of these services, little is known about the parties that facilitate it and maintain webpages to index links for free viewership.**

This paper presents a comprehensive analysis of the FLIS ecosystem by mapping all parties involved in the anonymous broadcast of live streams, discovering their modus operandi, and quantifying the consequences for common Internet users who utilize these services. We develop an infrastructure that enables us to perform more than 850,000 visits by identifying 5,685 free live streaming domains, and analyze more than 1 Terabyte of traffic to map the parties that constitute the FLIS ecosystem. On the one hand, our analysis reveals that users of FLIS websites are generally exposed to deceptive advertisements, malware, malicious browser extensions, and fraudulent scams. On the other hand, we find that FLIS parties are often reported for copyright violations and host their infrastructure predominantly in Europe and Belize. At the same time, we encounter substandard advertisement set-ups by the FLIS parties, along with potential trademark infringements through the abuse of domain names and logos of popular TV channels.

Given the magnitude of the discovered abuse, we engineer features that characterize FLIS pages and build a classifier to identify FLIS pages with high accuracy and low false positives, in an effort to help human analysts identify malicious services and, whenever appropriate, initiate content-takedown requests.

I. Introduction

Despite the growth and popularity of the Internet in the early 90's, transmission of sound and video over the Internet remained a huge challenge. It was not until 1995, when technology was able to cope with the requirements of online streaming, that the world's first live streaming event, a baseball match between the New York Yankees and the Seattle Mariners, was broadcasted by Progressive Networks [7]. Ever since, the online video utilization has risen massively, now with a million minutes of video traversing the Internet every second [38].

This massive consumption and endorsement of online video brought with it the rise of extremely popular services for free live streaming (FLIS). FLIS services enable free viewership of video content, albeit typically without the consent of a content owner, of TV channels and live events for Internet users. These services manage infrastructure to facilitate costless anonymous broadcasting of live streams, and maintain websites to index links for free live streams.

Like any other widely embraced online video service, the emergence of FLIS has given lift to digital copyright infringements. In fact, multibillion-dollar industries have been directly affected by these services. As a case in point, an estimated cost of FLIS to a particular soccer league was more than 15 million dollars per year, and that was only because of a single free live streaming website [31], [45]. It is also worth to note that TV broadcasters had invested more than 3 billion dollars for the exclusive rights of this league's event. Same is the case with other events where billions of dollars were invested to acquire the broadcasting rights that are illicitly being monetized by the FLIS services [16], [24]. These incidents highlight the extent of the damages FLIS services are causing to TV broadcast and related industries.

Apart from the copyright infringements, there is another serious, and practically unexplored, threat imposed by the FLIS services to their users i.e., deceptive exploitation for monetary gains. To date, FLIS services have been analyzed mostly from a legal perspective [28], [50]. However, to the best of our knowledge, there has been no study that systematically analyzes the workings of different FLIS parties and empirically assesses the threats for everyday users of FLIS services.

To this end, we argue that a careful analysis and thorough understanding of FLIS services is necessary for effectively combating them. It can enable the take-downs that will disrupt the free illegal live streaming operations [8], [9], [43], the identification of parties facilitating anonymous free broadcasts of live streams, and it is critical for shedding light on the malicious practices used to monetize the FLIS business.

In this paper, we highlight the negative effect of FLIS on users and expose the infrastructure of the FLIS ecosystem. Particularly, we target web based sports-specific FLIS services. The reason to target these services is that they are immensely popular[1], constantly emerging [16], [24], and often reported for copyright law violations [31], [45]. To uncover the FLIS ecosystem and quantify the threats to FLIS users, we conduct the following three-pronged analysis:

First, we develop an infrastructure that enables us to (1) gather unknown FLIS webpages by leveraging the infrastructure of search engines and, (2) inspect network traffic to identify the parties providing *media servers* for free anonymous broadcasting of live streams. Using our infrastructure, we identify more than 23,000 FLIS webpages corresponding to

[1] The most popular FLIS domain, rojadirecta.me, we analyze have a global Alexa rank of 1,553 with an estimated 8 million visits on this site monthly.

NDSS '16, 21-24 February 2016, San Diego, CA, USA
Copyright 2016 Internet Society, ISBN <TBD>
http://dx.doi.org/10.14722/ndss.2016.23030

5,685 domains. Next, we perform more than 850,000 visits to the identified FLIS domains and analyze more than 1 Terabyte of traffic to identify the parties providing media servers. Of these identified parties, we notice that 64% have been reported at least once for violating the copyrights of respective owners. Additionally, our investigations reveal that the FLIS services host their infrastructure predominantly in Europe and Belize. For instance, we discover that nearly 25% of the inspected free live sport streams were broadcasted from media servers hosted in Belize, and more than 60% of identified streams were broadcasted from media servers located in Switzerland, Belize, the Netherlands, Sweden, and Canada.

Second, through a series of automated and manual experiments, we find that FLIS services are involved in substandard advertisement practices, possible trademark infringements, and deceptive exploitations, targeting their users as well as TV broadcasters and sports organizations. Among others, by analyzing video *overlay ads* and 30,354 advertisement websites, we show that the users of FLIS services are often exposed to deceptive, unavoidable, and malicious ads. Our analysis reveal that one out of two ad websites, presented to the FLIS service users, is malicious in nature, offering malware (*zero-day* in one case), showing fake law enforcement messages to collect purported fines, and luring users to install malicious browser extensions. Additionally, we unintentionally find seven FLIS domains distributing malware disguised as as an application to watch free live streams on mobile devices.

Last, given the intensity of possible copyright violations and discovered threats, we present a FLIS classifier that aims to classify the FLIS webpages both effectively and efficiently. Instead of relying on signature-based techniques, the FLIS classifier models representative attributes of FLIS pages which achieve a high detection accuracy with a negligible false positive rate. Our data gathering infrastructure demonstrates a real-world utilization of our classifier where it was deployed in an online process to identify unknown FLIS pages. As an application, our classifier can be readily used by law enforcement to find previously unknown FLIS websites that can then be analyzed for potential abuses.

II. An Overview of Free Live Streaming Services

In this section, we map the ecosystem of free live streaming services. The FLIS ecosystem, as shown in Figure 1, consists of three main parties: *channel providers*, *aggregators*, and *advertisers*. We arrived at this model through the analysis of investigated FLIS services, and through the recording of common attributes. We now provide an overview of the identified FLIS parties, followed by a discussion of their business model (Section II-A), and their tactics for monetizing user views (Section II-B).

Channel Providers are the entities that provide the infrastructure to facilitate live streaming on the Web. Specifically, the channel provider maintains a media server that can be used by anyone for free. The purpose of the media server is to receive a live video stream from a remote machine (*origin machine*) and broadcast it to a wide range of viewers on the Internet. The origin machine can be operated by the channel provider itself or it may belong to a different third-party. For instance, a miscreant can digitally capture a live TV broadcast of any sport event and stream it online anonymously for free, thanks to the media server maintained by the channel provider.

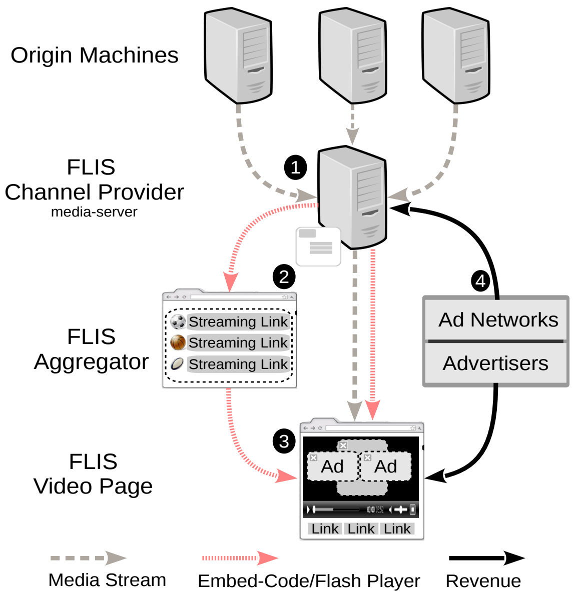


Figure 1: Overview of the operational model of FLIS. ❶ The channel provider maintains a media server to freely broadcast live streams received from the origin machines and provides embedding-code with a Flash player. ❷ The aggregator catalogs embedding-codes and index links of live streams on its webpage. ❸ A visitor lands on the aggregator's page and clicks on the indexed links, redirected to the FLIS video page on the aggregator's domain, and finds the ads displayed on the video player by the ad networks. ❹ Money flows from the advertiser to the ad network, the aggregator, and the channel provider.

To facilitate a free-of-cost live streaming service, channel providers usually maintain a simple web interface (e.g., jjcast.com, biggestplayer.me). When a user wishes to broadcast her stream for free, she creates a *channel* (on air broadcast) on the channel provider's web interface. After creating the channel, she gets a *media-server URL* to use with her origin machine for transmitting the media traffic through the channel provider's media server. Along with the media-server URL, she also receives *stream-embedding* code to place on her page, at the position where she wants the live stream to appear. The stream-embedding code is an HTML snippet or JavaScript code that creates the HTML snippet, in the form of an `<iframe>` element. It usually contains a customized Flash player from the channel provider along with the necessary configurations to broadcast the live stream. The general format of the stream-embedding code given by channel providers is as follows:

```
<script type="text/javascript"
src="http://channel-provider.com/
embed.php?stream=stream_key
&width=650
&height=450">
</script>
```

Aggregators catalog the stream-embedding codes, usually from different channel providers, and index links of various free live streams in their webpage. In other words, they provide a single site to watch numerous live events and TV channels for free. When a user lands on one of these websites, she is typically offered 2-3 links per live event. Once a user clicks the link, she is redirected to the *FLIS video page*, hosted on the same domain, where the stream-embedding code executes. This execution renders the Flash player on the page and automatically starts a live stream broadcast from the channel provider's media server.

In general, channel provider services and aggregator webpages can be maintained for both legitimate and illicit purposes. In this paper we focus on the entities that enable live streaming of the sports events and sports TV channels for free. Hereafter we use the term channel providers and aggregators to refer exclusively to those parties that manage infrastructure to broadcast free live sports events and index links of free live sports streams on their webpages.

A. The FLIS Ecosystem

We illustrate the FLIS ecosystem in terms of the business model of different parties involved.

Channel Providers. As mentioned earlier, the channel provider supplies the stream-embedding code for free live streaming. This code, along with the Flash player, carries additional JavaScript code from an ad network. This script displays ads on the top of the Flash player as overlay ads using `<iframe>` elements. The overlay ads are images or Flash content that "overlays" the video content and runs concurrently with the live stream [13]. Usually, the channel providers generate revenue from the overlay ads through cost-per-thousand (CPT) or click-through-rate (CTR) reporting metrics. While CPT is calculated by dividing the cost of an overlay ad placement by the number of impressions (expressed in thousands) that it generates, CTR is measured as the ratio of the number of times an overlay ad was displayed to the number of times it was clicked. We observe that, to maximize their profit, channel providers often include Javascript code from different ad networks. As an outcome, FLIS viewers have to interact with various overlay ads superimposed on top of each other, usually displayed in the middle of the player.

Aggregators. Like channel providers, the main source of revenue for aggregators is through advertisements. Aggregators use a variety of ad techniques that include pop-unders, pop-ups, and even overlay ads on the Flash player[2]. The aggregators include remote JavaScript code from ad networks that examine the composition of the aggregator page and present ads from other third-party sources. In addition to CPT and CTR, aggregator webpages also earn revenue through cost-per-click (CPC), i.e., the aggregator domain receives a commission from the ad network each time a user clicks on the delivered ad.

Advertisers. Advertisers and ad networks are the lifeblood of the FLIS ecosystem. As mentioned before, both channel providers and aggregators include JavaScript code from ad networks to monetize their operations. The ad network's code fetches and displays ads from different advertisers on top of the Flash player. If a user clicks on any of these ads, the website of the corresponding advertiser is opened, typically redirected through the ad network's tracking procedure. The advertiser will pay the ad network for the visitor, who, in turn, will pay the publisher (in this case either the channel provider or the aggregator) based on the pre-negotiated payment model i.e., CPT, CTR, or CPC.

B. Monetizing User Views

First and foremost, FLIS services employ deceptive techniques for monetary gains. Millions of users utilize the FLIS services in order to watch live sport events. As such, earning money from this massive user base is the key objective of the FLIS parties. To fulfill this objective, FLIS services make heavy use of substandard and deceptive advertisement techniques to monetize their business at the expense of user security. As a case in point, a user of FLIS webpage typically encounters a number of malicious overlay ads that are stuffed on the video player. These ads are usually loaded with a number of deceptive techniques. One such technique is to emboss the video player with fake close buttons. This technique can deceive a user to naively click on the fake button, potentially exposing her to malware-laden websites. In Section IV, we analyze several types of abuse and show the kind of deceptions and infections a user can experience while using FLIS services.

Additionally, FLIS parties are repeatedly reported for copyright infringements. As the owners of the broadcasting rights, sports organizations and TV channels hold exclusive rights on any broadcast of their games online. Law enforcement agencies can detect and block any domain or IP address that is involved in the broadcast of illegal sport streams based on their respective territorial jurisdiction [12], [16], [43], hence making it difficult to continue the lucrative FLIS business. Therefore, hiding behind third-parties and using a location with a flexible, or non-existent, jurisdiction are the usual practices of the FLIS parties. Moreover, the FLIS parties often take advantage of certain territorial laws by claiming that they are not involved in *direct copyright infringements* [28], [47]. Aggregators claim to only index links to the live sport streams, and channel providers argue that they only appear as a media server providers that transmit streams of an unrelated third party. In Section IV, we analyze the hosting preferences of the FLIS parties, their concealment of ownership, and the copyright removal requests submitted against them.

III. Data Gathering and Identification

In this section, we first describe how we identified the aggregator domains by leveraging search engines and explain the different phases of our data collection. We then present our methodology for identifying channel providers by analyzing the network traffic of live streams obtained through the crawling of numerous FLIS video pages.

A. Gathering Aggregator Domains

As a starting point to discover aggregator webpages, we searched for the string "free live sport streaming" in Google. From the search results, we manually identified 500 aggregator pages that index links to watch live sport streams for free.

To increase the coverage of our analysis and find new aggregator webpages, we opt for a *guided search* approach, i.e., an approach "guided" by the knowledge of known aggregator webpages (*seeds*) to find new ones. We chose this approach as

[2] Since the Flash video player is rendered on the aggregator page, the aggregator can display overlay ads on the top of the player regardless of already present overlay ads that were served through the channel provider's stream-embedding code. This is typically achieved by using the spatial coordinates of `<iframe>` responsible for rendering the Flash video player.

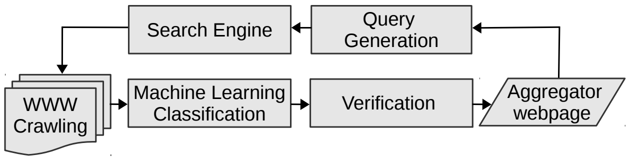


Figure 2: Guided search approach to find aggregator domains.

it has been proven effective in the context of finding unknown malicious webpages [29], [39] and because it leverages the infrastructure of search engines (such as Google) that have indexed a great part of the web.

Figure 2 shows the work-flow of our approach. We use the 500 manually verified aggregator pages as seeds and visit each page using a crawler based on Selenium[3], a testing framework for web applications, while storing the HTML of every loaded `<frame>` and `<iframe>` element, acquiring all images in the page, logging network traffic, and taking a screenshot of the webpage. In a next step, we leverage the Google search engine to find pages that contain similar attributes as the known aggregator pages. To do this, we use the crawled data of the seed pages to extract search queries that, when submitted to Google, return URLs that are likely to be aggregator pages. However, not all URLs returned from Google will necessarily belong to aggregator pages. Thus, as a next logical step, we crawled each URL returned by the search engine and employed a novel classifier to filter the non-FLIS aggregator pages in an automated fashion. However, this filtering is not perfectly accurate, and still requires manual verification of the webpages to eradicate any classification errors. Therefore, as a final step, we manually checked several hundred classified aggregator pages and separated the verified pages for our subsequent analysis. We now provide details on the different phases of our data gathering approach.

Query generation. This phase aims to extract relevant queries from the crawled data of seeds to search for new aggregator pages. To do this, we extracted terms from the known aggregator pages that are highly indicative of FLIS services, and can be used as subsequent search engine queries to find URLs of pages that are likely to be FLIS aggregator webpages. The problem here lies in the fact that aggregator pages, in general, hardly contain any text except for the links for live streaming. As a result, it is not always possible to extract strings from the main body of an aggregator page. Therefore, we focus on interpreting the `<meta>` elements, located in `<head>` container of HTML, that provide structured information about the page in specified tags. We noticed that almost every seed page includes a `<meta>` element named `keywords` that contains terms highly relevant to the FLIS aggregator pages. For instance, some of the interesting keyword terms we observed in the seed aggregator pages were: *firstrow, myp2p, rojadirecta, atdhe,* and *ilemi*. All these terms are highly indicative of FLIS aggregator pages and proved to be effective queries for discovering a wide variety of unknown aggregator pages. To this end, we extracted the keyword strings from the 500 seed pages and submitted them as queries to Google. The results returned by Google were stored for further analysis.

Classification. Overall, submitting the extracted queries to Google yielded nearly 500,000 URLs. To quickly filter the non-aggregator pages, we designed and employed a novel classifier to automatically identify pages that are likely to be FLIS aggregator webpages. We first trained a model (using the known aggregator pages) by extracting several representative features, some of which take advantage of the inherent nature of the FLIS services. We provide details on the extracted features and classifier in Section V. Once the model is trained, we use it to identify aggregator webpages by crawling the URLs that were acquired from the search engine results. For each crawled webpage, the engineered features are extracted and passed to the trained model for classification, which outputs a score indicating the URL's relevance to the FLIS aggregator page. If the score is greater than a given threshold, the model labels the URL as an aggregator page. In order to gather accurate data with high confidence, we set a threshold value that results in an outcome false positive rate of nearly 10^{-3} with a detection rate of more than 90%.

Verification. To verify the results of the classification phase, we manually checked several hundred labeled URLs. These URLs are randomly selected from the discarded pages as well as from the pages that were labeled as aggregator webpages. Through this process we aim to limit the false positives as much as possible for our subsequent analysis.

Limitations. Our data collection methodology that leverages the infrastructure of search engines has two main limitations. First, it is possible that search engines may not index FLIS webpages that violate copyrights laws. Second, the effectiveness of finding new aggregator webpages is dependent on the quantity and diversity of the known pages that we used to generate queries for the search engines. This can be improved by considering a larger and more diverse collection of aggregator pages. Additionally, we can analyze other sources to find new aggregator domains, like social media and public fora for FLIS. However, we found that these sources are rare, usually outdated, and only provide links to a selection of famous free live streaming webpages.

B. Identifying Channel Providers

In order to investigate the channel providers that maintain the media servers for free live streaming, we first sampled the 1,000 most popular aggregator domains (according to the Alexa ranking) from a set of verified aggregator pages. We then crawled each sampled domain and extracted the streaming URLs indexed on the page by interpreting the `<a>` elements. Accordingly, we filtered any URL that belongs to an advertisement network. Afterwards, we crawled the remaining URLs by sandboxing the instances of our crawler, each in a separate Linux Network Namespace. This sandboxing allows us to capture the network traffic of live media streams from an individual webpage, while crawling multiple pages in parallel. At the same time, we instructed our crawler to interpret every loaded `<frame>` or `<iframe>` in the page recursively[4], store their HTML, source URLs, and their location and size on the webpage.

[3] https://code.google.com/p/selenium/

[4] We use the Selenium's ChromeDriver with args **--disable-web-security** to disable the Same Origin Policy while crawling the webpages.

```
(Invoke) "connect"
(Transaction ID) 1.0
(Object1) {
 app:"live", flashVer: "LNX 16,0,0,296",
 swfUrl: "http://popeoftheplayers.eu/atdedead.swf",
 tcUrl: "rtmp://rtmp.popeoftheplayers.eu:1935/live",
 fpad: false, capabilities: 9947.75,
 audioCodecs: 3191, videoCodecs: 252,
 videoFunction: 1 ,
 pageUrl: "http://popeoftheplayers.eu/crichd.php?
id=35&width=600&height=450",
 objectEncoding: 3.0}
```

Figure 3: An example RTMP **connect** message used to identify the channel provider popeoftheplayers.eu.

Detecting live streams. Once the crawling process is finished, we inspected the network trace of every crawled webpage to find channel providers transmitting live media streams. Inspecting network traffic to find channel providers (or embedded players for media streaming) is much more precise than inspecting Flash embedded objects, `<iframe>` and `<script>` elements. This is because Flash embedding is frequently used for other purposes than video, such as small applications, games, and audio. Similarly, the identification of live streams via the use of specific iframes and scripts would require us to compile a whitelist of non-malicious resources.

By focusing on network traces, we can inspect the network trace of every crawled webpage by crafting a set of network signatures for various media streaming protocols and their variants, e.g, RTMP [5], HLS [1], and RTSP [6]. Our goal is to identify the presence of media traffic after the page load, and to find the channel provider transmitting these streams. We crafted these signatures to capture the protocol keywords present in the media streaming protocol messages (e.g., `connect` in RTMP [5]) [22], [40], [41]. In addition, we also built signatures to detect the `MIME` types specific to the streaming protocols that are based on HTTP (e.g., `Content-Type: application/vnd.apple.mpegurl` corresponds to HLS protocol). We then applied these custom-built signatures to all TCP and UDP connections in a network trace of the crawled page regardless of ports used. This allows us to analyze media streams that are transmitted using standard protocols on non-standard ports or streams that are encapsulated in plain-text protocols (e.g., RTMP tunneled in HTTP etc.). We found that both of these practices are common in the investigated FLIS services.

Interpreting media sources. Once the network signature matches, we try to automatically extract a source of the media server to identify the channel provider facilitating the FLIS. We found that the majority (85.7%) of media streams identified by our signatures were broadcasted using unencrypted variants of RTMP. In this case, we search for an RTMP `connect` message that is used to establish a network connection between the client (i.e., Flash player) and the media server. Figure 3 shows such a message sent to the channel provider popeoftheplayers.eu. This message contains, among other parameters, `swfUrl`: the URL of the Flash player, `tCurl`: the URL of the media server, and `pageUrl`: the URL of the page in which the Flash player was rendered. Note that the `pageUrl` in RTMP `connect` messages is the source URL of `<iframe>` that embeds a Flash player on the page.

Aggregators		Channel Providers	
Seeds	500	Inspected domains	1,000
SE URLs	513,324	URLs visited	859,126
URLs classified	23,549	Streams found	52,469
Unique domains	5,685	Channel providers	309

Table I: Summary of dataset gathered and analyzed.

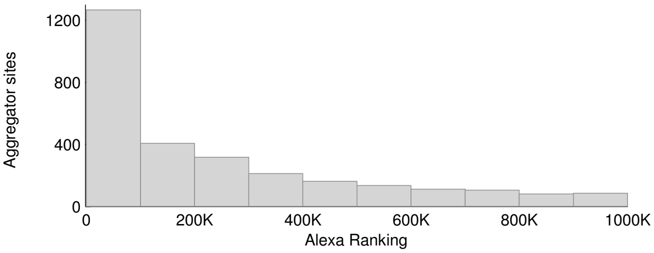


Figure 4: Distribution of FLIS aggregator websites in the Alexa top 1 million websites.

The other media streams that our signatures identified were broadcasted using HLS (8.1%) and encrypted RTMP (6.2%) from the media servers managed by just two different channel providers. In these cases, we use the HTTP `Host` header and source IP to identify the channel provider.

Dataset summary. We now provide a summary of the dataset gathered and analyzed in our study. The figures are summarized in Table I. We used 500 manually inspected aggregator pages to generate search engine queries, and subjected 513,234 URLs for classification of aggregator domains. We detected more than 23,000 aggregator URLs which correspond to more than 5,000 aggregator domains. We manually verified all the aggregator domains and performed more than 850,000 visits on the selected top 1,000 aggregator domains, analyzing more than 1 Terabyte of traffic. Our network signatures identified 52,469 media streams broadcasted from the infrastructure of only 309 channel providers. The modest proportion of streams identified is due to the fact that our crawler visited the URLs indexed on the aggregator pages and not all of these URLs correspond to pages that were broadcasting live streams when being visited by our crawler.

IV. Analysis of FLIS services

In this section, we analyze several aspects and practices of FLIS services. We start by inspecting the gathered dataset and provide insights about the ownership and hosting preferences of the FLIS parties. Next, we use *Google Transparency Report* to measure copyright removal requests submitted against FLIS parties. Furthermore, we design various methodologies to inspect and report: possible trademark abuse in FLIS domains; substandard, unavoidable, and deceptive advertisement setups; unexplored threats to FLIS users, such as unknown malware, fraudulent money grabbing scams, malicious browser extensions; and link hijacking threats.

A. *Operational Insights*

FLIS popularity. From the collected 5,685 aggregator domains, 50.74% were part of Alexa's top 1 million websites ranking. Figure 4 shows the distribution of the collected

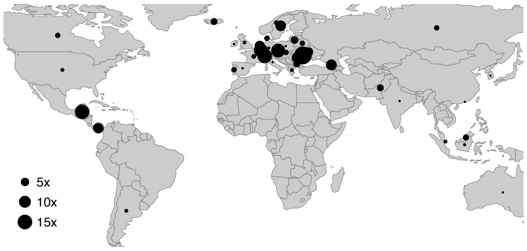


Figure 5: Relative distribution of the geographical location for aggregator websites. The size of each dot indicates the number of times a country is more prevalent in the distribution of aggregator websites compared to the distribution of the top 100,000 websites.

Hosting Company	Hosting Country	AS Number	# CP	% Streams
privatelayer.com	Switzerland	51852	14	11.7%
koddos.com	Belize	199636	11	24.2%
ecatel.net	Netherlands	29073	10	12.7%
ovh.ca	Canada	16276	10	1.2%
portlane.com	Sweden	42708	7	10.5%

Table II: Top hosting companies' infrastructure employed by the channel providers for hosting their media servers.

aggregator domains across the ranks of Alexa. As we can see from graph, nearly 22% of the aggregator domains are part of Alexa's top 100,000 websites. In fact, the most popular aggregator domain in our corpus is rojadirecta.me, having a global Alexa rank of 1,553 with an estimated 8 million monthly visits[5]. Our findings confirm that the aggregator domains are immensely popular and millions of users visit these pages to watch daily updated free live streams.

Aggregator ownership. To get an understanding of aggregator domains ownership, we performed a WHOIS lookup of the top 1,000 aggregator domains. We found that 760 out of the 1,000 domains have anonymized WHOIS records. For the remaining 240 domains, 52 belong to 6 distinct groups based on an identical name, email address, and organization name in their WHOIS record. The largest 2 groups contain 15 and 14 domains respectively, with all the domains in each group resolving to two specific IP addresses. Interestingly, the majority of the domains in both groups have the string *"firstrow"* common in their name. Overall, we were able to find 194 distinct domain owners to which the 240 domains belong.

Aggregator hosting location. In order to gain insights on the preferred geographical locations of aggregator domains, we computed the relative distribution of countries in which aggregator domains are hosted. More concretely, we performed a GeoIP lookup for all the aggregator domains and compared the distribution of hosting location to the distribution in the top 100,000 Alexa domains, which we used as a baseline. For the 10% of aggregator domains that resolved to an IP address of CloudFlare[6], we used two techniques which are known to disclose the original IP address of a CloudFlare-protected domain [35]. Using both techniques, we managed to uncover the actual IP address of 233 aggregator domains. The other aggregator websites for which we could not uncover the IP address, were excluded from the geographical distribution.

Figure 5 shows the relative distribution of the geographic location of aggregator services. From this graph, it is clear that, relative to the baseline distribution of the Alexa 100,000 websites, the distribution of aggregator websites is centered mainly around Europe. For instance, we found that the republic of Moldova appears approximately 20 times more frequent in the distribution of aggregator services. Similarly, the prevalence of several other European countries–including Switzerland, Czech Republic and Luxembourg–is more than ten times higher than in the distribution of the top 100,000 domains. Outside of Europe, we found Belize and Panama to be the most popular hosting locations for aggregator domains.

Channel provider ownership. Similar to the aggregator domains, we performed a WHOIS lookup on all the 309 identified channel provider domains. We found that 220 out of 309 channel provider domains have anonymized WHOIS records. Based on identical name, organization, and email in the WHOIS records, we were only able to identify a single group with four channel providers that belongs to the same owner. In addition, we observed that 31% of channel providers use the CloudFlare's services. Unlike aggregator domains, finding a real IP address of a channel provider is trivial. We inspect the media traffic and in 98% of cases a real IP was found by resolving the media-server domain, e.g., rtmp.popeoftheplayers.eu in Figure 3.

Channel provider hosting. Once we have the original IPs, we used myip.ms, an online hosting information service, to determine which hosting company has been delegated the IP address of a channel provider's media server. Table II shows the top 5 hosting companies employed by the channel providers including each company's geolocation, the assigned autonomous system (AS) number, the number of channel providers (CP) using the service, and the percentage of total streams originating from its infrastructure. We can see that more than 60% of the analyzed streams originate from the media servers provided by only 5 companies. Most of these companies are based in Europe. Outside of Europe, koddos.com, with all channel provider's servers in Belize, accounts for nearly 25% of all the observed media traffic. Other prevalent companies we found employed by more than one channel provider are located in Canada, Czech Republic, Romania, Ukraine, and the United States.

Overall, our analysis reveals that a significant number of FLIS parties have anonymized WHOIS records or use CloudFlare to conceal their hosting infrastructure. Moreover, the FLIS parties usually prefer Europe and Belize to host their infrastructure. One reasonable explanation of this trend can be that the FLIS parties may want to take advantage of certain jurisdictional benefits by hosting their infrastructure in territories with complex, or flexible, copyright laws [2], [44].

[5] http://www.trafficestimate.com/rojadirecta.me

[6] Cloudflare.com provides DNS services and sits between the visitor and the CloudFlare user's hosting provider. In fact, it behaves as a reverse proxy for a website and hides its original IP address.

B. Copyright Removal Requests

In order to measure the number of copyright removal requests submitted against FLIS parties, we used Google Transparency Report[7] data that contains detailed information on requests by copyright owners or their representatives to remove URLs from Google search. At the time of our evaluation, this data contained 48,719,483 records of reported domains–roughly 95% of the copyright removal requests that Google has received since July 2011.

Findings. We used this data to find copyright removal requests against 5,685 aggregator domains and found that more than 30% have been reported at least once by copyright owners. While the majority of domains have been reported less than 50 times, there is a significant number of domains reported repeatedly by copyright owners–with cricfree.tv being the top reported domain among the investigated websites. At the time of writing, some of the domains we analyzed have already been taken down by the U.S. Immigration and Customs Enforcement and City Police of London [8], [9].

Similar to the aggregators, we also examined the copyright removal requests against 309 channel providers. In total, we found that a large number of channel providers, 199 out of 309 (64.4%), have been reported at least once by copyright owners. The most commonly reported channel provider, p3g.tv, has been reported 789 times–with a median value of 28 filed reports per week from February 2015 to June 2015.

Overall, while our analysis of copyright removal requests from Google Transparency Report can not be treated as ground truth for copyright violations committed by the investigated FLIS parties, it clearly raises questions about the lawfulness of FLIS operations.

C. Possible Trademark Infringements

Trademark infringements do not directly affect sports organizations and TV broadcasters in the same way as copyright violations, but they can have a serious impact on both industries. The Anti-Cybersquatting Consumer Protection Act (ACPA) prohibits the use of trademark domain names and logos that create confusion among viewers as to the source or sponsorship of the webpage. As such, a FLIS website should, in principle, always avoid trademark infringements.

Domain name. The infringement of domain name trademarks can be categorized into two forms: *confusability* and *dilution*. While confusability deals with trademark infringement in cases where the trademark domain is not well-known, dilution deals with leveraging the reputable trademark of a third party to refer to something unrelated–e.g., skysportslive.tv, skyembed.com. Associating “sky” with “live” and “embed” could *dilute* the powerful association between “Sky Sports” TV channel and “live streaming” in the mind of the average Internet user. This practice can be therefore presumed as a possible trademark infringement.

To measure the prevalence of possible trademark infringements in FLIS services, we first compiled three comprehensive lists containing the names of popular sports TV channels, leagues, and organizations. These lists are compiled from a variety of sources including, but not limited to, official websites of FIFA, ICC, and NBA. Next, we use the entries in these lists to search for similar substrings in all of 5,685 aggregator domain names. To search for substrings, we normalized the aggregator domains by removing all delimiters, non-alphanumeric characters, and top level domains. In addition, we also searched for several distinct keywords (such as “sky” in the previously mentioned example) in the normalized aggregator domains. Finally, if the search exposes the existence of a clearly similar name of sports TV channel, league, or organization, we manually analyzed the domain and labeled it as a potential trademark-infringing domain.

Findings. Overall, out of the 5,685 investigated aggregator domains, we found 439 (7.72%) domains using trademarks of well known sports TV channels, leagues, and organizations. More specifically, 176 domains (3.09%) were found clearly utilizing the name of sports TV channels, 67 domains (1.17%) were found to use the name of sport organizations (e.g, FIFA etc.), and the remaining 196 domains (3.44%) were using trademarks of popular leagues in their domain names.

Trademark logo. In addition to possible trademark infringements in the domain names, we also found that a number of aggregator websites are using the logos of popular sports TV channels. Under ACPA, the unauthorized use of a trademark logo in such a way as to cause viewers of the webpage to believe that the page is affiliated with the respective TV channel, is prohibited.

In order to measure the prevalence of trademark logo utilization in aggregator websites, we downloaded several hundred sports TV channel logos from Google Images. Next, we use the downloaded images as an oracle and employ a lightweight image genre recognition method to quickly identify the webpages that contains images of trademark logos. (We provide details on image recognition method in Section V.) Finally, we manually analyzed the screenshot of each identified page to verify the presence of popular TV channel logos.

Findings. Two trends in the utilization of trademark logos were observed. First, we found that the aggregator pages use the TV channel logos as the link to the FLIS video page. Second, we noticed that some aggregator pages employ the TV channel’s logo as the main logo of their page (e.g., starsportslive.tv, skysportslive.tv). Overall, out of the 5,685 investigated aggregator domains, we found 282 (4.9%) domains using the logos of popular sports TV channels.

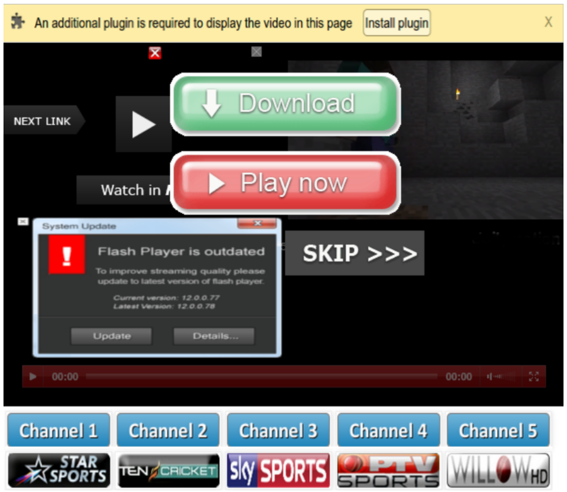
While utilization of trademark logos and domain names may not necessarily be categorized as trademark abuse (depending on jurisdiction in respective territories), such practices can easily confuse users about the nature of a FLIS service, which can, in turn, increase the probability of malware infections, as those described in the following sections.

D. Substandard, Deceptive, and Unavoidable Advertisement

In the context of this study, we analyze *overlay ads* that are unique to the online video services.[8] As described earlier (Section II-A), the overlay ads are displayed using `<iframe>` elements. These ads are typically served as images or Flash content that “overlays” the video content, usually superimposed on each other, running concurrently with the streaming

[7] http://www.google.com/transparencyreport/removals/copyright/

[8] Although, aggregators also use a variety of other ad methods to monetize their business (e.g., pop-under, pop-up etc.), we believe considering the overlay ads in our analysis potentially covers a significant breadth of illicit activities, while examining an involvement of all the key parties in the FLIS ecosystem i.e., the channel provider, the aggregator, and the advertisers.



An additional plugin is required to display the video in this page

System Update

Flash Player is outdated

To improve streaming quality please update to latest version of flash player.

Current version: 12.0.0.77
Latest Version: 12.0.0.78

Figure 6: Overlay ads example on video player taken from the popular FLIS website cricket-365.co.in. Numerous malicious overlays are stuffed on the video player, covering most part of the video player, while employing social engineering and deceptive techniques.

content. Both the aggregators and the channel providers, using an advertiser script, can overlay these ads. While watching a live sport event, a user is often presented with a number of overlay ads, generally blocking most part of the video player, and often requiring an action by the viewer to close them. Figure 6 shows one such example, displaying a number of potentially malicious overlay ads, super imposed on each other, employing both social engineering and deceptive techniques (e.g., fake close buttons ☒, fake video play button etc.), while blocking more than 75% area of the Flash player.

This practice of displaying ads not only lures a user into clicking a potentially malicious ad, but also goes against any given standard of the online advertisement industry. In this regard, we performed an empirical analysis to investigate the FLIS services' compliance, in displaying overlay ads, with the Interactive Advertising Bureau (IAB) standards.

Compliance with IAB standards. IAB is an organization that develops standards for the online advertising industry. It has defined metrics that deal with the behaviors specific to the nonlinear (overlay) video ads [13]. According to these metrics, an overlay ad should not cover more than 1/5 (20%) of the height of a player, with the best practice to display the ad at the bottom half of the streaming content.

To measure the FLIS services' compliance with IAB standards, we analyzed the crawled data that was used to identify the channel providers (see Section III-B for details). Along with the network traffic, this data contains HTML code, source URL, and absolute location and size of all `<iframe>` elements of each crawled webpage (*iframes-log*). We selected 44,960 pages from this data that were identified to be broadcasting live streams using the RTMP protocol. Next, we extracted the `pageUrl` from the network traffic of each crawled webpage by interpreting the RTMP `connect` message (Figure 3). This `pageUrl` corresponds to the source URL of an `<iframe>` that embeds a Flash player in the webpage. Once we have identified the source URL of the `<iframe>` that embeds a Flash player, we use the iframes-log to identify its location and size on the page (same as the position and size of the rendered Flash player). Finally, we again use the iframes-log to measure the occurrences and positions of other `<iframe>` elements (i.e., overlay ads) overlapping the area of Flash player. Doing this allows us to measure the average concentration of ads presented on the player by the FLIS services. Moreover, we can calculate the percentage of the player's area covered by the overlays.

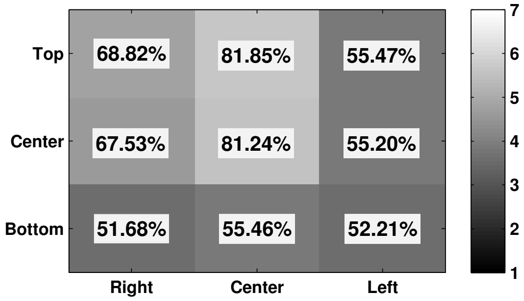


Figure 7: Heat map where the color strength indicates the average number of overlay ads on video players, and the percentage reflects ads superimposed on each other at the particular location on players.

Findings. Our measurements reveal that on average, there are 5-6 overlays present on the video players in the investigated FLIS webpages. Furthermore, on average, 93% of the video players were stuffed with overlays, hiding more than 80% area of the player. We observe that the majority of these ads consist of fake-button images displayed exactly in the center of a player to trick users into clicking. As such, this trickery directly benefits the FLIS services which earn ad commissions from unintended clicks on the ads. We also noticed that most of the displayed overlay ads were hidden under additional overlay ads. Figure 7 shows a heat map indicating the average number of ads as a color strength on a player, with 7 being highest and 1 being lowest. Furthermore, it shows the average percentage of ads superimposed on each other at different locations on the player. We can clearly observe that the majority of the overlays presented by the FLIS services were placed in the middle of video players, with more than 81% overlapping each other.

Anti-Adblock. The reader may realize that a user can avoid interactions with the overlay ads by using popular ad-block extensions. These extensions remove advertisements so that the player area otherwise stuffed with overlay ads can be cleared to view the live stream. For each visited FLIS page broadcasting live stream, we tested whether the page employed anti-adblock scripts to identify or bypass adblocking extensions.

Findings. Overall, we discovered that out of the top 1,000 investigated aggregator domains, 163 (16.3%) employed scripts that attempt to detect and defeat the ad-blockers. We noticed two most commonly used scripts. We now briefly detail their workings.

◇ *advertisement.js.* We found that several aggregators include a script named `advertisement.js`. By inspecting several versions of this script, we discovered that the mere purpose of these scripts is making a modification to the page's Document Object Model (DOM). Since the popular browser

Browsers	% Malware	% Scam	% Adult
Safari	75.71%	1.19%	4.40%
Chrome	67.26%	1.81%	1.29%
Firefox	38.65%	2.15%	5.31%
Internet Explorer	34.22%	11.3%	2.72%
Average	53.96%	4.11%	3.43%

Table III: The nature of ad websites opened after interacting with the overlay ads displayed on the video players using four different user-agent strings. The percentage values are given according to the labeled categories: malware, scam, and adult.

extensions used to filter advertisements (e.g., AdBlock, AdBlock Plus), block any script named `advertisement.js`, these DOM modification are not made when such an extension is present. This provides the aggregators with an oracle to detect the presence of an ad-blocker, which can then be used to force users to disable their ad-blockers.

◇ *antiblock*. Similar to the ad-block detection technique with `advertisement.js`, antiblock.org provides a script that allows website administrators to detect the presence of a multitude of extensions and plugins that block advertisements. In addition, the script attempts to avoid detection by a few ad-block extensions. In case the script detects that advertisements are still blocked, the default behavior is to make the webpage inaccessible. This ultimately forces users to disable their ad-blocking solutions if they want to access the page's contents.

Our analysis highlights a clear violence of the IAB standards by the FLIS services in displaying overlay ads on the video player. Moreover, our findings indicate that the parties in the FLIS ecosystem undeniably compel their users to interact with the overlay ads and adopt various deceptive techniques to gain ad revenue from the unintended clicks.

E. Exposing Users to Malware, Scam, and Adult Websites

In this section, we investigate the ad websites opened when a user is deceived (or lured) into clicking the overlay ads displayed on the video player by the FLIS parties. From our experiments, we noticed that the interaction of users with the overlay ads opened a variety of ad websites. These sites often present security threats including exposure to malware, scams, and link-hijacking.

Automated interaction with overlay video ads. To gather ad websites that opened when a user interacts with the overlay ads, we implemented an additional module in our crawler detailed in Section III-B. During the crawl, the module identifies the `<iframe>` responsible for rendering the Flash player by utilizing a whitelist of the identified channel providers. Afterwards, it uses the iframe-logs and tries to click the `<iframe>` elements (i.e., overlay ads etc.), overriding the Flash player on the webpage. Before clicking an overlay, the crawler sleeps for 20 seconds to allow any redirections. Finally, it log any redirections to different domains and capture the screenshots of the opened ad webpages. Our crawler visits each page using four different user-agent strings covering popular browsers and operating systems.

At the end of this process, our crawler collected the screenshots of 30,354 ad pages that were opened due to the clicking of overlays in the top 1,000 aggregator domains. Capturing an image of the ad page provides us with the essence of what a user would have been exposed to while interacting with the overlay ads. We observe that these images contained a large variety of sites for deceptive malware downloads, scams, and adult material.

Ethical considerations. To discover whether ads are malicious or not we have to, unavoidably, deliver clicks on them and monitor their final destination. As it becomes clear in the next paragraphs, the vast majority of ads present on FLIS services are of a malicious nature. This was not a surprise for us since the nature of ads that we encountered when considering this project was also mostly malicious and is what prompted us to conduct this study. We argue that, even though our crawler may have charged some advertisers[9] for the duration of our crawling experiment, this was probably beneficial for the Web at large since we absorbed the ads that would have otherwise victimized real users. Moreover, our crawling methodology is in line with previous studies that have seeked to understand online ads [15], [17].

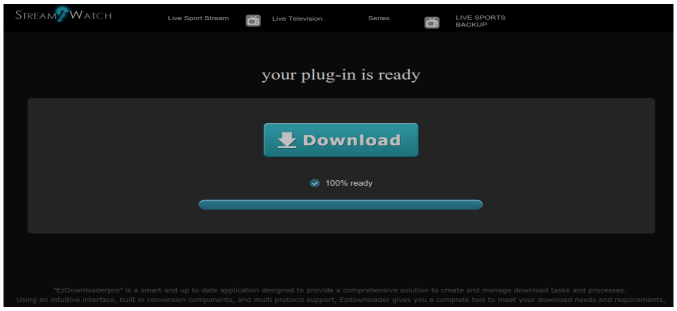
Classifying advertisements. To automatically categorize the collected ad websites we clustered the pages based on their visual appearance. We used a perceptual hash function [53] to automatically cluster the screenshots of the ad pages. A perceptual hash function returns similar hashes for two images, if one is visually similar to the other that may have gone through modifications such as scaling, aspect ratio alterations, or minor changes in color. We computed the perceptual hash of all screenshots and cluster them in groups by using the Hamming distance between hashes as our distance metric. If the distance between two hashes was less than an empirically calculated threshold[10], we clustered the corresponding pages. By using the perceptual hash functions, we achieved a precision of 99.8% and recall of 98.4% (compared against manually generated ground-truth of 1,000 screenshots). Once the clustering process finished, we manually verified the clusters, and examined each cluster for malware, scam, and adult ads. We categorize ads as malware when they lead to the installation of malware (binary or browser extension).

Findings. Table III shows the results of our clustering and labeling, separated by four user-agent. The first thing to notice is that, on average, 50% of the time, a click on an overlay ad leads the user to a malware-hosting webpage. The majority of malware-hosting pages were constructed to imitate the look and feel of the FLIS services, often trying to trick the user to install malware by pretending that she needs special software (binary or extension) to watch the live stream. Figure 8 shows an example of such a webpage, that was opened after clicking an overlay ad on stream2watch.com. This page is trying to trick the user into downloading a malicious plug-in as if it were provided by original streaming website.

Meanwhile, in Table III, one can notice that specific browsers were much more exposed to the malware-hosting webpages than others. Chrome and Safari, the two most

[9] Most reputed ad networks have deployed detection mechanisms to filter bot generated clicks and cease charging the advertisers when they identify artificial traffic [14], [21], [49].

[10] We select the value 0.3 as threshold. To do that, we computed the clustering accuracy on a subset of the extracted screenshots for each threshold value between [0,1] with a step of 0.1 [25]. A threshold of 0.3 achieved best precision and recall.



your plug-in is ready

"EzDownloaderpro" is a smart and up to date application designed to provide a comprehensive solution to create and manage download tasks and processes. Using an intuitive interface, built in conversion components, and multi protocol support, Ezdownloader gives you a complete tool to meet your download needs and requirements,

Figure 8: Screenshot of a malicious website that opened after interacting with overlay ads. The website imitates the look and feel of the FLIS webpage stream2watch.com to deceive users.

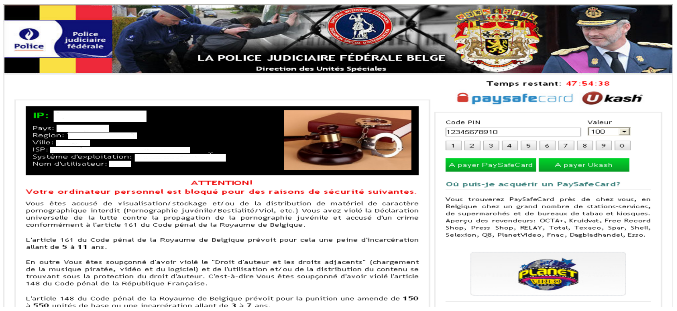
popular browsers, are the ones most exposed to the malware pages through overlay ad clicking. One logical reason for this trend is that, as depicted in recent security studies, attackers are more inclined in targeting the popular browser(s) for ad injections and malversting [48], [51], [30]. As such, for Internet Explorer and Firefox, it may be more beneficial for malicious advertisers, along with presenting malware sites, to expose FLIS users to money laundering scams, adult gaming/video websites, and fraudulent technician services.

Overall, while advertisers are the root cause for malicious ads, the involvement of the FLIS parties cannot be entirely exempted as they expose their users to security threats. From the prevalence of discovered abuse, it is evident that the FLIS parties are more inclined towards malicious advertisers to monetize their operations, exposing their users to malware-laden domains, fraudulent scams, and adult content.

F. Additional Malicious Activities

Immediate distributor of malware. During our experiment with overlay ads, we accidentally found that seven similar aggregator domains resolving to same IP address, were distributing a malicious Android application. When visited through a specific mobile browser user agent the website redirects to m.liveonlinetv247.info. This webpage offers a complimentary application to watch free live sport streams on the mobile device. The offered application is an adware known as Android Airpush. It contains an advertising package that has the capability to display advertisements without user content and carry out potential ad fraud. This finding highlights the direct involvement of the FLIS service in exploiting users for monetary gains.

Link hijacking. We also observed link hijacking while analyzing the overlay ads in FLIS domains. As mentioned earlier, FLIS services use advertiser scripts that render multiple `<iframe>` elements to place overlay ads on the video player. This use of an `<iframe>` adequately separates the advertiser from the including page, as the advertising scripts cannot interact with the DOM of the parent frame because of the Same-Origin Policy (SOP) [11], a web application security specification. However, there are certain exemptions in SOP that allow all frames to navigate any other frame which they can reference. As an example, an advertiser's overlay `<iframe>` can redirect the entire FLIS page to a different target by using Javascript's `window.top.location.href` method. This method allows any child `<iframe>` to access the location of the topmost window, in the windows hierarchy, and can redirect it to a potentially malicious webpage in the same tab.


Police
Police judiciaire fédérale
LA POLICE JUDICIAIRE FÉDÉRALE BELGE
Direction des Unités Spéciales
PLANET VIDEO

Temps restant: 47:54:38

paysafecard Ukash

IP: [redacted]
Pays: [redacted]
Region: [redacted]
Ville: [redacted]
ISP: [redacted]
Système d'exploitation: [redacted]
Nom d'utilisateur: [redacted]

ATTENTION!

Votre ordinateur personnel est bloqué pour des raisons de sécurité suivantes.

Vous êtes accusé de visualisation/stockage et/ou de la distribution de matériel de caractère pornographique interdit (Pornographie juvénile/Bestialité/Viol, etc.) Vous avez violé la Déclaration universelle de la lutte contre la propagation de la pornographie juvénile et accusé d'un crime conformément à l'article 161 du Code pénal de la Royaume de Belgique.

L'article 161 du Code pénal de la Royaume de Belgique prévoit pour cela une peine d'incarcération allant de **5** à **11** ans.

En outre Vous êtes soupçonné d'avoir violé le "Droit d'auteur et les droits adjacents" (chargement de la musique piratée, vidéo et du logiciel) et de l'utilisation et/ou de la distribution du contenu se trouvant sous la protection du droit d'auteur. C'est-à-dire Vous êtes soupçonné d'avoir violé l'article 148 du Code pénal de la République Française.

L'article 148 du Code pénal de la Royaume de Belgique prévoit pour la punition une amende de **150** à **550** unités de base ou une incarcération allant de **3** à **7** ans.

Code PIN: 12345678910
Valeur: 100

Où puis-je acquérir un PaySafeCard?

Vous trouverez PaySafeCard près de chez vous, en Belgique chez un grand nombre de stations-services, de supermarchés et de bureaux de tabac et kiosques. Aperçu des revendeurs: OCTA+, Kruidvat, Free Record Shop, Press Shop, RELAY, Total, Texaco, Spar, Shell, Selexion, Q8, PlanetVideo, Fnac, Dagbladhandel, Esso.

Figure 9: Screenshot of a scam website to which our crawler was redirected after visiting a FLIS page. The webpage pretends to be from a regional law enforcement office which demands a sum of money as a "penalty" for a purported crime that our crawler committed.

We found that 1.6% of ads displayed on the crawled FLIS pages had escaped their `<iframe>` and redirected the entire page to a malicious website. We observed two different types of abuses in these malicious websites. In the first type, the malicious page imitates adobe.com and offers the malware disguised as the latest version of Adobe Flash. In the second type, the webpage shows pop-up *modals* that prevent the normal use of the browser until the user accepts or rejects the download or pays the fraudulent ransom. Figure 9 shows an example of such an ad we observed, demanding a fraudulent ransom from the regional law enforcement.

G. Malicious Payloads Investigation

To find out more about the nature of malware offered to the users of FLIS services, we downloaded all the payloads from the labeled malware ad webpages. We found that based on the user-agent string, in other words, based on the browser and OS of a victim, malicious domains present environment-specific payloads. For instance, a user with a Google Chrome browser was presented with a malicious extension when redirected to the ad website. Similarly, an executable was presented for Firefox running on Microsoft Windows, an Apple Mac image for Safari OS X and so on. To analyze malware binaries we used the VirusTotal (VT) service [10], to determine whether the binary had ever been scanned before and whether it was labeled as malicious by an antivirus vendor. To examine malicious Chrome extensions, we leverage the techniques from [48], [51] and manually analyze the behavior of collected extensions in the browser.

Malware binaries and their distributors. Table IV summarizes the malware dataset obtained during our analysis of FLIS services. During our experiments with overlay ads, we downloaded 12,683 malware payloads, yielding 1,353 distinct binaries, out of which 629 samples were unknown to VT. This means, that at the time of scanning the binary using the VT service, the binary was not in the VT's database. Of these, one binary was initially classified as benign by all the AV in VT and later labeled as malware by a reputable AV after the few days re-scan. Thus, this file is considered as a *zero-day* malware sample. At the same time we noticed that, most of

Malware Obtained		Top Advertisers
Total Binaries	12,863	1. 3c41ddc0.se
Distinct (by SHA1)	1,353	2. s.ad[0-9]{3}m.com
Unknown to VT	629	3. creative.ad[0-9]{3}m.com
Zero-Day	1	4. ad.directrev.com
Malicious domains	96	5. vipcpms.com

Table IV: Summary of malware collected from the ad webpages displayed by the FLIS services and top 5 advertisers leading to malware domains.

Rank	Extension Name	User base	Redirection %
1	iLivid	10,000,000+	20.72%
2	Search-By-Zooms	1,192,815	17.72%
3	Free-Games-Zone	1,880,238	7.47%
4	Musix-Search	462,934	5.33%
5	Retrogamer	183,675	0.09%
6	Support-our-Cause	115,667	2.34%
7	tabSent	22,340	4.73%
8	Zwinky	20,156	0.09%
9	Search-Point	2,840	1.16%
10	Zapyo	2,648	0.94%
11	GamesFanatic	328	0.02%

Table V: Malicious extensions discovered during the analysis of overlay ads displayed by the FLIS services.

the malware samples submitted to VT belong to families like fake installers for malicious browser plug-ins, adwares, and browser activity monitors.

Most of the websites offering malware were reached due to a small number of advertisers providing overlay ads. The right side of the Table IV shows the top 5 advertisers leading to the malware website when clicked on the overlay ads. These advertisers are either directly malicious or have been used as intermediaries in the delivery of malicious ads.

Malicious extensions and their distributors. Table V lists the 11 extensions we found from the ad websites opened after clicking on the overlay ads using a Chrome specific user-agent. The table also shows the percentage of redirections to the ad websites presenting these extensions. Moreover, the table presents the user base of each extensions as shown on the Chrome web store. We can observe that some of these extensions have millions of active victim users. We manually analyzed these extensions and flagged several malicious activities like ad-injection, hijacking new-tab pages, and injection of malicious `<iframe>`. Overall, our analysis flagged most of the extensions as malicious and few of them as suspicious. It might be the case that the latter category are legitimate extensions, but we consider this possibility highly unlikely, as these extensions are neither particularly useful nor well designed, yet have somehow amassed millions of installations.

Figure 10 shows the redirection chains to the opened websites, offering the discovered malicious extensions, after clicking the overlay ads. The graph shows that some of the domains, like lp.ilividnewtab.com, lp.gamesnewtab.com, can be reached through several intermediate entities after a click on an overlay ad. In other cases, we can observe a webpage, presenting a malicious extension, being reached more directly, for example, zwinky.com, retrogamer.com from a single advertiser i.e., adcash.com. Overall, these interactions and intermediate party redirections hide the direct association of the FLIS services in serving the malicious browser extensions.

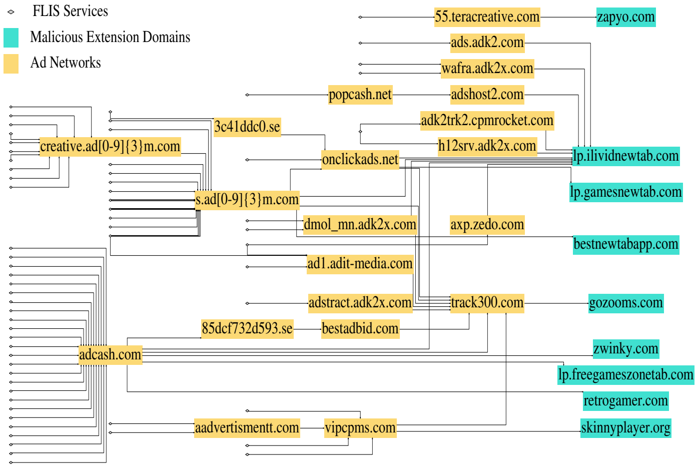


Figure 10: Redirection chains leading to domains offering malicious browser extensions.

H. Summary of Findings

By performing more than 850,000 visits on the identified 5,685 aggregator domains and by analyzing more than 1 Terabyte of traffic, we found that the majority of the parties in the FLIS ecosystem are hosting their infrastructure mostly in Europe and Belize. For instance, we discovered that nearly 25% of live streams originates from the servers hosted in Belize, and more than 60% of analyzed streams originates from the media servers provided by only 5 companies located in Belize, Switzerland, the Netherlands, Sweden, and Canada. Additionally, we found that more than 64% of parties providing these streams have been reported at least once for violating the copyrights of content owners. Since only a handful of channel providers are responsible for broadcasting the majority of the live streams, we argue that a strict control on the operations of these entities, can effectively minimize the volume of illegal live streaming.

In addition, we found that 5-7% FLIS pages leverage trademark names and logos of popular TV channels and sports organizations to attract more visitors. Moreover, through a series of experiments, we found that FLIS services do not respect the standards for the online advertising. We discovered that, on average, 93% of the video players on FLIS webpages were stuffed with overlay ads, hiding more than 80% area of the player. Furthermore, the displayed overlay ads found in FLIS services are embossed with deceptive close buttons to collect unintended clicks from visitors, leading to visitors opening advertisement websites while trying to close these overlays.

Finally, we examined the nature of websites opened when users are deceived (or lured) into clicking on overlay ads. On average, we found that 50% of ad-related websites are malicious in nature, offering malware, malicious browser extensions, and all sorts of scam pages. At the same time, we discovered that some FLIS services are directly involved in malware distribution via an Android application. Overall, these practices, along with the frequent accusation of copyright infringement, clearly show that FLIS services are inclined towards intrusive and malicious monetization schemes, at the expense of user security.

V. FLIS Classifier

Given the ever-increasing incidents of copyright violations and discovered abuse, both against users as well as legitimate content providers, it is clear that current FLIS services are a rather parasitic part of the Web. As such, automatic detection techniques are necessary to identify the aggregator webpages serving viewers with an index of free streams, most of which are commonly reported as illegal. To this end, we designed and developed FLIS classifier, a system that is able to perform online detection of FLIS aggregator pages. Our data gathering infrastructure (Section III-A) already demonstrated a real-world utilization of our classifier where we deployed it to identify unknown FLIS pages. As an application, our classifier can readily be used by human analysts to find unknown FLIS websites that can then be analyzed for potential abuses.

We now describe the architecture of the FLIS classifier by providing details on feature extraction, implementation strategies, and a performance evaluation on the gathered dataset.

A. Feature Extraction

When extracting features for the FLIS classifier, we set the following requirements: (1) features should target the *look and feel* and modus operandi of the FLIS aggregator pages (for accurate classification), and (2) the feature extraction process should be efficient in terms of processing overhead. To this end, we focus on extracting information from various live sport streaming indicators, network traffic, specific widgets, and from the images found on a webpage. Specifically, we extract five HTML features, three network traffic features, two frame features, and two image features. To extract these features, we crawl a webpage using a crawler based on Selenium, a testing framework for web applications, while storing the HTML of every loaded `<frame>` and `<iframe>` element, collecting all images in the page, logging network traffic, and taking a screenshot of the webpage. For each crawled page, the extracted features are incorporated in a feature vector. In the following paragraphs, we provide details on the features and our intuition for choosing them.

HTML features. This set of features is obtained by interpreting the HTML code of every loaded `<frame>` and `<iframe>` in a webpage. As such, this set models information from the look and feel of the aggregator pages.

◇ **Element_text-to-global_text ratio.** Typically, the main body of an aggregator page does not contain much text. The majority of text found on these pages resides in specific HTML elements, such as links and meta elements i.e., `<a>`, `<description>`, `<title>`, `<keywords>`. To exploit this trait, we use the Python goose-extractor module [4] to extract the text from link elements, meta elements, and the main body. We then measure the ratio of text that is located within the link and meta elements to the global amount of text found on the page. We use this ratio as a numeric feature.

◇ **Number of indicative words in URLs.** Aggregator pages usually contain text in `<a>` elements and indexed URLs of live streams, which generally represents numerous sports, TV channels, and events between different countries or sports clubs (e.g., `/watch/baseball/foxsports.html`). To use this specific characteristic as a set of features, we compiled four comprehensive word lists that contain several hundred entries of different sports, countries, sports clubs, and sports TV channels. We then use these lists to search for indicative words in the visible text of `<a>` elements, and in normalized text extracted from the URLs. We use a token extraction technique from [18] to extract the normalized text from URLs. For example, `/watch/baseball/foxsports.html` would be split into the tokens `watch`, `baseball`, `foxsports`, and `html`. Finally, we count the occurrences of the indicative words for each of the four categories on a webpage and include the count values as a set of numeric features.

◇ **n-grams.** In addition to indicative words, we also measure the presence of FLIS representative word sequences (n-grams) in a webpage. The intuition is that n-grams which appear much more frequently in aggregator pages than in non-FLIS ones are a good marker for FLIS aggregator pages (e.g., “watch free live football streams” etc.). To this end, we extract n-grams that vary from length $n = 1$ to $n = 5$ from the meta elements, the visible text of `<a>` elements, and from any text found on the main body of the known aggregator pages. Afterwards, we select the top 1,000 n-grams by measuring their importance in the known FLIS aggregator and non-FLIS pages using TF-IDF [42]. Lastly, we measure the frequency of the selected n-grams on a webpage and incorporate these frequencies as a set of numeric features.

◇ **Presence of indicative widgets.** We also inspect the presence of FLIS indicative widgets on a webpage. These widgets are stand-alone applications from particular third-parties which are embedded into the aggregator webpage. Specifically, we found that a certain type of aggregator webpages often contain specific stream (like http://ifirstrow.eu/webmaster/ etc.), chat, and clock widgets. Hence, presence of these widgets on a page is a good indicator of the FLIS aggregator page. Therefore, we compiled a list of indicative widgets’ URLs from the known aggregator pages. We then use these lists to identify the presence of FLIS indicative widgets in a page and incorporate this knowledge as a set of boolean features.

◇ **Presence of reporting link.** Aggregator domains typically host a *notice page* for reporting of illegal streams indexed on their websites. We observe that the visible text in the link (`<a>` element) of these notice pages usually has keywords such as “dmca”, “noticetakedown”, “notice”, “report” etc. We use these characteristics and compile a list of notice keywords from the known aggregator pages. Afterwards, we use this list to identify the presence of notice keywords in the visible text of `<a>` elements of a webpage and use it as a boolean feature.

Network traffic features. This set of features is extracted from the network trace that is recorded while crawling a page. By its very nature, this set models information from the modus operandi of the FLIS aggregator pages.

◇ **Third-party request ratio.** This feature deals with the third-party content on the aggregator webpage. Aggregator pages often include HTML content from third-party services, such as the use of frames provided by channel providers, the overlay ads etc. To incorporate this information as a numeric feature, we measure the ratio of HTTP requests to third-parties (other domains) compared to the total amount of the HTTP requests.

◇ **Presence of media traffic.** We observe that it is also common for aggregator webpages to have a Flash player embedded on the page along with the indexed streaming links (as shown in Figure 6). The player broadcasts live sport streams using a specific media protocol (e.g., RTMP etc.). To capture this fact, we use the protocol signatures (described in Section III-B) to detect the presence of the media traffic and incorporate this information as a boolean feature.

◇ **Non-standard port streaming.** In connection with the previous feature, this feature indicates a presence of media-related traffic on non-standard ports. The rationale for incorporating this information, is that FLIS services often use non-standard ports for transmitting media traffic using standard protocols (e.g., RTMP on port 443 etc.). We expect that most non-FLIS websites broadcast media traffic using the standard protocol ports. Thus, we include a boolean feature that indicates the use of a non-standard protocol port for broadcasting streams.

Frame features. This set of features is extracted by analyzing all loaded `<frame>` and `<iframe>` elements on the aggregator webpage, in an effort to further model the workings of the FLIS aggregator pages.

◇ **Number of frames.** The usage of `<iframe>` elements to index streaming links, embed video players, show ads etc. is very common in FLIS services. Thus, we incorporate this characteristic, as a set of numeric features, by counting the number of `<frame>` and `<iframe>` elements found on a page in combination with their child `<frame>` elements, in a recursive fashion.

◇ **Average and maximum nesting of frames.** In addition to the number of frames, we also include a set of numeric features to measure the average and maximum level of nesting of any given `<iframe>`. The rationale is that most of the aggregator pages include deeply nested `<iframe>` elements to display ads on videos or to embed third-party stream widgets. We expect non-FLIS websites to have fewer nested `<iframe>` elements than the FLIS aggregator webpages.

Image features. This set of features aims to model information from images on the FLIS aggregator pages, focusing on the look of FLIS aggregator webpages.

◇ **Average and maximum image size.** It is common for aggregator pages to have several images of sports equipments, flags, sports clubs, and TV channel logos. These images are often placed alongside the indexed links of different TV channels and sport streams. We observe that these images are often small in size (on average 1.4 kilobytes) and account for the majority of images on the aggregator webpages. Therefore, we measure the average and maximum size of images found on a webpage and incorporate these measurements as a set of numeric features.

◇ **Ratio of indicative images.** As discussed earlier, aggregator pages make heavy use of images that belong to sports equipments, country flags, clubs, and TV channel logos. Thus, a high ratio of these indicative images in comparison to all other images on a webpage, is a good indicator of a FLIS aggregator webpage. To measure the number of FLIS indicative images, we first extracted all the images from the known aggregator pages. Next, we clustered these images using a perceptual hash function (PHash) (as described in Section IV-E). Afterwards, we manually inspect the clusters and remove all the irrelevant clusters (i.e., banners etc.). At this point, all remaining clusters belong to the FLIS-indicative image categories (i.e., sports equipments, country flags, TV channels and club logos). In our approach, each cluster is expressed by a candidate PHash value, which is representative of all images in it. Now, to measure the ratio of FLIS indicative images, we extract all the images from a page, compute their PHash values, and use the Hamming distance as metric, between the candidate PHash value of all the clusters and the PHash values of the extracted images. If the distance between any of the extracted image's PHash and any candidate PHash value is less than 0.3, we label the image as FLIS indicative. Finally, we measure the ratio of the labeled FLIS indicative images to the other images found on a page and incorporate this ratio as a numeric feature.

B. Evaluation

Implementation. To build the FLIS classifier, we opt for a supervised learning approach. In this approach, we first collected a set of labeled webpages, and we have used this set to train the classifier by extracting the feature vectors. Once the classifier is trained, a new page is crawled, translated into the feature vector, and passed to the FLIS classifier. The FLIS classifier then labels the page as a FLIS or non-FLIS page. Specifically, for each new page, the classifier outputs a score. If this score is greater than a selected threshold, the classifier labels the page as a FLIS aggregator page.

The components of the FLIS classifier used to crawl a website and extract the features are written in Python. To build the FLIS classifier, we used the Random Forest algorithm [19] implementation in Weka [27]. The rational of using the random forest algorithm is that it is fast, robust with regards to outliers, yields extremely accurate predictions, and can process a large number of input features without overfitting. To foster future research in FLIS services, we will be making the prototype implementation of the FLIS classifier publicly available.

Evaluation datasets. To evaluate the performance of the FLIS classifier, we assembled three different datasets that we carefully examined and labeled. We now provide the details on each of the datasets.

◇ **Balanced dataset (BD)**: The balanced dataset consists of an equal number of positive (FLIS aggregator pages) and negative training samples (non-FLIS pages). We collected non-FLIS pages by randomly crawling the links of Alexa top 1,000 domains and label each instance through manual inspection. For aggregator pages, we extract the subset of webpages from the gathered dataset and label each page through manual inspection. Overall, this dataset consists of 3,500 aggregator pages and 3,500 non-FLIS pages.

◇ **Imbalanced dataset (ID)**: Besides the balanced dataset, we also evaluate the performance of the FLIS classifier on an imbalanced dataset. In reality, there are more non-FLIS webpages than FLIS aggregator pages. This unequal distribution of webpages can bias the performance of classifier towards the majority class (i.e., non-FLIS pages). To exhaustively evaluate the discriminative nature of features and the classification algorithm, we built a dataset with a class imbalance ratio of 10 to 1. Specifically, the imbalanced dataset contains 15,000 non-FLIS and 1,500 FLIS aggregator pages.

◇ **Special testing dataset (STD)**: In both the balanced and imbalanced datasets, the nature of the negative training samples is substantially different from the positive samples. As such, an evaluation on only these datasets will represent our classifier's ability to distinguish between FLIS webpages from entirely different non-FLIS webpages (e.g., fb.com and bbc.com). To

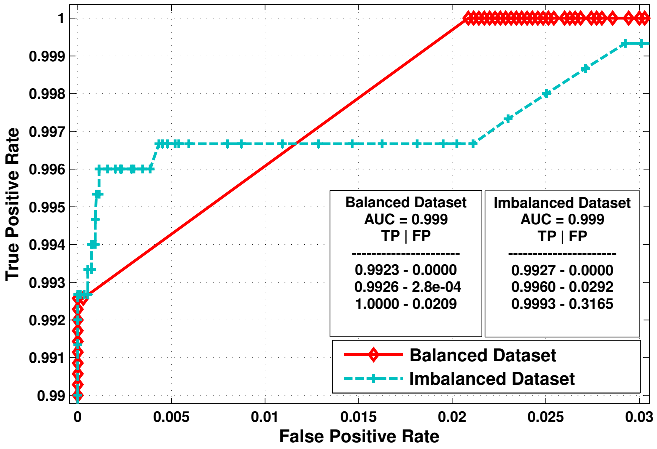


Figure 11: Zoomed ROC curves of the FLIS classifier on balanced & imbalanced datasets.

this end, we assemble an additional special testing dataset to evaluate our classifier's ability of distinguishing between common sports webpages and FLIS webpages. This dataset contains 1,000 randomly crawled non-FLIS pages listed under the "Sports" category of the open directory project [3].

Cross validation. To evaluate the detection accuracy of the FLIS classifier, we perform a 10 fold cross-validation test on both the balanced and imbalanced datasets. In the 10 fold cross-validation test, the dataset is randomly divided into 10 smaller subsets, out of which 9 subsets are used for training the classifier and 1 subset is used for testing (unseen pages during training). This process is then repeated 10 times, with each of the 10 subsets used exactly once as the testing data. To avoid any artifacts, we repeat the process of *10 fold cross-validation 10 times*, each time the data is randomly divided into 10 smaller subsets with a different seed value.

Figure 11 shows the ROC curves that we obtained on averaged results of the tests for the balanced and imbalanced dataset. In order to emphasize the FLIS classifier performance at low false positives, we plot the zoomed-in ROC where the false positive rates range from 0% to 3%. The tables shown in Figure 11 provide details of the area under the ROC (AUC) and the trade-off between the true positive (TP) rate and false positive (FP) rate for a few preferred operating points on the balanced and imbalanced ROC curves. The 99.9% area under the curve (AUC) for both datasets, shows that our classifier can properly handle both balanced and imbalanced datasets. Moreover, we can see that for both datasets, when we select a threshold value to achieve a false positive rate of 0%, the classifier still yields a true positive rate of approximately 99.2%. These results indicate the accuracy of our classifier when distinguishing the FLIS webpages from entirely different non-FLIS webpages.

STD experiment. For this experiment, we first train a model on 15,000 non-FLIS pages (from **ID**) and 3,500 FLIS pages (from **BD**). The trained model is then used to classify the special testing dataset (**STD**). Out of the 1,000 non-FLIS sports pages in **STD**, the FLIS classifier misclassified only 3 pages as FLIS aggregator webpages (false positive rate = 0.3%) demonstrating the accuracy of the FLIS classifier when dealing with the non-FLIS sports webpages.

Run-time performance. In addition to classification results, we also measured the run-time performance of the FLIS classifier. Running as a single threaded application on a 64-bit 2.8 GHz Intel Core i5 CPU, our classifier takes 18.4 seconds on average to classify a given webpage, the slowest being 75 seconds. The most expensive processes are extracting the features from the HTML sources and network trace. These processes are IO bound and account for the majority of the runtime. Overall, in our data gathering process (as demonstrated in Section III-A), we observed that the FLIS classifier scales well in an online process.

Classifier evasion. The presented FLIS classifier is built on attributes targeting the look and modus operandi of aggregator pages. Therefore, an aggregator's attempt to purposefully evade detection is not an easy task. While an adversary can evade a few specific features used in the learning process, this is likely going to result in either increased operating costs, or a loss of a percentage of their viewers. For instance, if aggregators stop using text and images related with sports and legitimate broadcast channels, they are likely to attract less users to click on their links and interact with their malicious ads. Overall, we argue that our FLIS classifier provides robust detection of aggregator pages, which could be used both by law enforcement as part of a take-down process, as well as by users who may confuse an aggregator page for a legitimate service provided by a reputable channel.

VI. Related Work

There is a significant amount of prior work on the piracy of live broadcasts from a legal perspective. Specifically, the focus of this research is on highlighting copyright law [28], [33], [36], [50], analyzing the consequences of piracy on related organizations [16], [24], [47], and issuing proposals to improve judicial conducts [28]. In contrast to these studies, we map the FLIS ecosystem through real-world experiments and empirically quantify the threats for both users landing on FLIS domains, as well as for related companies whose copyrights and trademarks are potentially abused by FLIS services.

Other research has focused on analyzing malicious advertisement in the context of online fraud [26], [32], [46], and how certain Internet services have been abused for monetary gain [20], [34], [51], [52]. Studies that specifically target deceptive advertisement techniques mainly focus on examining the security implications of deceptive ad banners [23], [37]. Our work differs from these studies in that it focuses on the interactions of users with the, practically unexplored, video overlay ads and the numerous threats associated with it.

VII. Conclusion

In this paper, we presented the results of the first empirical study of free live streaming services. We developed an infrastructure that enabled us to map the ecosystem of FLIS services, identify the parties that facilitate anonymous broadcast of live streams, and analyze the deceptive advertising content that users are exposed to when they watch live broadcasts on FLIS websites. In this process, we discovered various types of abuse including malware distribution, malicious browser extensions, substandard overlay advertising, and scams that could cost users their personal information as well as financial loss.

Given the extent of the observed abuse and the large number of copyright complaints, we engineered a classifier that can be used to, among others, alert users that they are currently interacting with potentially dangerous FLIS page, or help analysts find unknown FLIS pages in an effort to curb copyright infringements. We employed the proposed classifier in an online process to find new aggregator pages, and showed that our classifier achieves high accuracy with low false positives.

VIII. Acknowledgments.

We thank Jesse Davis for providing his insightful input on the evaluation of FLIS classifier, Alexandros Kapravelos for helping us with analyzing the Chrome extensions, Nicolas Christin for his help with the preparation of the final version of this paper, and the anonymous reviewers for their valuable comments. We also thank Amazon for providing us with the virtual machines that enabled our large-scale, advertisement-characterizing experiments. For KU Leuven, this research is partially funded by the Research Fund KU Leuven, with the financial support from the Prevention of and Fight against Crime Programme of the European Union (B-CCENTRE), and by the EU FP7 project NESSoS. For Stony Brook University, this work was supported by the National Science Foundation (NSF) under grant CNS-1527086.

References

[1] HTTP Live Streaming. https://goo.gl/crRhm9.

[2] InfoSoc Directive 2001/29. http://goo.gl/SOVjac.

[3] Open Directory Project. https://www.dmoz.org/.

[4] Python-Goose. https://github.com/grangier/python-goose.

[5] Real Time Messaging Protocol. http://goo.gl/d1NO9l.

[6] Real Time Streaming Protocol. https://www.ietf.org/rfc/rfc2326.txt.

[7] Realnetworks Incorporation History. http://goo.gl/IxHQRB.

[8] Siezed Domain. http://atdhe.net/.

[9] Siezed Domain. http://frombar.com/.

[10] VirusTotal. https://www.virustotal.com/.

[11] W3C: Same Origin Policy - Web Security. http://goo.gl/Xps3Ph.

[12] Wizwig taken-down. http://www.wiziwig.tv/offline.html.

[13] Digital video in-stream ad format guidelines and best practices. 2008. http://www.iab.net/media/file/IAB-Video-Ad-Format-Standards.pdf.

[14] Sumayah Alrwais, Kan Yuan, et al. Understanding the dark side of domain parking. In *USENIX Security*, 2014.

[15] Marco Balduzzi, Manuel Egele, Engin Kirda, Davide Balzarotti, and Christopher Kruegel. A solution for the automated detection of click-jacking attacks. In *ASIA CCS*, 2010.

[16] Barclay Ballard. Premier League knocks out Wiziwig in illegal streaming crackdown. http://goo.gl/ETCjH2.

[17] Paul Barford, Igor Canadi, Darja Krushevskaja, Qiang Ma, and S Muthukrishnan. Adscape: Harvesting and analyzing online display ads. In *WWW*, 2014.

[18] Eda Baykan, Monika Henzinger, Ludmila Marian, and Ingmar Weber. Purely URL-based topic classification. In *WWW*, 2009.

[19] Leo Breiman. Random forests. *Machine learning*, 45(1), 2001.

[20] Nicolas Christin, Sally S Yanagihara, and Keisuke Kamataki. Dissecting one click frauds. In *CCS*, 2010.

[21] Vacha Dave, Saikat Guha, and Yin Zhang. Viceroi: Catching click-spam in search ad networks. In *CCS*, 2013.

[22] Holger Dreger, Anja Feldmann, Michael Mai, Vern Paxson, and Robin Sommer. Dynamic application-layer protocol analysis for network intrusion detection. In *USENIX Security*, 2006.

[23] Sevtap Duman, Kaan Onarlioglu, Ali Osman Ulusoy, William Robertson, and Engin Kirda. Trueclick: automatically distinguishing trick banners from genuine download links. In *ACSAC*, 2014.

[24] Aaron Elstein. Web pirates are stealing from sports broadcasters. http://goo.gl/TVOxRi.

[25] Rgis Gras, Einoshin Suzuki, Fabrice Guillet, and Filippo Spagnolo. *Statistical Implicative Analysis*. Springer, 2008.

[26] Hamed Haddadi. Fighting online click-fraud using bluff ads. *SIGCOMM Computer Communication Review*, 40(2), 2010.

[27] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1), 2009.

[28] Stephanie N Horner. DMCA: Professional sports leagues' answer to protecting their broadcasting rights against illegal streaming. *Marq. Sports L. Rev.*, 24, 2014.

[29] Luca Invernizzi, Paolo Milani, Stefano Benvenuti, Christopher Kruegel, Marco Cova, and Giovanni Vigna. EvilSeed: A guided approach to finding malicious web pages. In *Oakland*, 2012.

[30] Nav Jagpal, Eric Dingle, Moheeb Abu Rajab, Panayiotis Mavrommatis, Niels Provos, and Kurt Thomas. Trends and lessons from three years fighting malicious extensions. In *USENIX Security*, 2015.

[31] Dave Lee. Premier league wins piracy block of first row sports. http://www.bbc.com/news/technology-23342349.

[32] Zhou Li, Kehuan Zhang, Yinglian Xie, Fang Yu, and XiaoFeng Wang. Knowing your enemy: Understanding and detecting malicious web advertising. In *CCS*, 2012.

[33] Michael J Mellis. Internet piracy of live sports telecasts. *Marq. Sports L. Rev.*, 18, 2007.

[34] Nick Nikiforakis, Federico Maggi, Gianluca Stringhini, M Zubair Rafique, Wouter Joosen, et al. Stranger danger: exploring the ecosystem of ad-based url shortening services. In *WWW*, 2014.

[35] Allison Nixon and Christopher Camejo. DDoS protection bypass techniques. In *Black Hat Briefings*, 2013. https://goo.gl/58Ah2j.

[36] Association of Internet Security Professional. Illegal streaming and cyber security risks: A dangerous status quo?, 2014. http://goo.gl/WE43IM.

[37] Kaan Onarlioglu, Utku Ozan Yilmaz, Engin Kirda, and Davide Balzarotti. Insights into user behavior in dealing with internet attacks. In *NDSS*, 2012.

[38] Cisco Press Release. Global Internet traffic projected to quadruple by 2015. http://goo.gl/MXi3pN.

[39] Niels Provos, Mavrommatis Panayiotis, Moheeb Abu Rajab, and Fabian Monrose. All your iframes point to us. In *USENIX Security*, 2008.

[40] M Zubair Rafique and Juan Caballero. FIRMA: Malware clustering and network signature generation with mixed network behaviors. In *RAID*. 2013.

[41] M Zubair Rafique, Ping Chen, et al. Evolutionary algorithms for classification of malware families through different network behaviors. In *GECCO*, 2014.

[42] Gerard Salton and MJ McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill Book Co, 1983.

[43] Homeland Security Investigations. Curbing illegal streaming: The investigation and the case. Anti-Piracy and Content Protection Summit. http://www.antipiracycontentsummit.com/media/1000508/44011.pdf.

[44] Chris Smith. Pirating copyrighted content is legal in Europe, if done correctly. http://goo.gl/G6OoCh.

[45] Pete South. Illegal football streams war shows no sign of ending for premier league. http://goo.gl/ajnxcQ.

[46] Kevin Springborn and Paul Barford. Impression fraud in on-line advertising via pay-per-view networks. In *USENIX Security*, 2013.

[47] Christina Sterbenz. How sketchy streaming sites really work and why some are legal. http://goo.gl/e6FYXo.

[48] Kurt Thomas, Elie Bursztein, Chris Grier, et al. Ad injection at scale: Assessing deceptive advertisement modifications. In *Oakland*, 2015.

[49] Alexander Tuzhilin. The Lanes Gifts v. Google Report. http://bit.ly/13ABxSZ.

[50] Carson S Walker. A la carte television: A solution to online piracy. *CommLaw Conspectus*, 20, 2011.

[51] Xinyu Xing, Wei Meng, Byoungyoung Lee, et al. Understanding malvertising through ad-injecting browser extensions. In *WWW*, 2015.

[52] Apostolis Zarras, Alexandros Kapravelos, Gianluca Stringhini, et al. The dark alleys of madison avenue: Understanding malicious advertisements. In *IMC*, 2014.

[53] Christoph Zauner. Implementation and benchmarking of perceptual image hash functions. Master's thesis, Upper Austria University of Applied Sciences, 2010.